

Sensur av hovedoppgaver

Høgskolen i Buskerud og Vestfold

Fakultet for teknologi og maritime fag



Prosjektnummer: **2014-3**

For studieåret: **2013/2014**

Emnekode: **SFHO3201**

Prosjektnavn

GCS - Gir- & Clutch Simulator

GCS - Gear & Clutch Simulator

Utført i samarbeid med: Kongsberg Automotive

Ekstern sensor: Tom Heistad

Ekstern veileder: Espen Elvekrok

Sammendrag: I prosjektet har vi laget en simulator som skal kunne gjengi motstanden en lastebilgirskasse gir. Når man girer med girspaken, skal det kjennes ut som man girer virkelig. I tillegg skal man kunne koble på en clutchpedal og simulere en motkraft.

Stikkord:

- Girsimulator
- Regulering
- Kraft og posisjon

Tilgjengelig: JA

Prosjekt deltagere og karakter:

Navn	Karakter
Oskar Dramdal Gunbjørnsen	
Martin Vesthaug Ingvaldsen	
Kenneth Bergerud	
Krister Håkenrud	

Dato: 26. Juni 2014

Øyvind Eek-Jensen
Intern Veileder

Karoline Moholth
Intern Sensor

Tom Heistad
Ekstern Sensor

Espen Elvekrok
Ekstern Veileder

Bachelorprosjekt

GCS – Gir &

Clutchsimulator

Prosjektdeltakere:

Kenneth Bergerud, Krister Håkenrud,
Martin Ingvaldsen og Oskar Gunbjørnsen



KONGSBERG
AUTOMOTIVE

Innhold i perm

1. Prosjektplan
2. Iterasjonsdokument
3. Total timeliste
4. Økonomi
5. Kravspesifikasjon
6. Testplan
7. Testspek
8. Oppfølgingsdokumenter
9. Møtereferat
10. Teknologidokument
11. Brukermanual
12. Prosjektrapport

Prosjektplan

GCS – Gir &

Clutchsimulator

Forfatter: Kenneth, Martin og Oskar



Innholdsfortegnelse

1 – Gruppas bakgrunn	3
2 – Ansvarsområder	4
3 – Oppgaven	5
4 – Aktiviteter	6
4 – Aktiviteter forts.	7
5 – Aktiviteter for krav	8
6 – Krav til aktiviteter.....	9
7 – Prosjektmodell: Modifisert Unified Process	10
8 – Fremdriftsplan.....	11
8.1 – Planlegging	12
8.2 – Research	13
8.2 – Research forts.	14
8.3 – Implementasjon	15
8.4 – Slutt.....	16
9 – Timelister	17
10 – Målsetting for oppgaven	18

1 – Gruppas bakgrunn

Gruppen er bestående av Kenneth Bergerud, Krister Håkenrud, Martin Ingvaldsen og Oskar Gunbjørnsen. Krister, Martin og Oskar er elektrostuderter mens Kenneth er datastudent. Vi valgte denne sammensetningen fordi hovedutfordringen i oppgaven er regulering, noe elektrostuderterene har en grunnleggende forståelse for. Det er også en liten datadel i oppgaven noe som gjør at en datastudent er nødvendig.

2 – Ansvarsområder

Kenneth Bergerud	Prosjektleder, økonomi, web/informasjon og LabVIEW
Krister Håkenrud	Gantt-diagram, research av servodriver, beregninger og IndraWorks
Martin Ingvaldsen	Generell dokumentasjon og elektrisk oppkobling
Oskar Gunbjørnsen	Spesifikasjon, test, administrative dokumenter og IndraWorks

Tabell 1: Ansvarsområder

Prosjektplan	Kenneth Bergerud
Kravspesifikasjon	Martin Ingvaldsen
Gantt diagram	Krister Håkenrud
Testplan og testspesifikasjon	Oskar Gunbjørnsen
Teknologiske dokumenter	Kenneth Bergerud
Timelister	Martin Ingvaldsen
Bilder og skisser	Krister Håkenrud

Tabell 2: Ansvarsområder dokumenter

3 – Oppgaven

Kongsberg Automotive lager og utvikler girskift og clutch-aktuering for lastebiler med manuelle girkasser. For å teste disse produktene må en montere de på ulike typer clutch og girkasser. Gruppas oppgave er å lage en simulator som etterligner funksjonen på clutch og girkasser, som kan brukes for å teste ulike varianter av produktene under utvikling.

Simulatoren skal "være" girkassen og clutchen. Simulatoren må tilføre en motkraft til kraften som påføres. Motkraften skal endre seg i forhold til posisjonen simulatoren er i. Posisjonen på simulatoren endres ved å tilføre en kraft som er enten større eller mindre enn motkraften. Det skal også være en database med forskjellige hastigheter. Dette er fordi kraften endrer seg ettersom hastigheten man girer i endrer seg. Det skal da tas med tre forskjellige hastigheter. Sakte, normal og hurtig.

Oppgaven er todelt med en del for clutchsimulering og en del for girskiftsimulering. Tanken er at det ved hjelp av enkle modifikasjoner på simulatorriggen skal kunne skiftes fra clutch til girskift og motsatt. Kongsberg Automotive har girskiftsimulatoren høyest på ønskelista så dette er prioritert.

Hvorfor ønsker de simulatorene?

Kongsberg Automotive ønsker seg disse simulatorene fordi de ønsker et verktøy for å gjøre kundedemonstrasjoner enklere og mer lettfattelig. Det er også et ønske om å sette simulatorene sammen med en lastebilsimulator slik at lastebilsimulatoren får en mer realistisk følelse av Kongsberg Automotives produkter.

Hvorfor har de ikke gjort dette selv?

Denne oppgaven er noe testlaboratoriet til Kongsberg Automotive ikke har tid til å prioritere.

4 – Aktiviteter

Vi har delt opp ting som må gjøres i underoverskrifter, heretter kalt aktiviteter. En full oversikt over aktivitetenes start- og sluttid, ressursestimering og hvem som er ansvarlig, er definert i et vedlagt Gantt-diagram i Microsoft Excel.

Gantt-diagram ligger i mappe 01

Oversiktsliste: aktivitet, ID og kort beskrivelse:

Navn	ID	Beskrivelse av aktivitet
Administrativt, gruppemøter og planlegging	1	Korte gruppemøter, innkalling til møter og lese/skrive mail.
Møte med veileder og KA, møtereferat og oppfølgingsdokument	2	Møter med ekstern eller intern veileder, store gruppemøter og skriving av møtereferat fra disse møtene. Skrive ukentlig oppfølgingsdokument.
Gantt-diagram/ Timelister	3	Opprettelse av timelister og oppdatere disse, lage Gantt-diagram og oppdatere dette.
Kravspek	4	Opprettelse av kravspek, skrive og modifisere denne
Prosjektplan	5	Opprettelse av prosjektplan, skrive og modifisere denne.
Testplan/Testspek	6	Opprettelse av testspek og testplan, planleggetester og skrive resultater i testspek.
1. Presentasjon	7	Klargjøre en PowerPoint presentasjon, fordele manus og klargjøre dokumenter i riktig format.
Research av utstyr, sensorer, I/O behov og prinsippskisser	8	Gjennomgang av eksisterende utstyr og forskning på nye nødvendige alternativer. Gjelder alt av sensorer, driver til servo og motoren. Opprettelse av prinsippskisser.
Måling av girkasse	9	Måle krefter på girkasse for å få en graf man kan ta utgangspunkt i.
Kontakt/Innkjøp leverandører	10	Bestilling av deler og utstyr. Kontakt med leverandører over mail eller telefon.

Tabell 3: Aktiviteter

4 – Aktiviteter forts.

Test del 1 og dokumentasjon.	11	Test av sensorer, servomotor, driver, og riggen generelt og dokumentere dette.
Sensor- og drivervalg (sammen men KA)	12	Avgjøre hvilke sensorer og driver som skal bli brukt til prosjektet.
Elektrisk klargjøring og oppkobling	13	Elektrisk kobling og konstruksjon av utstyr, feks kabler.
Research regulering	14	Finne ut hvordan regulatoren virker/skal lages og regne matematisk på dette.
Labviewprogrammering	15	Programmere i labview og eventuell research rundt dette.
Web-design	16	Lage og vedlikeholde gruppens webside.
2. Presentasjon	17	Klargjøre en PowerPoint presentasjon, fordele manus og klargjøre dokumenter i riktig format.
Sammenstilling og implementasjon	18	Sette sammen systemet, feks koble til sensorene.
Regulering	19	Bygge og implementere regulator. Denne aktiviteten gjelder også for justering av regulatoren.
Test del 2 og dokumentasjon	20	En test av hele systemet i sin helhet. Teste implementasjonen og se til at dette fungerer.
Brukermanual	21	Skrive en brukermanual for hvordan simulatoren fungerer og skal opereres.
Sluttrapport	22	Opprette, sette sammen og skrive en sluttrapport.
3. Presentasjon	23	Klargjøre en PowerPoint presentasjon, fordele manus og klargjøre dokumenter i riktig format.

Tabell 4: Aktiviteter forts.

5 – Aktiviteter for krav

Kravnummer	Aktivitets ID
1.1	8, 12, 14, 19, 20
1.2	8, 12, 14, 19, 20
1.3	11, 14, 19, 20
1.4	11, 14, 19, 20
1.5	11, 20
1.6	14, 19, 20
2.1	8, 12, 14, 19, 20
2.2	8, 12, 14, 19, 20
2.3	11, 14, 19, 20
2.4	15, 20
3.1	15, 20
3.2	15, 20
3.3	15, 20
3.4	15, 20
3.5	15, 20
3.6	15
3.7	8, 10, 11, 12, 14, 20
4.1	8, 9, 11, 12
4.2	8, 9, 11, 12
5.1	8, 12, 13
5.2	8, 12, 13
5.3	13
5.4	8, 13
5.5	8
5.6	8, 13
5.7	8
6.1	12, 15
6.2	12, 13, 18
6.3	10, 12
6.5	12, 13, 18, 19

Tabell 5: Aktiviteter for krav

6 – Krav til aktiviteter

Aktivitets ID	Kravnummer
1	Ingen spesielle krav
2	Ingen spesielle krav
3	Ingen spesielle krav
4	Ingen spesielle krav
5	Ingen spesielle krav
6	Ingen spesielle krav
7	Ingen spesielle krav
8	1.1, 1.2, 1.5, 2.1, 2.2, 2.4, 3.1, 3.4, 3.5, 4.1, 4.2, 5.1, 5.2, 5.4, 5.5, 5.6, 5.7
9	4.1, 4.2
10	Ingen spesielle krav
11	1.1, 1.2, 1.3, 1.4, 2.1, 2.2, 2.3, 2.4, 4.1, 4.2, 5.1, 5.2, 5.3, 5.4, 5.5, 5.6, 5.7
12	3.1, 3.2, 3.3, 3.4, 3.5, 6.1, 6.2, 6.3, 6.5
13	5.1, 5.2, 5.3, 5.4, 5.5, 5.6, 5.7
14	1.1, 1.2, 1.3, 1.4, 1.5, 2.1, 2.2, 2.3, 2.4
15	1.5, 2.4, 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7
16	Ingen spesielle krav
17	Ingen spesielle krav
18	1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 2.1, 2.2, 2.3, 2.4, 4.1, 4.2, 5.1, 5.2, 5.3, 5.4, 5.6, 5.7
19	1.1, 1.2, 1.3, 1.4, 1.5, 2.1, 2.2, 2.3, 2.4
20	Alle
21	Ingen spesielle krav
22	Ingen spesielle krav
23	Ingen spesielle krav

Tabell 6: Krav til aktiviteter

7 – Prosjektmodell: Modifisert Unified Process

I utgangspunktet hadde vi evolusjonær prototyping som vår prosjektmodell, men denne valgte vi å gå bort fra da denne ikke passet vår måte å jobbe på. Den ble vanskelig å følge og oppfylte ikke våre krav til en prosjektmodell. Vi valgte derfor en ny prosjektmodell som er en modifisert versjon av Unified Process. Med modifisert mener vi at vi ikke har valgt å låse oss til at vi må ha ni disipliner, men heller det antall som passer best for vårt prosjekt, altså tretten.

Unified Process er i utgangspunktet laget for software-prosjekter, så vi var nødt til å lage noen modifikasjoner for å tilpasse vårt prosjekt. Siden prosjektet er meget krevende og det er mange spørsmål i luften om ting er gjennomførbart eller ikke, passer denne prosjektmodellen for oss. Vi er nødt til å spikre en sikker og bra løsning på for eksempel servodriver, for at ikke oppdragsgiver bruker masse penger på en driver som ikke fungerer til vårt bruk. Derfor er filosofien med faser og iterasjoner veldig nyttig for vår del.

Våre største milepæler er å gjøre ferdig en fase og begynne på neste, samt presentasjoner. En fase må være avsluttet før neste begynner. Med denne måten å jobbe på sikrer vi at vi ikke tar noen dårlige beslutninger grunnet dårlig tid. Hverken vi eller oppdragsgiver ønsker selvfølgelig å ofre masse ressurser i form av tid og penger på et konsept som ikke er gjennomførbart.

Etter hver iterasjon blir det skrevet et iterasjonsdokument som omhandler hva som ble gjort i den iterasjonen, generell status på prosjektet og hva som skal skje i neste iterasjon. Gruppen har et gruppemøte hver mandag kl. 0900 der ukens oppgaver blir fordelt til hver enkelt. Ukens oppfølgingsdokument skal også være ferdig på dette tidspunktet, og en møteinnkalling til ukens veiledermøte med intern veileder blir sendt, men oppfølgingsdokumentet som vedlegg. Hver uke er det møte med intern veileder og ca. hver andre uke er det møte med ekstern veileder. Dette kan også være oftere etter behov.

8 – Fremdriftsplan

Planlegging	Research		Implementasjon		Slutt
<ul style="list-style-type: none"> • Grunnleggende dokumentasjon - Prosjektplan - Kravspek - Testplan - Testspek - Gantt-diagram 	<ul style="list-style-type: none"> • Research utstyr - Måling av girkasse - Servo - Servodriver - Sensorer - Kommunikasjon - Kompabilitet <ul style="list-style-type: none"> • Testing - Servo og rigg - IndraWorks mot servo - Sensorer - Kommunikasjon - Kompabilitet 	<ul style="list-style-type: none"> • Servodriver - Endelig valg - Bestilling <ul style="list-style-type: none"> • Research-dokument <ul style="list-style-type: none"> • Web-side 	<ul style="list-style-type: none"> • Regulering <ul style="list-style-type: none"> • LabVIEW <ul style="list-style-type: none"> • Klargjøring og oppkobling 	<ul style="list-style-type: none"> • Systemtest - Regulering - LabVIEW - Kommunikasjon 	<ul style="list-style-type: none"> • Brukermanual <ul style="list-style-type: none"> • Sluttrapport <ul style="list-style-type: none"> • 3. presentasjon
Iterasjon 1.1	Iterasjon 2.1	Iterasjon 2.2	Iterasjon 3.1	Iterasjon 3.2	Iterasjon 4.1
1. Presentasjon ↑		2. Presentasjon ↑ Ferdig med research ↑		System ferdig ↑	Innlevering ↑ 3. Presentasjon ↑

Figur 1: Fremdriftsplan

Som man ser i figuren, er prosjektet delt opp i fire forskjellige faser, som igjen er delt opp i seks iterasjoner og 13 disipliner:

- Planlegging
- Research
- Implementasjon
- Slutt

Innenfor hver iterasjon vil vi kjøre ”looper”, hvor vi for eksempel: gjør en research, tester og dokumenterer. Disse ”loopene” vil kjøres helt til iterasjonen er ferdig, og iterasjonsdokumentet skrives.

8.1 – Planlegging

Denne fasen varte frem til 1. Presentasjon som var 12.02.14. Fasen handlet først og fremst om å kartlegge og planlegge prosjektet vårt. Den inneholder Iterasjon 1.1 som igjen inneholder disiplinen Grunnleggende dokumentasjon.

Iterasjon 1.1:

Grunnleggende dokumentasjon:

Denne disiplinen inneholder de dokumentene som var krav fra HBV at skulle være med. I og med at vi valgte å ha presentasjonen så tidlig som vi gjorde, rakk vi ikke å jobbe med mer enn disse dokumentene.

Prosjektplanen inneholder beskrivelse av oppgave prosjektmodellen vår, fremdriftsplan og beskriver hvordan vi skal arbeide med prosjektet.

Kravspek består av alle krav som oppgaven krever, og skal brukes for å kunne løse oppgaven.

Testplanen beskriver hvordan vi skal gå frem i testingen, og hvordan testingen foregår.

Testspek tester om kravene i kravspek er oppfylt

Gantt-diagram er en overordnet tidsplan som viser tidsrommet for de forskjellige aktivitetene i prosjektet. Den inneholder også milepeler.

8.2 – Research

Denne fasen bruker vi for å kartlegge hva slags komponenter, utstyr og andre ting som trengs og hvordan de virker for å få systemet vårt til å fungere. Den vil vare til litt etter 2.

Presentasjon, det vil si frem til påske.

Iterasjon 2.1:

I denne iterasjonen gjør vi grunnarbeidet for å kunne lage systemet.

Research utstyr:

For å finne ut eksakt hvilke komponenter og spesifikasjoner vi trenger for systemet, trenger vi å gå igjennom forskjellige alternativer, for å finne det som passer best.

Måling av girkasse vi har allerede fått en graf for å gi et utgangspunkt til hvordan kraftkurven vil se ut, men den trenger en oppdatering.

Servo, servodriver, sensorer, kommunikasjon og kompatibilitet vi gjorde research for å finne ut av hva slags utstyr vi trengte til systemet.

Testing:

I denne disiplinen tester vi utstyret vi trenger for å se at det holder mål.

Servo og rigg, IndraWorks mot servo, sensorer, kommunikasjon og kompatibilitet vanlig komponenttesting for å se hvordan det fungerer og om det holder mål.

8.2 – Research forts.

Iterasjon 2.2:

Her avslutter vi research-fasen ved å ta valg om hva vi trenger

Servodriver:

Det viktigste valget vi tar i prosjektet, her er det kritisk at vi gjør et riktig valg, fordi dette vil være grunnlaget for styring av systemet, og er det vi må kjøpe inn.

Endelig valg og bestilling konklusjon om hvilken driver vi velger.

Research-dokument:

Samling av alle teknologidokumenter som innebærer research og testing av komponenter.

Web-side:

Krav fra HBV var at alle gruppene skulle ha en web-side for prosjektet, denne blir fremstilt til
2. Presentasjon.

8.3 – Implementasjon

I implementasjons-fasen vil vi sette sammen hele systemet, teste og kalibrere det. Fasen varer frem til litt etter midten av mai, og avsluttes forhåpentligvis med et ferdig produkt.

Iterasjon 3.1:

Første iterasjon i denne fasen, vil omhandle å koble opp hele systemet, og gjøre det klart for systemtest.

Regulering:

Her skal vi regne ut hvordan vi skal kunne styre servoen ut ifra kraftkurven vi målte.

LabVIEW:

LabVIEW-programmeringen bruker vi til å lage programmet som styrer systemet.

Klargjøring og oppkobling:

Her vil vi koble sammen og klargjøre alle komponentene i systemet for det siste systemtesten.

Iterasjon 3.2:

Systemtest:

Avsluttende del i implementasjonen. Her skal hele systemet testes og ferdigstilles.

Regulering her sjekker vi at systemet oppfører seg etter det vi har satt som standard.

LabVIEW MMI (Man Machine Interface) testes, og vi ser om brukeren får styrt systemet riktig.

Kommunikasjon mellom data, system og komponenter.

8.4 – Slutt

I siste fase av prosjektet vil vi avslutte med å skrive en brukermanual, dokumentere alt og forberede siste presentasjon. Denne fasen avsluttes med 3. Presentasjon en gang i starten av juni.

Iterasjon 4.1:

Brukermanual:

Hele systemet skal dokumenteres og ha en brukermanual. Dette er for at kunden skal kunne kjøre systemet på egenhånd.

Sluttrapport:

Her skal vi samle alle dokumenter og skrive en endelig rapport på prosjektet.

3. Presentasjon:

Siste presentasjon skal omhandle en teknisk bit hvor vi beskriver prosjektet og hvordan det har gått, samt en salgsbit hvor vi skal fremme produktet vi har lagd.

9 – Timelister

Timelistene fylles ut hver uke med aktivitetene man har jobbet på, og hvor lenge man har jobbet på hver aktivitet. Forrige ukes timelister skal være ferdig utfylt før kl. 0900 hver mandag. Da settes timene inn i et eget ark der man ser hvor mange timer hver enkelt har på hver aktivitet, hvor mange timer totalt som er blitt brukt på hver aktivitet og hvor mange timer hver person har totalt i prosjektet.

Timeliste Gruppe 3				
1				
2	Navn:			
3	Ukenr:			
4	Ukedag	Aktivitetsnr.	Beskrivelse	Tidsbruk
5				
6	Mandag			
7				
8				
9	Tirsdag			
10				
11				
12	Onsdag			
13				
14				
15	Torsdag			
16				
17				
18	Fredag			
19				
20				
21	Lørdag			
22				
23				
24	Søndag			
25				
26				
27			Sum	0

Figur 2: Timeliste-mal

10 – Målsetting for oppgaven

Hva forventer bedriften å få ut av denne oppgaven?

Kongsberg Automotive ønsker få en fullt fungerende gir og clutchsimulator som kan testes på deres produkter. De ønsker også et lett og representativt grensesnitt for styring av simulatorene.

Hva forventer vi som studenter å få ut av denne oppgaven?

Dette er det første prosjektet vi gjør i skolesammenheng av denne dimensjonen. Vi forventer å få erfaring med å jobbe ikke bare som et team, men et team blandet med flere fagfelt. Vi kjenner ikke hverandre på forhånd så det blir litt ekstra trening å jobbe utenfor sin komfortsone med de man vanligvis gjør prosjekter sammen med. Vi ser på dette som en bonus.

Iterasjonsdokument

GCS – Gir &

Clutchsimulator

Forfatter: Kenneth, Martin og Oskar



Innledning

Vi valgte å dele prosjektet vårt inn i fire faser, hvor to av de inneholder flere iterasjoner. De fire fasene er planlegging, research, implementasjon og slutt. Etter hver av disse iterasjonene har vi skrevet et eget iterasjonsdokument, som er samlet i dette dokumentet. Dokumentene inneholder informasjon om hva vi har gjort i iterasjonen, status på prosjektet, og hva som skjer videre i neste iterasjon. De tre siste iterasjonene går dessverre litt over hverandre, på grunn av tidspress mot slutten av prosjektet.

Iterasjon 1.1

Hva har vi gjort i denne iterasjonen?

Dette har vært en planleggingsfase med mest fokus på dokumenter. Det er kun en disiplin i denne iterasjonen og det er grunnleggende dokumenter. Dokumentene som har blitt opprettet er de grunnleggende dokumentene som kravspesifikasjon, prosjektplan, Gantt-diagram, testplan og testspek. Det har blitt valgt en prosjektmodell, evolusjonær prototyping (senere byttet til en modifisert versjon av Unified Process). Vi har også hatt flere møter med vår oppdragsgiver Kongsberg Automotive for å bli bedre kjent med oppgaven. Vår første milepæl var første presentasjon der vi la frem hva oppgaven går ut på, noen av de viktigste kravene, prosjektplanen og prosjektmodellen. Alle punkt i disiplinen ble oppfylt og vi sier oss ferdig med både første fase og første iterasjon.

Status på prosjekt

Vi føler vi har kommet godt i gang. Vi hadde presentasjon først av alle og hadde dermed ikke så god tid til å få gjort masse før 1. Presentasjon. Vi fikk noen tilbakemeldinger angående prosjektmodellen vår, at denne ikke var samkjørt med Gantt-diagrammet vårt. Dette har blitt sett nærmere på og vi har byttet prosjektmodell til Unified Process for denne passer bedre til vårt prosjekt og måte og jobbe på. Samarbeidet i gruppen fungerer utmerket og vi har en god tone på grupperommet.

Hva skjer i neste iterasjon?

I neste iterasjon som er en del av andre fase er det to disipliner; Research utstyr og testing. Dette innebærer at vi skal teste nåværende rigg og gjøre oss godt kjent med utstyret og se om det klarer å innfri våre krav. Et eksempel på dette er styring av servomotor til vårt bruk, om dette kan gjøres med eksisterende driver og kommunikasjon. Dersom det ikke går skal det gjøres research rundt alternative drivere og kommunikasjonsmoduler. Kraft/posisjonsgrafene til oppdragsgiver er litt utdatert så vi skal også måle kreftene på en girkasse så vi får mer oppdaterte grafer.

Iterasjon 2.1

Hva har vi gjort i denne iterasjonen?

I den første av to iterasjoner i research-fasen har vi hatt fokus på to disipliner. Den første er «Research av utstyr», her har vi tatt for oss muligens de aller viktigste dokumentene i hele prosjektet. Nemlig de hvor vi blant annet detaljerer valg av servo-driver. Om vi skulle vært så uheldige å velge feil servo-driver, som kanskje skulle vise seg og ikke være kompatibel med kommunikasjonsmidlene våre som for eksempel LabVIEW eller kanskje noe av det andre utstyret vårt, ville det vært katastrofalt for vår del og ikke minst KA som finansierer prosjektet.

Den andre disiplinen som er beskrevet, detaljerer blant annet testing av servomotoren på riggen, IndraWorks opp mot servo, sensorer samt responstider og kommunikasjon. Disse testene har også foregått nede på Kongsberg Automotives test lab.

Det eneste punktet i de to disiplinene som ikke ble helt ferdig, var test av lastcelle og dens responstider. Først var vi nødt til å lage en ny kabel som var passet sammen med den man koblet til lastcellen. Så fikk vi ingen respons når vi skulle koble oss til riggen, og plutselig hadde vi ikke mer tid til rådighet.

Status på prosjekt

Annet enn et par unngåelige obligatoriske oppgaver som krevde en del tid fra prosjektet, føler vi fortsatt at vi ligger innenfor tidsskjemaet. Vi må imidlertid fortsatt være bevisst på hvor lang bestillingstiden på driver og alt annet utstyret vi behøver er, siden dette fort kan spise opp masse tid. Vi har i tillegg satt av en uke til både eksamensøving og påskeferie. Siden første presentasjon har vi tatt en skikkelig revidering av Gantt-diagrammet vårt, slik at det nå bedre gjenspeiler det vi allerede har gjort og det vi kommer til å gjøre i ukene framover. Vi har også startet å skrive oppfølgingsdokumenter for hver uke, slik at vi bedre dokumenterer hvor mange timer vi setter av og hvor mange timer vi bruker hver uke på den gitte aktiviteten.

Hva skjer i neste iterasjon?

I neste fase er det tre disipliner.

Servo-driver – I dette dokumentet blir det endelige valget tatt med henhold til servo-driveren og bestilling.

Research-dokument – Her skal alle research-dokumenter være ferdige, altså alle disipliner i iterasjon 2.1 og 2.2.

Web-side – Etter noen utsettelse har vi bestemt at web-siden skal være helt ferdig innen 2. Presentasjon. Grunnen til utsettelsene var rett og slett at Øyvind mente at vi burde prioritere andre ting så lenge den var oppe innen slutten av researchfasen og helst før presentasjon 2.

Iterasjon 2.2

Hva har vi gjort i denne iterasjonen?

I den siste iterasjonen av researchfasen endret vi til en nye prosjektmodell, Modifisert Unified Process. Her innførte vi også iterasjonsdokumentene som skrives etter hver iterasjon. Denne iterasjonen bestod av tre disipliner:

I **Servodriver** har vi foretatt det viktigste valget i prosjektet, valg av servodriver. Etter mye frem og tilbake, nye funn og nesten like mange avslag om mulig kommunikasjon, kom vi endelig i mål med valget. Vi valgte å gå for Sercos uten busskommunikasjon, da dette var kommunikasjonen som passet best med våres system. Vi fikk sendt inn bestillingen før påske, og har blitt lovet utstyret 9. mai.

Research dokumentasjon: Vi har også fått ferdigstilt alle dokumenter som trengs for å kunne starte på den praktiske delen av prosjektet, disse vil til slutt bli samlet i et felles dokument.

Web-side: Vi har fått opprettet en web-side med informasjon om gruppe og prosjektet. Alle disiplinene ble fullført i denne iterasjonen, vi slet en del med valg av servodriver, men kom til slutt i mål. I tillegg fremførte vi 2. Presentasjon, og fikk gode tilbakemeldinger.

Status på prosjekt

I følge tidsplanen vår, ligger vi i samsvar med skjema, men vi ser at det kommer til å bli litt dårlig tid, på grunn av leveringstid på utstyret. Reguleringen og programmering vil også ta en del tid, så vi kommer til å jobbe hardt for å få klart dette til utstyret kommer.

Vi jobber jevnlig med prosjektet og har hyppig dialog sammen med KA. Samarbeidet innad i gruppen fungerer også godt.

Hva skjer i neste iterasjon?

Neste iterasjon er starten på tredje fase: implementasjon. Her vil vi forberede alt til systemtesten. Den består av tre disipliner:

Regulering – Vi må kunne klare å følge den gitte girkurven, så servoen må gi riktig motkraft til en hver tid. Dette er en kompleks operasjon, og i denne disiplinen skal vi få servoen til å kunne følge kurven.

LabVIEW – Dette skal fungere som avlesning av systemet, der vi kan se kraft- og posisjonskurver med mer.

Klargjøring og oppkobling – For å kunne kjøre systemet når resten av utstyret ankommer, må riggen kobles opp riktig, og enkelte ting må testes, som for eksempel enkle deler av regulering, avlesning av sensorer og andre ting.

Iterasjon 3.1

Hva har vi gjort i denne iterasjonen?

Vi har fått lagd et teoretisk grunnlag for reguleringen, men trenger fortsatt en del finjustering for å kunne implementere det i systemet.

I LabVIEW har vi fått laget et program hvor vi kan logge og kalibrere sensorer, og skrive de til graf. Disse signalene brukes til å sammenligne med en csv. fil som leses opp og får det til å vise x(posisjon) og y(kraft) i en graf. Ut ifra grafene om posisjon på sensorene skal vi kunne beregne kraft og sende til MLD. I tillegg sender digitale signaler for å starte og stoppe MLD. Frem til MLD'en ankom, gjorde vi klar de forskjellige sensorene, koblet koblene og kontaktene som trengtes, sjekket signal og fikk kommunikasjon med LabVIEW. Når vi fikk MLD'en 05.05, byttet vi ut den gamle modulen, og koblet den nye opp mot skapet og datamaskinene. I tillegg har vi fått satt på en girspak, slik at vi har noe å kjøre servoen mot.

Status på prosjekt

På grunn av at vi fikk delene vi trengte så sent som vi gjorde, merker vi at det blir litt dårlig tid, men vi håper å klare å bli ferdig med systemet i neste iterasjon. Alle jobber hardt med hver sine arbeidsoppgaver og vi henger bare litt bak tidsskjemaet.

Hva skjer i neste iterasjon?

Siste del av implementasjonsfasen består av selve systemtesten hvor vi skal ferdigstille produktet vårt. Her vil vi teste og implementere alt vi har gjort, og prøve å få systemet til å kjøre etter girgrafene våre. Denne iterasjonen blir litt kritisk med tanke på tid, så vi håper vi rekker å få fullført alt.

Iterasjon 3.2

Hva har vi gjort i denne iterasjonen?

I denne iterasjonen har vi fått kommunikasjon med LabVIEW og IndraWorks, med tanke på sensorer og servo. Vi har fått laget ferdig LabVIEW-programmet, der sensorsignalene vises på graf og logges. Vi laster posisjonssignalet inn i LabVIEW og gjør det om til tilsvarende referansekraft, som sendes videre til IndraWorks. Vi har også mulighet til å kalibrere sensorene. Csv-filene vises i en xy-graf. PID-reguleringen er godt i gang, men vi mangler en liten bit for å få regulert servoen.

Status på prosjekt

Vi har ikke fått utarbeidet hele PID-regulatoren, så vi fikk ikke regulert slik vi ville, men vi håper å få det ferdig til presentasjonen. Ellers er vi i mål med alle andre ting.

Hva skjer i neste iterasjon?

Den siste fasen og iterasjonen har gått litt parallelt med denne på grunn av dårlig tid. Der skal all dokumentasjon ferdigstilles, blant annet en brukermanual for systemet. Den inneholder også siste presentasjon hvor vi skal fremføre prosjektet våres.

Iterasjon 4.1

Hva har vi gjort i denne iterasjonen?

I denne iterasjon har vi skrevet og gjort ferdig all dokumentasjonen. Permen og CDen er blitt levert, og vi skal starte på den endelige presentasjonen.

Status på prosjekt

Prosjektet er innlevert, men vi jobber videre på riggen for å prøve å få til reguleringen før presentasjonen.

Hva skjer i neste iterasjon?

Det er ingen iterasjoner etter denne.

Total timeliste

Aktivitet	Beskrivelse	Kenneth	Krister	Oskar	Martin	Alle
1	Administrativt, gruppemøter og planlegging	48	18	47	43	156
2	Møte med veileder og KA, møtereferat og oppfølgingsdokument	44,5	31,5	49	37	162
3	Gantt diagram/ Timelister	5	24	4	28	61
4	Kravspek	9	5	31	23	68
5	Prosjektplan	41	7	42	37	127
6	Testplan/ Testspek	10	5	47	16	78
7	1. Presentasjon	19	5	19	19	62
8	Research av utstyr, sensorer, I/O behov og prinsippkisser	16	54	17	4	91
9	Måling av girkasse	4	5	2	2	13
10	Kontakt/Innkjøp leverandører		33			33
11	Test del 1 og dokumentasjon	8	3	11	4	26
12	Sensor- og drivvalg (sammen med KA)	5		27	16	48
13	Elektrisk klargjøring og oppkobling	18	5	8	55	85
14	Research regulering		6	7	1	14
15	Labviewprogrammering	122				122
16	Web-design	9				9
17	2. Presentasjon	16	9	19	20	64
18	Sammenstilling og implementasjon		45	7	5	57
19	Regulering	7	161	82	25	275
20	Test del 2 og dokumentasjon		3	60	31	94
21	Brukermanual			16		16
22	Sluttrapport	20		22	35	77
23	3. Presentasjon					0
						0
	Totalt	401	419,5	517	400,5	1738

Fullstendig Gantt-diagram ligger vedlagt på CD i mappe: 01 Gantt Fremdriftsplan og Timelister

Økonomi

Produkt	Pris
Printerkreditt	100
Blekkpatroner	260
Whiteboard-tusj	25
MLD	9321,85
Firmware	3657,75
Software	2656,5
Kaffe fra kantina	100
Perm og CD-mappe	100
Sum	16221,10

Kravspesifikasjon

GCS – Gir &

Clutchsimulator

Forfatter: Kenneth, Martin og Oskar



Innholdsfortegnelse

Innholdsfortegnelse	2
1 – Introduksjon	3
2 – Tekniske krav	4
2.1 – Clutchsimulator	5
2.2 – Girsimulator	6
2.3 – Software	8
2.4 – Sensorer	9
2.5 – Testing og sikkerhet	10
2.6 – Servodriver	11
3 – Referanser	12

1 – Introduksjon

Oppgaven er utlevert fra Kongsberg Automotive som Bacheloroppgave for en gruppe på HBV bestående av: Kenneth Bergerud, Krister Håkenrud, Martin Ingvaldsen og Oskar Gunbjørnsen.

Hensikten med systemet er å simulere motkraften som en girkasse vil gi, som i dette tilfelle er gitt fra en graf. Kraften skal reguleres ved hjelp av input-signaler fra sensorer som måler kraft, posisjon og hastighet. Disse signalene skal prosesseres og gi en output på en aktuator (vi har valgt en servomotor). Det er en forutsetning at alle data skal kunne skrives og leses inn i LabVIEW. Dette nye systemet vil hjelpe Kongsberg Automotive å teste deres aktuatorer for gir og clutch. Systemet skal kunne erstatte girkassen ved simulering. Det skal også lett kunne kobles til deres eksisterende systemer.

2 – Tekniske krav

De tekniske kravene blir beskrevet i dette kapitlet, og er delt opp i henhold til hvilken del av systemet det tilhører.

Det er ønskelig fra KA sin side at girsimulatoren blir prioritert først, så alle krav som er koblet mot denne vil bli oppfylt først. Hvis det i tillegg blir tid til å lage clutchsimulatoren, vil også disse kravene bli oppfylt.

Alle krav er gradert etter hvor viktige de er for oppgaven; A-, B- og C-krav. Alle A-krav er absolutte og skal oppfylles før man går videre i oppgaven. B-kravene er viktige og bør helst oppfylles, men kan sløyfes hvis det er nødvendig. C-krav er ønskelige krav og dermed siste prioritet og skal ikke oppfylles før alle andre krav er oppfylt.

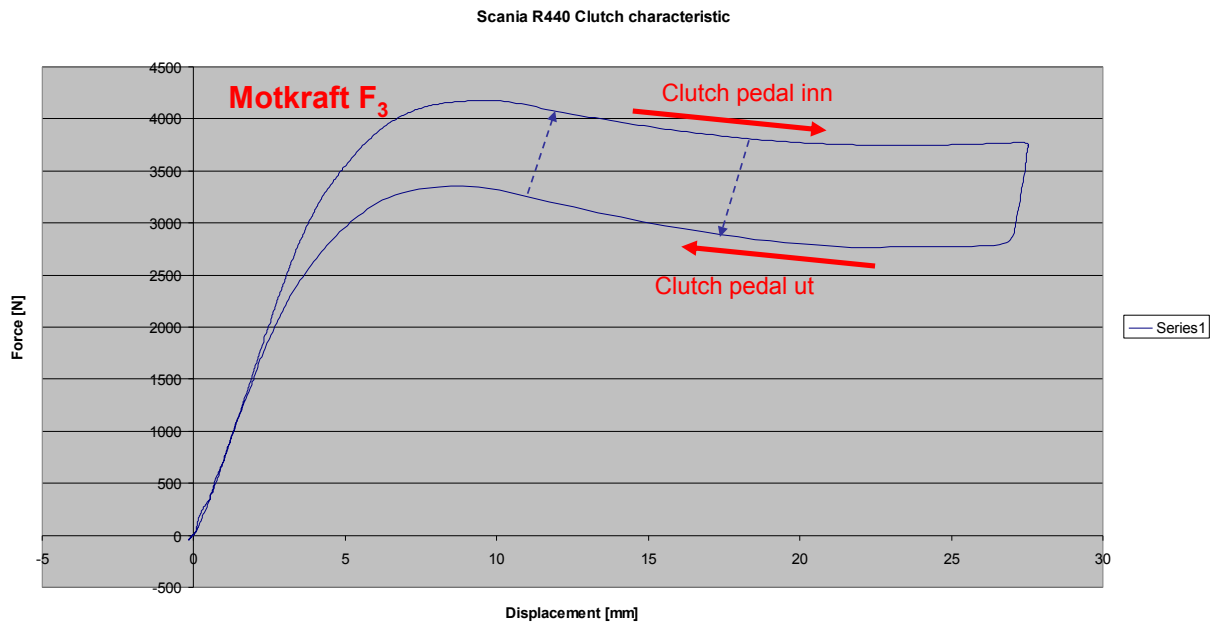
Noen av kravene er ikke endelige og kan bli forandret underveis.

Alle fysiske krav i denne kravspeken er tilpasset en fullskalamodell. Hvis det blir problematisk å oppfylle kraft-kravene, har KA sagt at det er mulighet for å bygge en mindre skalamodell av systemet.

2.1 – Clutchsimulator

Krav nr.	Kravbeskrivelse	Prioritering
1.1	For å kunne simulere den clutchen vi har fått spesifisert, må simulatoren virke med en kraft på maks 5000N.	C
1.2	Posisjonen til armen på servoen skal kunne endres opp til 40mm fra en datum posisjon.	C
1.3	Siden et girskifte tar så liten tid, må armen kunne endre posisjon med en hastighet på 25mm per 0,1 sekunder.	C
1.4	Armen skal være i stand til å gå fra 0 til 100mm og tilbake til 0 på 0,8 sekunder.	C
1.5	Grafene vi får ut av LabVIEW skal følge kurveformen på figur 1.	C
1.6	Tillatt avvik: 0,2mm	C

Tabell 1: Krav til clutchsimulator



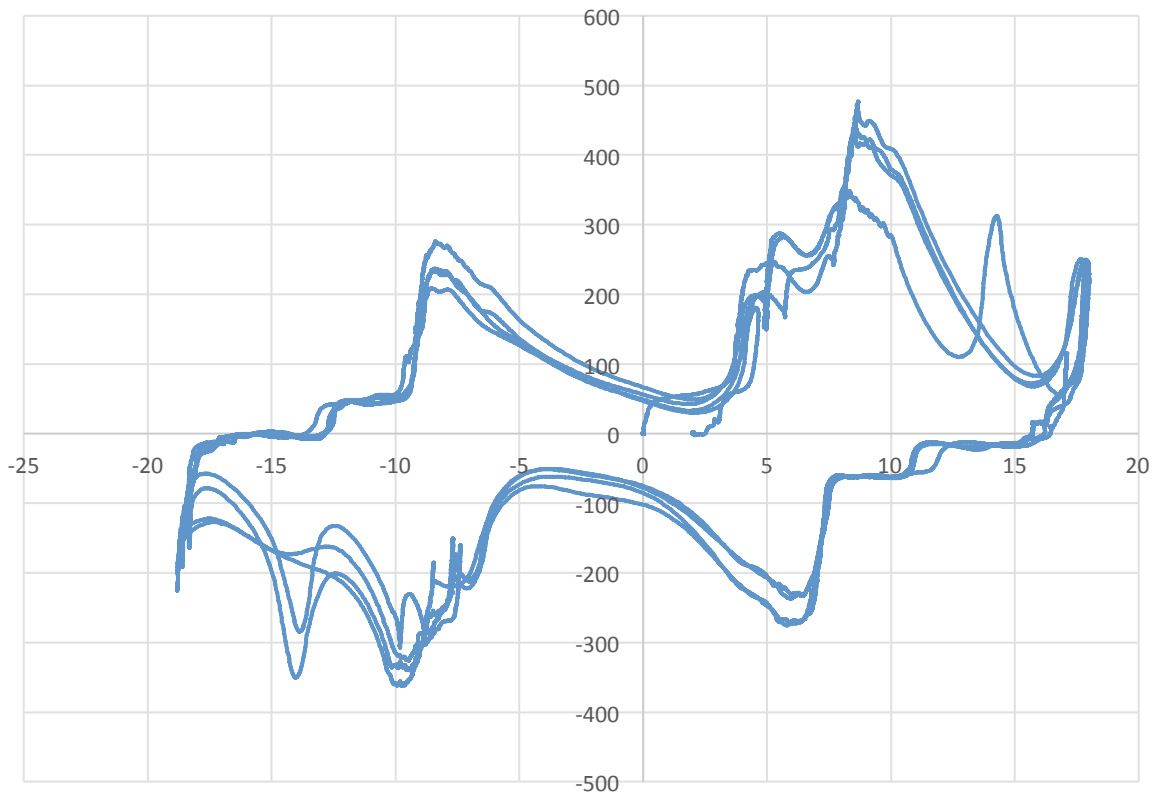
Figur 1: Motkraft-kurve for clutchsimulator

2.2 – Girsimulator

Krav nr.	Kravbeskrivelse	Prioritering
2.1	Girsimulatoren må ha en maks motkraft på 2000N.	A
2.2	Kravet til posisjon på girsimulatoren er -35,8mm til +35,8mm (71,67mm) med 0,2mm toleranse.	A
2.3	Posisjonen skal kunne endres med en fart på 50mm på 0,2 sekunder.	A
2.4	Grafene vi får ut av LabVIEW skal følge kurveformen på figur 2.	A

Tabell 2: Krav til girsimulator

GB force Interval=30



Figur 2: Motkraft-kurve girsimulator

2.3 – Software

Krav nr.	Kravbeskrivelse	Prioritering
3.1	Data fra lastceller skal plottes og vises på en graf.	A
3.2	Grafene skal logges og skrives til en fil.	A
3.3	MMI (Man Machine Interface) skal programmeres i LabVIEW og IndraWorks som et master-slave system.	A
3.4	Programmet skal kunne kalibrere sensorene vi bruker.	C
3.5	Analysere resultater.	C
3.6	For å kunne bedre selge produktet er det ønskelig at vi gjør noe ekstra ut av brukergrensesnittet eller måten man leser data.	C

Tabell 3: Krav til Software

2.4 – Sensorer

Krav nr.	Kravbeskrivelse	Prioritering
4.1	Maksimum krefter som kraft sensoren blir utsatt for er 5000N, men for å ha litt sikkerhetsmargin på cellen velger vi å gå ut i fra 6000N.	C
4.2	Posisjonsgiveren bør ha en nokså god toleranse på avstandsmåling (+-0,2mm).	B

Tabell 4: Krav til sensorer

2.5 – Testing og sikkerhet

Krav nr.	Kravbeskrivelse	Prioritering
5.1	Alt utstyr skal være CE merket.	A
5.2	Bruke den eksisterende riggen hos KA.	A
5.3	Alt elektrisk utstyr skal være i henhold til NEK (Norsk Elektroteknisk Komité).	A
5.4	Skulle det oppstå fare for helse eller utstyr skal det være en nødstoppkrets for å kunne stoppe riggen umiddelbart.	A
5.5	Utstyrets brukstemperatur skal være ca. romtemperatur (20 grader celsius) +/-10 grader celsius.	C
5.6	Det skal brukes et sikkerhetsrelé og en reset bryter slik at simulatorene ikke begynner å gå igjen når nødstopp blir koblet ut.	A
5.7	Anvendt lufttrykk skal ikke overstige 10 bar.	C

Tabell 5: Testing- og sikkerhetskrav

2.6 – Servodriver

Krav nr.	Kravbeskrivelse	Prioritering
6.1	Systemet vårt krever en høy responstid/bitgang.	A
6.2	Kompabilitet med resten av utstyret.	A
6.3	Prisen på utstyret vi bestiller må ikke overstige det KA er villige til å betale.	A
6.5	MLD-en skal brukes som interface for blant annet tre forskjellige parametere. Disse er da potmeteret på girspaken, potmeteret på aktuatoren og lastcellen.	B

Tabell 6: Servodriver krav

3 – Referanser

Kravspesifikasjon. Tilgjengelig fra:

<http://anskaffelser.no/anskaffelser/fagtema/ulike-krav-i-anskaffelsesprosessen/kravspesifikasjon>

[Hentet 20.01.14]

Ny verktøykasse: Veileder for kravspesifikasjon. Tilgjengelig fra:

<http://iktsenteret.no/aktuelt/ny-verktoykasse-veileder-kravspesifikasjon#.UvPVrbQ7GGo>

[Hentet 20.01.14]

Leksjon: 7, Utforming av system/kravspesifikasjon. Tilgjengelig fra:

http://www.aitel.hist.no/fag/_dad/tips/kravspek.html

[Hentet 20.01.14]

Teknisk kravspesifikasjon. Tilgjengelig fra:

<https://innsida.ntnu.no/wiki/-/wiki/Norsk/Teknisk+kravspesifikasjon>

[Hentet 20.01.14]

Testplan

GCS – Gir &

Clutchsimulator

Forfatter: Kenneth, Krister, Martin og Oskar



Innholdsfortegnelse

1 – Introduksjon	3
2 – Funksjoner og egenskaper som skal testes	4
2.1 – Testmetoder/kategorier	4
2.2 – Testing av sensorer	4
2.2 – Testing av sensorer forts.	5
2.3 – Testing av servomotor/aktuatorer	6
2.4 – Testing av LabVIEW opp mot systemet.....	6
2.5 – Måling av girkasse.....	7
2.6 – IndraWorks.....	7
3 – Referanser	8

1 – Introduksjon

I denne testplanen beskriver vi testene som skal verifisere funksjonaliteten til de forskjellige funksjonene og elementene i systemet vårt. Systemet er en girsimulator og skal kunne gi en reel motkraft til det en virkelig lastebil-girkasse ville gitt ved et girskift. Komponentene som skal testes er i hovedsak;

Rigg: Trengs for å holde systemet sammen, kan forhåpentligvis bruke tidligere gruppes testrigg.

Servo/aktuator (evt. hydraulikk): Brukes til å simulere motkraften som girkassen gir. KA har noen tilgjengelige, men må kanskje anskaffe ny.

Sensorer: Vi trenger litt forskjellige sensorer for å kunne måle avstand, hastighet, kraft og litt diverse. Noen av servoene har innebygde sensorer, så vi håper å kunne bruke disse.

LabVIEW: For å kunne styre systemet og lese av data, trenger vi LabVIEW. Vi får disponere en lisens av KA for å kunne bruke programmet.

Test del 1 vil være litt prøving og feiling. Vi skal teste reguleringen og koden opp mot servoen og sensorene på riggen. Vi vil teste ut forskjellige komponenter for å finne ut hva som passer vårt behov.

Test del 2 vil være en slutttesting for systemet. Her vi går igjennom alle elementer, og helheten av systemet vil bli realistisk testet. Resultater vil bli finjustert etter Kongsberg Automotives ønsker.

2 – Funksjoner og egenskaper som skal testes

Når det gjelder testing av utstyr og hardware eller software og dets funksjonalitet eller egenskap, må visse tekniske og administrative krav være på plass. I vårt tilfelle er disse relatert til sensorer, servomotorer/aktuator, LabVIEW og sikkerhet. Kravene vil være detaljerte og beskrevet i kravspesifikasjonen. I denne seksjonen skal vi ta for oss blant annet feil som vi mest sannsynlig møter på under testing, hva som skal testes og spesifisere klassifisering til testen med tanke på grundighet (A-C).

2.1 – Testmetoder/kategorier

Under testingen er det i hovedsak fire typer tester ut ifra to kategorier vi skal gå gjennom. Den første kategorien tar for seg «blackbox»- og «whitebox»-testing. Blackbox-testing går ut på at man gir systemet en input og observerer hva slags effekt dette har på output. I whitebox-testing derimot ser på alle funksjonene i systemet. Den andre kategorien er statisk- og dynamisk-testing, der statisk betyr at systemet ikke kjører, mens dynamisk vil si at det faktisk kjører.

2.2 – Testing av sensorer

Vi har i hovedsak to typer sensorer vi skal teste.

- Kraftsensorer (lastcelle)
- Posisjonsgiver (proximity-sensor)

2.2 – Testing av sensorer forts.

Testingen av disse to typene sensorer vil hovedsakelig omhandle kalibrering, der vi sammenlikner sensorene mot en normal eller en annen sensor med større nøyaktighet for da å bestemme avviket fra korrekt verdi.

Den maksimale belastningen som lastcellen vil bli utsatt for vil være 5000N eller ca. 509,85kg, men siden sensoren vi kommer til å bruke vil ha en sikkerhets margin på minst 1000N, er det liten sannsynlighet for belastningsskader på cellen. Det er fortsatt relativt store krefter involvert, og derfor har vi satt opp krav om en nødstoppkrets for å stoppe riggen umiddelbart samt et sikkerhetsrelé slik at simulatorene ikke starter opp igjen med en gang man kobler ut nødstoppbryteren.

Det at last cellen er kalibrert riktig og gir riktig informasjon om hvor mye kraft som påtrykkes til en hver tid er en kritisk funksjon å få på plass, som da tilsvarer en «A-gradering». Dette er også for å unngå at testingen blir mye dyrere enn forventet. Testingen i del 1 av prosjektet vil primært være statisk siden hele systemet ikke kommer til å være klart for det enda. Del to derimot vil være mer dynamisk, dette gjelder for alle sensorer.

Første test av lastcelle og responstider er planlagt til fredag den 5. mars. Vi kommer da til å koble lastcellen til en av inngangene på rigg R-144 via en 15-pin «banan-plugg» på siden av skapet.

Posisjongiveren må kunne gi data til programmet om et utslag på maks 100mm, som da er avstanden fra et gir til et annet. Vi skal kun ta for oss giring i en rett linje, på en Scania lastebil-girkasse vil dette være mellom 2. og 3. gir. Toleransen har vi ikke blitt enige om enda, men vi tenker at det kan være i nærheten av ± 5 mm. Hvor grundig vi tester posisjongiveren svarer til en «B-gradering».

Potensielle feil vil mest sannsynligvis være feil på kalibreringen. Det å oppnå en nøyaktighet på 1:1 er ikke nødvendigvis noen smal sak.

Den første testen vi har planlagt er den 13. februar, der vi skal plote en ny kraftkurve til girsimulatoren. Grunnen til dette er at KA mener grafen de allerede har er noe utdatert.

2.3 – Testing av servomotor

Den elektriske servomotoren vi skal benytte oss av må kunne komme med en kraft på 5000N.

Vi har snakket med KA og de mener den allerede eksisterende motoren de har liggende hos seg kommer til å ha nok kraft, men dette finner vi ut sikkert under testing. Dette vil jo mest sannsynlig være den mest forekomne feilen. Selv om dette kanskje er en triviell test, er det et must, og bør derfor ha minst en B-gradering.

Den første testen av servomotoren er planlagt er den 24. februar, der vi skal avgjøre om kravet til hastighet og posisjon er oppfylt. Vi skal også avgjøre overføringen fra antall omdreininger til mm.

2.4 – Testing av LabVIEW opp mot systemet

LabVIEW er det grafiske programmeringsspråket vi skal benytte oss av i løpet av dette prosjektet. Her henter vi inn alle Excel-filer med data for posisjon og tilsvarende kraft, deler den kraften på 100 i og med at IndraWorks bare kan ta imot signaler mellom ± 10 , og deretter sendes det da til IndraWorks. Her i test del to vil som sagt dynamisk testing dominere, sammen med blackbox-tesing.

Hovedpoenget med denne testen er å forsikre oss om at programmet skriver de riktige kraft-, posisjon- og hastighets-grafene til fil. Dette er både en kritisk og kompleks funksjon på den måten at det er viktig i drift, ofte brukt og inneholder programmeringsteknisk risiko (systemet kan krasje). Denne testen opp mot IndraWorks og reguleringen har vi anslått at krever mest tid og grundighet av alle sammen, nettopp fordi oppgaven går veldig mye ut på reguleringsteknikk. Feil vi forventer under denne testen vil antageligvis ha med reguleringen å gjøre.

2.5 – Måling av girkasse

Torsdag den 13. februar har vi planlagt en test for å måle en ny kraftkurve. Oppdragsgiver mistenker at den allerede eksisterende kurven de har målt for en stund siden er noe utdatert. Dette vil bli utført på rigg R-128 som er utstyrt med en Scania girkasse. Siden vi ikke skal teste noe av utstyret vi skal bruke, hører det hverken til test del 1- eller -2, men heller en egen aktivitet.

Å ha en oppdatert kraftkurve er essensielt, selv om vi forventer at den nye kurven er nokså lik som den gamle skal testen fortsatt ha en B-gradering med tanke på grundighet.

Noen forkommende feil kommer nok til å være spikes på kurven som vi enten må klippe ut i paint eller ta gjennomsnittet av.

2.6 – IndraWorks

Tirsdag den 6. mai er det planlagt oppstart av kommunikasjon mellom MLD og PC med IndraWorks, samt regulering og kjøring. IndraWorks er standard programmet som brukes som brukergrensesnitt for Bosch Rexroth sine MLD-enheter. Vi lånte PCen til vår eksterne veileder Espen Elvekrok, siden det originale utstyret benyttet IndraWorks DS. Det vil vise seg hvor representativt dette programmet er i forhold til den versjonen vi skal ta i bruk nå.

Som sagt krever dette og LabVIEW grundig testing, siden det i grove trekk er hoveddelen av oppgaven. Forekommende feil her vil ganske sikkert være enten problemer med kommunikasjon og feilmeldinger relatert til dette, eller vanskeligheter rundt IP-innstillinger om det enten måtte være fra selve programmet eller på displayet til MLDen.

3 – Referanser

Test plan. Tilgjengelig fra:

http://en.wikipedia.org/wiki/Test_plan

[Hentet 5.2.2014]

Schaefer, Hans. (2000) *MAL FOR TESTPLANER*. Tilgjengelig fra:

http://www.google.no/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0CCgQFjAA&url=http%3A%2F%2Fwww.softwaretesting.no%2Ftesting%2Ftestplanmal.rtf&ei=QLr3Uu2ND6PF7Abv5YGICg&usg=AFQjCNGPi9W3wKxLlkr0M8PuYTPpmLBhpA&sig2=Gn40zgk5wRJyTlzvR1y_Ow&bvm=bv.60983673,d.bGE

[Hentet 5.2.2014]

Testspek

GCS – Gir &

Clutchsimulator

Forfatter: Kenneth, Martin og Oskar



Innholdsfortegnelse

1 – Clutchsimulator	3
2 – Girsimulator	4
3 – Software.....	5
4 – Sensorer	5
5 – Testing og sikkerhet	6
6 – Servodriver.....	7

1 – Clutchsimulator

Kravbeskrivelse	Nr.	Ja	Nei
Er kravet om at simulatoren skal virke med en kraft på 5000N innfridd? Utgår på grunn av prioritet av girsimulator	1.1		
Er kravet om at posisjonen til armen skal kunne endres opp til 40mm innfridd? Utgår på grunn av prioritet av girsimulator	1.2		
Er kravet om at armen skal kunne endre posisjon med en hastighet på 25mm per 0,1 sekund innfridd? Utgår på grunn av prioritet av girsimulator	1.3		
Er kravet om at armen skal være i stand til å gå fra 0 til 25mm og tilbake til 0 på 0,2 sekunder innfridd? Utgår på grunn av prioritet av girsimulator	1.4		
Er kravet om at grafene i LabVIEW skal følge kurveformen på figur 1 i kravspesifikasjonen innfridd? Utgår på grunn av prioritet av girsimulator	1.5		
Er kravet om tillat avvik på 0,2mm med mer innfridd? Utgår på grunn av prioritet av girsimulator	1.6		

Tabell 1: Testspek clutchsimulator

2 – Girsimulator

Kravbeskrivelse	Nr.	Ja	Nei
Er kravet om maks motkraft på 2000N innfridd? Vi fikk bekreftet av KA at servomotoren deres var fullt i stand til å oppnå denne kraften.	2.1		
Er kravet om posisjon fra -35,8mm til +35,8mm innfridd samt toleranse innfridd? Når vi først testet servomotorens overføring i henhold til vandring på riggen per rotasjon den 24. februar, målte vi med skyvelære og så at dette kravet var oppfylt.	2.2		
Er kravet om at posisjonen skal kunne endres med en fart på 50mm på 0,2 sekunder innfridd? Under den samme testen, målte vi også tiden det tok for armen å vandre fra en datum posisjon, til +100mm og tilbake. Resultatet ble på ca. 0,8 sekunder, som er godkjent.	2.3		
Er kravet om at grafene vi får ut av LabVIEW skal følge kurveformen på figur 2 i kravspesifikasjonen innfridd?	2.4		

Tabell 2: Testspek girsimulator

3 – Software

Kravbeskrivelse	Nr.	Ja	Nei
Er kravet om at data fra lastceller skal plottes inn og vises på en graf innfridd? Dette kravet ble oppfylt når vi fant limit-values i grader for sylindren.	3.1		
Er kravene om at grafene skal logges og skrives til en fil innfridd? Grafene skrives og logges i LabVIEW	3.2		
Er kravet om at MMI (Man Machine Interface) skal programmeres i LabVIEW og IndraWorks innfridd? IndraWorks ble bestilt sammen med MLD og firmware den 11. april, og LabVIEW lisens har KA liggende hos seg. Alt som gjelder nå er å implementere det i systemet vårt som et master-slave oppsett.	3.3		
Er kravet om at programmet skal kalibrere sensorene vi bruker innfridd? Kalibrering av sensorer skjer i LabVIEW	3.4		
Er kravet om analysering av data innfridd? Nei, dette er blitt nedprioritert	3.5		
Er kravet om at man skal gjøre noe ekstra ut av brukergrensesnittet eller måten man leser data innfridd? Nei, dette er blitt nedprioritert	3.6		

Tabell 3: Testspek software

4 – Sensorer

Kravbeskrivelse	Nr.	Ja	Nei
Er kravet om sikkerhetsmargin på last cellen innfridd? Vi ser dette enkelt ved å lese av på den gitte sensoren sin spesifikasjon, vi kommer ikke til å bevege oss i nærheten av denne marginen.	4.1		
Er kravet om toleranse på posisjonsgiveren innfridd? Ja, posisjonsgiverne er nøyaktige nok	4.2		

Tabell 4: Testspek sensorer

5 – Testing og sikkerhet

Kravbeskrivelse	Nr.	Ja	Nei
Er krav om at alt utstyr skal være CE merket innfridd? Ja, alt utstyret er godkjent	5.1		
Er kravet om at det blir brukt den allerede eksisterende riggen hos KA innfridd? Ja, vi fikk brukt riggen	5.2		
Er kravet om at alt elektrisk utstyr er i henhold til NEK? Ja, alt utstyr er godkjent og koblet etter regelverk	5.3		
Er kravet om nødstopp innfridd? Ja, riggen var allerede utstyrt med en nødstopp-bryter	5.4		
Er kravet om temperatur på 20 grader celsius +-10 grader celsius innfridd? Ja, systemet står på laben, og der er det alltid romtemperatur	5.5		
Er kravet om sikkerhetsrelé innfridd? Ja, riggen var allerede utstyrt med dette	5.6		
Er kravet om at lufttrykket ikke skal overstige 10 bar innfridd? Lufttrykk brukte vi kun under måling av ny kraftkurve, og var vi langt unna denne grensen. Kravet er da innfridd.	5.7		

Tabell 5: Testspek testing og sikkerhet

6 – Servodriver

Kravbeskrivelse	Nr.	Ja	Nei
<p>Er kravet om høy responstid/bitgang innfridd?</p> <p>Dette ble undersøkt grundig når vi tok valget om hvilken løsning vi skulle gå for. Vi valgte da Sercos uten busskommunikasjon etter mye research og samtale med Rexroth, og kan derfor si at dette kravet er godkjent.</p>	6.1		
<p>Er kravet om kompatibilitet med resten av utstyret innfridd?</p> <p>Bekreftelse på nettopp dette fikk vi også under research-fasen etter samtale med Rexroth.</p>	6.2		
<p>Er kravet om pris innfridd?</p> <p>Prisen totalt ble på rett over 15 000 kr, noe som KA var villige til å gå for. De originale estimatene var på godt over 20 000 kr, derfor var dette en positiv overraskelse.</p>	6.3		
<p>Er kravet om at MLD-en skal brukes som interface for posisjongivere og lastcelle innfridd?</p> <p>Pluss og minus på lastcellen kobles til inngang 4 og 5 på X32 henholdsvis. Mens pluss og jord på begge posisjons-givere kobles til 2, 4 og 1 henholdsvis da på MA1.</p>	6.5		

Tabell 6: Testspek servodriver

Oppfølgingsdokumenter

GCS – Gir &

Clutchsimulator



KONGSBERG
AUTOMOTIVE

Uke: 7

Hva skulle vi gjøre?

Legge inn aktiviteter i gantt-diagram.

Gjøre ferdig kravspekken.

Starte på og bli ferdig med testplan og testspek.

Bli ferdig med prosjektplanen.

Velge en prosjektmodell.

Hva har vi gjort?

Aktiviteter inn i Gantt, ryddet og pyntet.

Rettet opp og fullførte kravspekken.

Startet å skrive på testplan.

Lagde prosjektmodell-figur.

Skrev om prosjektmodellen vi valgte, altså evolusjonær prototyping og ble ferdig med prosjektplanen.

Kommentar: Ble ferdig med alle punkter.

Uke 8

Hva skal vi gjøre?

Gjøre ferdig første presentasjon til onsdag den 12. februar.

Måle krefter på riggen hos KA for å få en oppdatert versjon av kraftkurvene.

Starte å skrive oppfølgingsdokumenter.

Oppdatere flere dokumenter.

Starte på teknologidokumenter.

Uke: 8

Hva skulle vi gjøre?

Gjøre ferdig første presentasjon til onsdag den 12. februar.

Måle krefter på riggen hos KA for å få en oppdatert versjon av kraftkurvene.

Starte å skrive oppfølgingsdokumenter.

Oppdatere flere dokumenter.

Starte på teknologidokumenter.

Hva har vi gjort?

Fullført første presentasjon.

Målt, tatt bilder og lastet ned grafer med kraftkurver.

Startet med oppfølgingsdokumenter.

Oppdatert aktivitets liste og prosjektplan.

Startet med teknologidokumenter.

Ringt produsent av servomotoren og gjort research rundt denne.

Kommentar: Ble ferdig med alle punkter.

Uke 9

Hva skal vi gjøre?

Web-side skal opp.

Rydde opp/reinskrive i prosjektmodell/plan og kravspek

Fortsette på teknologidokumentet til testen vi utførte den 13.02.

Starte med LabVIEW programmering.

Ringt produsent av servomotoren og gjort research rundt denne.

Uke: 9

Hva skulle vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
16	Web-side skal opp	
4/5	Rydder opp/reinskrive i prosjektmodell/plan og kravspek	
11	Fortsette på teknologidokumentet til testen vi utførte den 13.02	
15	Research rundt LabVIEW programmering	
10	Ringt produsent av servomotoren og gjøre research rundt denne	

Hva har vi gjort?

Aktivitet	Beskrivelse	Tid brukt
15	Startet med research rundt LabVIEW-programmering	
8	Research rundt servodriver	
11	Fortsette på teknologidokumentet til testen vi gjorde	
5	Ryddet opp i prosjektplanen.	
10	Ringt produsent av servomotoren og gjort research rundt denne	

Kommentar: Rakk ikke å starte med web-design

Uke: 10

Hva skal vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
16	Web-side skal opp	4 timer
15	Fortsette med research til LabVIEW-programmering	10 timer
10	Prate mer med produsent og gjøre mer research	30 timer
8	Begynne å velge sensorer	20 timer
8	Bli kjent med utstyr og kikke på koblingskjemaer	20 timer
3/5	Prosjektplan og timelister overhaling	50 timer

Uke: 10

Hva skulle vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
16	Web-side skal opp	4 timer
15	Fortsette med LabVIEW research	10 timer
10	Prate mer med produsent og gjøre mer research	30 timer
8	Velge sensorer	20 timer
8	Bli kjent med utstyr og kikke på koblingskjemaer	20 timer
3, 5	Prosjektplan og timelister overhaling	50 timer

Hva har vi gjort?

Aktivitet	Beskrivelse	Estimert tid
1	Gruppemøte planlegging	21
10	Prate mer med produsent og gjøre mer research	5
3	Jobbet med nye timelister	16
8	Bli kjent med utstyr, koblingskjemaer og manualer	2
2	Forberedelse til møte med KA	10
11	Test av lastcelle	11
5	Prosjektplan	4
10	Kommunikasjon med Rexroth	8

Kommentar: Vi fikk beskjed av Øyvind at å få web-siden opp, trengte ikke å prioriteres før 2. presentasjon. Timene vi forventet å bruke på hver av aktivitetene ble estimert ganske grovt, i tillegg ble Krister syk.

Uke: 11

Hva skal vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
8	Gjennomgang av servodriver og research rundt dette	20
3	Endring av gantt-diagram og timelister	10
10	Samtale med Rexroth	5
6	Testplan/testspek må oppdateres	10
5	Prosjektplan	10
11	Test del 1 nede hos KA	20
2	Veileder møte og møtereferat	10
1	Gruppemøte og planlegging	20

Uke: 11

Hva skulle vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
8	Gjennomgang av servodriver og research rundt dette	20
3	Endring av gantt-diagram og timelister	10
10	Samtale med Rexroth	5
6	Testplan/testspek må oppdateres	10
5	Prosjektplan	10
11	Test del 1 nede hos KA	20
2	Veiledermøte og møtereferat	10
1	Gruppemøte og planlegging	20
		Sum: 105

Hva har vi gjort?

Aktivitet	Beskrivelse	Tid brukt
1	Gruppemøte og planlegging	25
2	Veiledermøte og møtereferat	9
3	Endring av gantt-diagram og timelister	20
5	Prosjektplan	1
6	Lagt til flere seksjoner og skrevet på testplan	4
8	Gjennomgang av servodriver og research rundt dette	12
10	Samtale med Rexroth	1
11	Skrevet på rapport til test av lastcelle og responstider	2
		Sum: 74

Kommentar: Vi fikk ikke tid til å fullføre testen av lastcelle og responstider, og heller ikke skrevet like mye på prosjektplanen som vi hadde ønsket.

Uke: 12

Hva skal vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
1	Gruppemøte og planlegging	20
2	Veiledermøte og møtereferat	10
3	Endring av gantt-diagram og timellister	5
16	Web-side	10
6	Testspek	2
11	Teknologidokument for servodriver	5
15	LabVIEW kurs	3
14	Research regulering	35
		Sum: 90

Uke: 12

Hva skulle vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
1	Gruppemøte og planlegging	20
2	Veiledermøte og møtereferat	10
3	Endring av gantt-diagram og timellister	5
16	Web-side	10
6	Testspek	2
12	Teknologidokument for servodriver	5
15	LabVIEW kurs	3
14	Research regulering	35
		Sum: 90

Hva har vi gjort?

Aktivitet	Beskrivelse	Tid brukt
1	Gruppemøte og planlegging	9
2	Veiledermøte og møtereferat	6
5	Prosjektplan	5
6	Testspek	2
8	Research av utstyr, sensorer, I/O behov og prinsippsskisser	5
9	Redigering av girkassemålinger	3
10	Samtale med Rexroth	6
12	Gjennomgang av servo-driver, teknologidokument og pugh-matrise	39
14	Research regulering	4
15	LabVIEW kurs	4
16	Web-side	4,5
		Sum: 87,5

Kommentar: På aktivitet 12 og 14 ble det byttet om på timeantallet på grunn av prioritering, ellers ble det ikke brukt like mye tid på det administrative og gruppemøter som vi hadde trodd.

Uke: 13

Hva skal vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
1	Gruppemøte og planlegging.	20
2	Veiledermøte og møtereferat.	10
5	Lage ny prosjektmodell.	10
8	Research av utstyr, sensorer, I/O behov og prinsippskisser	5
12	Research rundt driver kommunikasjon må bli ferdig	40
13	Elektrisk oppkobling	5
14	Research regulering	5
15	LabVIEW programmering	5
16	Webside	5
		Sum: 105

Uke: 13

Hva skulle vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
1	Gruppemøte og planlegging.	20
2	Veiledermøte og møtereferat.	10
5	Lage ny prosjektmodell.	10
8	Research av utstyr, sensorer, I/O behov og prinsippsskisser	5
12	Research rundt driver kommunikasjon må bli ferdig	40
13	Elektrisk oppkobling	5
14	Research regulering	5
15	LabVIEW programmering	5
16	Webside	5
		Sum: 105

Hva har vi gjort?

Aktivitet	Beskrivelse	Tid brukt
1	Gruppemøte og planlegging	21
2	Veiledermøte og møtereferat	4
3	Gantt	3
5	Lage ny prosjektmodell	47
8	Research av utstyr, sensorer, I/O behov og prinsippsskisser	15
10	Samtale med Rexroth	5
16	Webside	3
		Sum: 98

Kommentar: Vi undervurderte kraftig hvor lang tid det faktisk kom til å ta og endre på hele prosjektmodellen. Vi hadde antageligvis brukt enda mere tid om ikke Øyvind hadde fortalt oss at vi burde bruke tiden vår på andre ting.

Uke: 14

Hva skal vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
1	Gruppemøte og planlegging	20
2	Veiledermøte og møtereferat	10
10	Samtale med Rexroth	5
11	Dokumentasjon til test del 1	10
12	Drivervalg	5
14	Research regulering	10
17	Presentasjon 2	40
		Sum: 100

Uke: 14

Hva skulle vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
1	Gruppemøte og planlegging	20
2	Veiledermøte og møtereferat	10
10	Samtale med Rexroth	5
11	Dokumentasjon til test del 1	10
12	Drivervalg	5
14	Research regulering	10
17	Presentasjon 2	40
		Sum: 100

Hva har vi gjort?

Aktivitet	Beskrivelse	Tid brukt
1	Gruppemøte og planlegging	9
2	Veiledermøte og møtereferat	12
5	Ferdigstilling av prosjektplan og iterasjonsdokument	14
8	Prinsippkisser	19
10	Samtale med Rexroth	6
12	Drivervalg	3
14	Research regulering	1
16	Webside	1
17	Gjennomføring av 2. presentasjon	64
		Sum: 129

Kommentar: Vi måtte fokusere på 2. presentasjon, derfor rakk vi ikke å jobbe like mye med de andre aktivitetene.

Uke: 15

Hva skal vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
		Sum:

Kommentar: På grunn av eksamen, har vi ikke satt av noe tid til prosjektet denne uka.

Uke: 15

Hva skulle vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
		Sum:

Hva har vi gjort?

Aktivitet	Beskrivelse	Tid brukt
1	Administrativt	2
2	Møte med veileder	2
10	Kommunikasjon med Bosch Rexroth angående tilbud	1
		Sum: 5

Kommentar: Kenneth og Krister fikk litt tid til administrativt arbeid samt møte med veileder likevel.

Uke: 16

Hva skal vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
		Sum:

Kommentar: I påsken har vi ikke satt av noe tid til prosjektet.

Uke: 16

Hva skulle vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
		Sum:

Hva har vi gjort?

Aktivitet	Beskrivelse	Tid brukt
1	Administrativt	4
12	Oppdaterte valg av servodriver dokument	1
		Sum: 5

Uke: 17

Hva skal vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
1	Gruppemøter og planlegging	20
2	Møte med veileder og med KA	10
3	Timelister og gantt diagram	5
5	Iterasjonsdokument	10
6	Testspek	15
10	Kontakt med leverandør	5
13	Elektrisk klargjøring og oppkobling	10
15	Labviewprogrammering	15
18	Sammenstilling og implementasjon	5
19	Regulering	10
20	Test del 2 og dokumentasjon	5
		Sum: 110

Uke: 17

Hva skulle vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
1	Gruppemøter og planlegging	20
2	Møte med veileder og med KA	10
3	Timelister og gantt diagram	5
5	Iterasjonsdokument	10
6	Testspek	15
10	Kontakt med leverandør	5
13	Elektrisk klargjøring og oppkobling	10
15	Labview-programmering	15
18	Sammenstilling og implementasjon	5
19	Regulering	10
20	Test del 2 og dokumentasjon	5
		Sum: 110

Hva har vi gjort?

Aktivitet	Beskrivelse	Tid brukt
1	Gruppemøter og planlegging	19
2	Møte med veileder og med KA	17
3	Timelister og gantt diagram	2
4	Kravspek	2
5	Iterasjonsdokument	5
6	Testspek og testplan	8
8	Prinsippskisser	4
10	Kontakt med leverandør	1
12	Oppdatering av servodriver dokument	5
13	Elektrisk klargjøring og oppkobling	1
15	Labview-programmering	3
19	Regulering	29
		Sum: 96

Uke: 18

Hva skal vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
1	Gruppemøter og planlegging	20
2	Møte med veileder og med KA	15
3	Timelister og gantt diagram	5
4	Kravspek	10
5	Prosjektplan og iterasjonsdokument	5
6	Testspek og testplan	10
10	Kontakt med leverandør	5
13	Elektrisk klargjøring og oppkobling	15
15	Labview-programmering	10
18	Sammenstilling og implementasjon	10
19	Regulering	20
20	Test del 2 og implementasjon	5
		Sum: 130

Kommentar: Det ble stokket litt om på hvor mye tid vi brukte på hver av de forskjellige aktivitetene denne uken, og sammenstillingen rakk vi ikke gjøre noe med.

Uke: 18

Hva skulle vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
1	Gruppemøter og planlegging	20
2	Møte med veileder og med KA	15
3	Timelister og gantt diagram	5
4	Kravspek	10
5	Prosjektplan og iterasjonsdokument	5
6	Testspek og testplan	10
10	Kontakt med leverandør	5
13	Elektrisk klargjøring og oppkobling	15
15	Labview-programmering	10
18	Sammenstilling og implementasjon	10
19	Regulering	20
20	Test del 2 og implementasjon	5
		Sum: 130

Hva har vi gjort?

Aktivitet	Beskrivelse	Tid brukt
1	Gruppemøter og planlegging	16
2	Møte med veileder og med KA	6
3	Timelister og gantt diagram	1
5	Iterasjonsdokument	6
6	Testspek og testplan	10
8	Prinsippskisser	1
13	Elektrisk klargjøring og oppkobling	10
15	LabVIEW	17
19	Regulering	33,5
20	Test del 2 og dokumentasjon	3
		Sum: 103,5

Kommentar: Det ble ikke tid til å oppdatere kravspekken og heller ikke sammenstilling.

Uke: 19

Hva skal vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
1	Gruppemøter og planlegging	20
2	Møte med veileder og med KA	10
13	Elektrisk klargjøring og oppkobling	20
15	Labview-programmering	20
18	Sammenstilling og implementasjon	10
19	Regulering	20
20	Test del 2 og implementasjon	10
		Sum: 110

Uke: 19

Hva skulle vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
1	Gruppemøter og planlegging	20
2	Møte med veileder og med KA	10
13	Elektrisk klargjøring og oppkobling	20
15	Labview-programmering	20
18	Sammenstilling og implementasjon	10
19	Regulering	20
20	Test del 2 og implementasjon	10
		Sum: 110

Hva har vi gjort?

Aktivitet	Beskrivelse	Tid brukt
1	Gruppemøter og planlegging	9
2	Møte med veileder og med KA	5
3	Timelister	1
13	Elektrisk klargjøring og oppkobling	37
15	LabVIEW	27
18	Sammenstilling og implementasjon	8
19	Regulering	72
20	Test del 2 og dokumentasjon	10
		Sum: 169

Kommentar: Det ble liten tid til gruppemøter og planlegging denne uken. Isteden brukte vi hele uken på oppkobling og regulering.

Uke: 20

Hva skal vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
1	Gruppemøter og planlegging	10
2	Møte med veileder og med KA	10
13	Elektrisk klargjøring og oppkobling	10
15	Labview-programmering	20
18	Sammenstilling og implementasjon	20
19	Regulering	50
20	Test del 2 og implementasjon	20
		Sum: 140

Uke: 20

Hva skulle vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
1	Gruppemøter og planlegging	10
2	Møte med veileder og med KA	10
13	Elektrisk klargjøring og oppkobling	10
15	Labview-programmering	20
18	Sammenstilling og implementasjon	20
19	Regulering	50
20	Test del 2 og implementasjon	20
		Sum: 140

Hva har vi gjort?

Aktivitet	Beskrivelse	Tid brukt
1	Gruppemøter og planlegging	2
2	Møte med veileder og med KA	7
13	Elektrisk klargjøring og oppkobling	35
15	LabVIEW	23,5
18	Sammenstilling og implementasjon	43
19	Regulering	41
20	Test del 2 og dokumentasjon	17
22	Sluttrapport	1
		Sum: 169,5

Kommentar: Vi fikk nesten ikke tid til gruppemøter og planlegging siden det gikk mer tid til elektrisk klargjøring og oppkobling samt sammenstilling og implementasjon enn først trodde.

Uke: 21

Hva skal vi gjøre?

Aktivitet	Beskrivelse	Estimert tid
1	Gruppemøter og planlegging	5
2	Møte med veileder og med KA	5
15	Labview-programmering	40
18	Sammenstilling og implementasjon	20
19	Regulering	50
20	Test del 2 og implementasjon	50
21	Brukermanual	20
22	Sluttrapport	20
		Sum: 210

Møtereferat for:
Veiledermøter
Møter med KA
Gruppemøte
GCS – Gir &
Clutchsimulator



Møtetype: Veiledningsmøte med intern veileder	Sted: HBV D230
Dato: 30.01.14	Neste møte: 06.02.14
Deltakere: Øyvind Eek Jensen, Oskar Gunbjørnsen, Martin Ingvaldsen, Krister Håkenrud og Kenneth Bergerud	
Referent: Kenneth	

Agenda

Sak 1: Introduksjon til oppgaven

Sak 2: Web

Sak 3: Besøk til Kongsberg Automotive

Sak 1

Diskusjon: Grappa introduserte veileder til oppgaven og hva den gikk ut på. Det ble også en gjennomgang av fordelingen av arbeid og gruppesammensetning.

Konklusjon: Veileder har nå grunnleggende forståelse av hva oppgaven går ut på og et innblikk i gruppens bakgrunn.

Ansvarlig person: alle

Sak 2

Diskusjon: Grappa skal ha en webside bestående av informasjon om hva grappa driver med og det endelige resultatet. Det skal foreligge informasjon om hva oppgave går ut på og dokumenter med løsninger.

Konklusjon: Veileder ønsker at mest mulig informasjon skal gjøres tilgjengelig for neste års studenter. Vi må respektere Kongsberg Automotives ønske om å hemlighetsholde informasjon så en mulig løsning er passordbeskyttelse. Dette vil gjøre at informasjonen kun er tilgjengelig for skolen og deres studenter. Dette kommer man tilbake til.

Ansvarlig person: alle

Sak 3

Diskusjon: Veileder har blitt invitert til oppdragsgiver for et møte. Dette er for å få et mer fysisk innblikk i hva grappa faktisk skal gjøre.

Konklusjon: Møtet finner sted på Kongsberg Automotive sine lokaler torsdag 06.02.14.

Ansvarlig person: alle

Møtetype: Veiledermøte	Sted: HBV D230
Dato: 13.02.2014	Neste møte: 20.02.2014
Deltakere: Øyvind Eek Jensen, Oskar Gunbjørnsen, Martin Ingvaldsen, Krister Håkenrud og Kenneth Bergerud	
Referent: Oskar	

Agenda

Sak 1: Prosjektmodellen

Sak 2: Teknologidokumenter

Sak 3: Iterasjoner av testing

Sak 4: Møteinnkalling/agenda

Sak 5: Oppfølgingsdokument

Sak 1

Diskusjon: Karoline følte ikke at prosjektmodellen vi viste fram under presentasjonen vår passet med måten vi har sagt vi skal jobbe på.

Konklusjon: Prosjektmodellen skal være et styringsverktøy som skal være sammenkoblet med Gantt-diagrammet. Prosjektplanen skal avspeile modellen vi har valgt.

Ansvarlig person: Alle

Sak 2

Diskusjon: Lite teknologidokumenter lagt ved i det vi leverte.

Konklusjon: Teknologidokumentene skal være en samling av prinsippskisser og diverse andre dokumenter i form av en rapport.

Ansvarlig person: Alle

Sak 3

Diskusjon: Øyvind mener det er viktig at vi har mange iterasjoner av testing.

Konklusjon: Vi ble enige om å utvide Gantt-diagrammet vårt slik at vi får med mange flere iterasjoner av testene. For eksempel tester vi en hvilken som helst sensor og implementerer den i systemet, så kjører vi en test til av det systemet osv.

Ansvarlig person: Alle

Sak 4

Diskusjon: Øyvind fikk ikke mail om møteinnkalling.

Konklusjon: Vi skal sende mail til Øyvind minst to dager i forveien om møteinnkalling, agenda for hva som skal tas opp og referat.

Ansvarlig person: Kenneth

Sak 5

Diskusjon: Skrivning av oppfølgingsdokumenter

Konklusjon: Vi skal lage et oppfølgingsdokument der vi fører hva vi gjorde den forrige uka og deretter hva vi skal gjøre den nye uka.

Ansvarlig person: Oskar

Møtetype: Veiledermøte	Sted: HBV D230
Dato: 20.02.14	Neste møte: 27.02.14
Deltakere: Øyvind Eek Jensen, Oskar Gunbjørnsen, Krister Håkenrud, Martin Ingvaldsen og Kenneth Bergerud	
Referent: Kenneth	

Agenda

Sak 1: Oppfølging til intern veileder

Sak 2: Dokumenter og dokumentasjon

Sak 3: 2. Presentasjon

Sak 1

Diskusjon: Det er ønsket av veileder å få en liten statusrapport om ting vi gjør/har planer om å gjøre da dette er relevant på det ukentlige veiledermøtet.

Konklusjon: Vi ble enige om å inkludere oppfølgingsdokument med møtoreferat. Dette gir et grunnleggende innblikk i gruppas status og plan på kort sikt.

Ansvarlig person: Alle

Sak 2

Diskusjon: Skal økonomidokumentet inkludere KAs utgifter? Hvordan unngå at gantt diagrammet drukner i dokumentasjon? Skal sensorer og generelt alternativer vi velger å ikke bruke dokumenteres?

Konklusjon: Alt som er relevant for oppgaven skal dokumenteres på økonomidokumentet, både gruppas private utgifter og KAs utgifter i forbindelse med prosjektet. Gantt diagrammet kan deles inn i faser og fordeles på flere sider. Dette er for å unngå ett stort dokument man ikke har like stor nytte av. Alle valg skal dokumenteres. Dette blir sett på som en studie av prosjektet og ved å dokumentere hva vi velger å ikke bruke, dokumenterer vi også avgjørelser.

Ansvarlig person: Alle

Sak 3

Diskusjon: Det ble gjort en diskusjon på når det er naturlig å gjennomføre 2. presentasjon, og hva denne skal inneholde.

Konklusjon: 2. presentasjon bør holdes når konsepter og valg er tatt. Da har man en fullstendig overblikk over oppgaven og de fleste problemstillingen rundt dette. Det kan gjerne repeteres litt på presentasjonen for å gi folk som ikke jobber med dette daglig bedre innblikk.

Ansvarlig person: Alle

Møtetype: Veiledermøte	Sted: HBV D230
Dato: 27.02.14	Neste møte: 13.03.14
Deltakere: Øyvind Eek Jensen, Oskar Gunbjørnsen, Krister Håkenrud, Martin Ingvaldsen og Kenneth Bergerud.	
Referent: Oskar	

Agenda

Sak 1: Ta en diskusjon på KA neste torsdag.

Sak 2: Microsoft Visio lisens.

Sak 3: Gjøre oppfølingsdokumentene mer tydelige.

Sak 1

Diskusjon: Vi var usikker på hvordan vi kunne bryte ned systemet vårt i forhold til teorien rundt reguleringsteknikken i oppgaven. Som for eksempel hvordan vi finner ut hva transferfunksjonen til systemet vårt er når man tar i betraktning at systemet vært er såpass ulineært.

Konklusjon: Øyvind foreslo å snakke med Tor neste gang vi er hos KA og få hans mening om reguleringssløyfa og problemstillinga. I og med at han har master i kybernetikk.

Ansvarlig person: Alle

Sak 2

Diskusjon: Koblingskjema til riggen eksisterer bare i Microsoft Visio. Dette programmet koster i underkant av 6000 kr, som ganske opplagt ikke gjør det til et alternativ.

Konklusjon: Vi ble enige om å enten laste ned en trial versjon eller gå til Richard Thue og spørre om en student lisens.

Ansvarlig person: Alle

Sak 3

Diskusjon: Øyvind mente at det burde være enklere og tydeligere å finne ut akkurat hvilket oppfølgingsdokument som hørte til når.

Konklusjon: Fra nå av skal vi skrive på hvilken uke det gjelder i «Hva her vi gjort?» seksjonen og i «Hva skal vi gjøre?» seksjonen, samt å skrive på antall timer vi forventer å bruke på hver aktivitet den gitte uka.

Ansvarlig person: Alle

Møtetype: Veiledermøte	Sted: HBV D230
Dato: 13.03.14	Neste møte: 20.03.14
Deltakere: Oskar Gunbjørnsen, Martin Ingvaldsen, Kenneth Bergerud, Krister Håkenrud og Øyvind Eek Jensen.	
Referent: Oskar	

Agenda

Sak 1: Valg av driver

Sak 2: Opprydding i timelister og Gantt-diagram

Sak 3: Fordelingen av oppgaver

Sak 1

Diskusjon: På fredag ettermiddag fikk vi mail om at driveren vår ikke var god nok til de formålene vi ønsket å bruke den til. Dette betydde at vi satt igjen med tre alternativer som måtte velges relativt fortløpende.

Konklusjon: Vi skal informere Øyvind så fort vi har tatt et valg.

Ansvarlig person: Alle

Sak 2

Diskusjon: Vi fant ut at vi var nødt til å oppdatere og rydde opp i timelistene samt gantt-diagrammet for å få litt bedre oversikt over hva hver enkelt person skal gjøre til alle tider. Det var flere ganger tidligere at vi ikke var helt klar over hva vi skulle gjøre og derfor satt vi oss ned med et dokument som egentlig ikke trengte å redigeres.

Konklusjon: Øyvind syntes det hørt fornuftig ut og hadde ingen spesielle innspill på dette.

Ansvarlig person: Martin Ingvaldsen

Sak 3

Diskusjon: Øyvind lurte på hvordan arbeidsfordelingen lå an de siste ukene.

Konklusjon: Martin har hovedsakelig holdt på med oppdatering av timelister, Krister har hatt ansvar for samtale med Rexroth, Kenneth har oppdatert prosjektplanen og Oskar har hjulpet Martin med timelister og skrevet på teknologidokumenter.

Ansvarlig person: Alle

Møtetype: Veiledermøte	Sted: HBV D230
Dato: 20.03.14	Neste møte: 27.03.14
Deltakere: Oskar Gunbjørnsen, Martin Ingvaldsen, Krister Håkenrud og Øyvind Eek Jensen.	
Referent: Oskar	

Agenda

Sak 1: Drivervalg

Sak 2: Prosjektmodell

Sak 1

Diskusjon: Vi fikk nylig mail av Tom der han fortalte litt om hva vi måtte fokusere mer på, og hva som ikke var like interessant. Noe av det han var klar på, var at vi hadde funnet masse informasjon på ting som for eksempel syklustider og litt informasjon rundt profibus, men det var ikke klart hvordan systemet vil fungere opp mot det vi ønsker.

Konklusjon: Vi skal ha et møte med KA fredag den 21. mars for drøfte fordeler og ulemper med Sercos in-line modul (analog kommunikasjon) og profibus (digital kommunikasjon) for så og ta en avgjørelse. Responstid, kostnad og kompleksitet med henhold til regulering er blant annet noe av det som skal vektlegges samt diskuteres.

Ansvarlig person: Alle

Sak 2

Diskusjon: Vi diskuterte også litt rundt prosjektmodellen samt iterasjonene i gantt-diagrammet og hvordan de skal gjenspeile hvordan vi jobber.

Konklusjon: Øyvind syntes de endringene vi hadde gjort i gantt-diagrammet var helt ok, men han fortalte at prosjektmodellen vi valgte, nemlig evolusjonær prototyping ikke passet til vært prosjekt. Han foreslo isteden at vi tok en kikk på «unified process», som er en iterativ og inkrementell programvareutviklingsprosess.

Ansvarlig person: Alle

Møtetype: Veiledermøte	Sted: HBV D230
Dato: 27.03.14	Neste møte: 03.04.14
Deltakere: Oskar Gunbjørnsen, Martin Ingvaldsen, Kenneth Bergerud, Krister Håkenrud og Øyvind Eek Jensen.	
Referent: Oskar	

Agenda

Sak 1: Prosjektmodell

Sak 2: Iterasjonsdokument

Sak 3: 2. Presentasjon

Sak 1

Diskusjon: Vi har vært nokså stresset og frustrerte den siste uka over prosjektmodellen med tanke på hvordan vi skal utforme samt i det hele tatt bli ferdig med den.

Konklusjon: Øyvind mente at vi ikke burde bruke alt for mye tid på dette og heller konsentrere oss med viktigere ting som valg av servo-driver og generell forberedelse til 2. presentasjon. Vi skal heller benytte en modifisert versjon av unified process, som er bedre tilpasset vårt prosjekt og rask å lage.

Ansvarlig person: Alle

Sak 2

Diskusjon: Etter å ha vært med på et par av de andre gruppene sine presentasjoner, kom vi på at vi må skrive et iterasjonsdokument etter hver iterasjon av den aktiviteten det er snakk om.

Konklusjon: Dette dokumentet skal skrives etter hver iterasjon av en aktivitet slik at vi bedre dokumenterer alt vi gjør.

Ansvarlig person: Alle

Sak 3

Diskusjon: Vi diskuterte også litt rundt 2. presentasjon og hva den skal inneholde.

Konklusjon: Vi ble enige om å ta med bilder, video, prinsippskisser, webside og hvor langt vi har kommet i prosjektet bare for å nevne noen. Øyvind mente også at det beste er å presentere det vi har på dette tidspunktet og det vi muligens lærer på mandag, neste torsdag.

Ansvarlig person: Alle

Møtetype: Veiledermøte	Sted: HBV D230
Dato: 23.04.14	Neste møte: 30.04.14
Deltakere: Oskar Gunbjørnsen, Martin Ingvaldsen, Kenneth Bergerud og Øyvind Eek Jensen.	
Referent: Oskar	

Agenda

Sak 1: Forsinkelser på bestilt utstyr

Sak 2: Testspesifikasjon

Sak 3: Gjennomgang av krav

Sak 1

Diskusjon: Softwaren vi bestilte er i rute til å ankomme den 24. april, men kontroll enheten (MLD) blir forsinket til den 9. mai på grunn av problemer med lisens.

Konklusjon: Vi har mer enn nok å gjøre slik at vi ikke på noen måte blir nødt til å sitte og vente på at vi får alt utstyret vårt, men flere utsettelse har vi ikke tid til.

Ansvarlig person: Alle

Sak 2

Diskusjon: På 2. presentasjon fikk vi generelt god tilbakemelding, men noe av det vi fikk beskjed om at måtte forbedres, var testspesifikasjonen.

Konklusjon: I testspesifikasjonen skal vi beskrive hvordan vi tester kravene våre. Den må kunne linkes til kravene, altså hva, hvordan og når.

Ansvarlig person: Alle

Sak 3

Diskusjon: Øyvind fortalte at vi var nødt til å være mer kritiske til graderingen av kravene våre. Hvis stort sett alle kravene er «A-krav», så har det ingen hensikt å gradere dem.

Konklusjon: Vi har et møte med KA den 24. april hvor vi skal gjennomgå hvilke krav vi muligens kan forkaste, hvilke vi kan endre graderingen til og hva slags nye krav vi stiller.

Ansvarlig person: Alle

Møtetype: Veiledningsmøte	Sted: HBV D230
Dato: 30.04.2014	Neste møte: 08.05.2014
Deltakere: Marting Ingvaldsen, Øyvind Eek Jensen og Kenneth Bergerud	
Referent: Kenneth	

Agenda

Sak 1: Regulering

Sak 2: Iterasjonsplan

Sak 1

Diskusjon: Reguleringen kan være ganske omfattende og være veldig tidkrevende.

Konklusjon: I første omgang gjøres dette enklest og mest linearisert som mulig. Dette er for å oppnå en virkende regulering mtp tiden. Blir det tid til det kan dette bygges på. Veileder og gruppe var enig i dette.

Ansvarlig person: Alle

Sak 2

Diskusjon: Det skrives allerede et iterasjonsdokument etter hver iterasjon. Dekker dette dokumentasjonen om iterasjonene.

Konklusjon: Grappa skal også opprette en iterasjonsplan som fungerer som et overordnet dokument til iterasjonsdokumentene. Denne skal inneholde planer om hva hver iterasjon skal inneholde.

Ansvarlig person: Alle

Møtetype: Veiledningsmøte	Sted: HBV D230
Dato: 08.05.14	Neste møte: 15.05.14
Deltakere: Marting Ingvaldsen, Øyvind Eek Jensen, Oskar Gunbjørnsen, Krister Håkenrud og Kenneth Bergerud	
Referent: Oskar	

Agenda

Sak 1: PID-regulering

Sak 2: Status i implementasjonsfasen

Sak 1

Diskusjon: Vi har kommet frem til to alternativer når det gjelder hva slags PID metode vi skal bruke. Enten blir det Ziegler Nichols eller Skogestad.

Konklusjon: Etter samtale med Øyvind, skal vi prøve å gå for Skogestads metode. Dette kommer av at Ziegler Nichols kan være noe tungvinn i blant.

Ansvarlig person: Alle

Sak 2

Diskusjon: Vi ligger bedre an enn vi forventet, siden MLDen originalt ikke skulle komme før den 9. mai. Det har gått greit med det elektriske og Labview, men vi sliter med å få kontakt mellom MLD og IndraWorks på PCen.

Konklusjon: Vi er nødt til å lese i manualene og finne en løsning på den måten.

Ansvarlig person: Alle

Møtetype: Møte om krav til kravspek	Sted: KA lab
Dato: 16.01.2014	Neste møte: 30.01.2014
Deltakere: Tom Heistad, Kenneth Bergerud, Krister Håkenrud, Martin Ingvaldsen og Oskar Gunbjørnsen	
Referent: Kenenth	

Agenda

Sak 1: Definere prioritet mellom clutch og girsimulator.

Sak 2: Definere noen krav, både funksjonelle og ikke-funksjonelle.

Sak 3: Diverse

Sak 1

Diskusjon: Oppgaven er todelt med en bit for clutchsimuleringen og en bit for girsimuleringen. KA ønsker seg girsimuleringen mest.

Konklusjon: Det ble bestemt at gruppa prioriterer girsimulering. Det skal simuleres fra et gir til et annet og det skal ses bort fra select bevegelse (diagonal giring).

Ansvarlig person: Hele gruppa.

Sak 2

Diskusjon: Utenom de kravene som er gitt i oppgavespesifikasjonen skal det også tilføyes noen ytterlige krav. Det er bevegelige deler så noen sikkerhetskrav må tilføyes etter en risikoanalyse. Det skal også legges til noen Software krav og eventuelle bonuskrav.

Konklusjon: Alt av utstyr skal være CE godkjent. Det skal monteres en nødstoppkrets og eventuelle andre sikkerhetsbarrierer som gjerder. Software skal være i stand til å logge resultatene/grafene og kalibrere sensorer. Eventuelle bonuskrav er temperatur, der vi ble enige om 20 grader +/- 10 grader.

Ansvarlig person: Hele gruppa.

Sak 3

Diskusjon: Gruppa lager fortløpende et dokument til KA med inputs de trenger

Konklusjon: 1. prioritet når man begynner den fysiske biten blir å måle på girkassen for få bedre enn eksisterende data.

Ansvarlig person: Hele gruppa

Møtetype: Veiledningsmøte	Sted: Kongsberg Automotive
Dato: 06.02.2014	Neste møte: 12.02.2014
Deltakere: Kenneth Bergerud, Krister Håkenrud, Martin Ingvaldsen, Oskar Gunbjørnsen, Øyvind Eek Jensen, Tom Heistad og Espen Elvekrok.	
Referent: Kenneth	

Agenda

Sak 1: Krav

Sak 2: Hva kan publiseres på web siden

Sak 3: Format KA ønsker på dokumentasjon før 1.presentasjon

Sak 1

Diskusjon: Er kravene fastlåst? Eventuelt hvem krav kan fires på?

Konklusjon: Hvis nødvendig kan clutchsimulatorens krav om 0mm til 25mm og tilbake til 0mm på 0.2 sekunder økes til 0.5 sekunder. Og kravet om posisjon på gir kan senkes til 30mm (?). Enkelte krav kan ses nærmere på etterhvert som målinger blir utført.

Ansvarlig person: Alle

Sak 2

Diskusjon: HBV ønsker at studenter fra skolen skal ha tilgang til websiden. Hva kan man publisere?

Konklusjon: Websiden blir en slags "skryteside" som inneholder hva oppgaven går ut på og hvem gruppen består av. Det skal ikke publiseres noe om KAs produkter.

Ansvarlig person: Alle

Sak 3

Diskusjon: Ekstern veileder og sensor skal få dokumentasjonen på forhånd av presentasjonen.

Konklusjon: Det skal opprettes en egen mappe i Dropbox som inneholder alle relevante dokumenter.

Ansvarlig person: Alle

Møtetype: Møte om spørsmål til servoen	Sted: KA lab
Dato: 20.02.14	Neste møte: 06.03.14
Deltakere: Tom Heistad, Kenneth Bergerud, Krister Håkenrud, Martin Ingvaldsen, Oskar Gunbjørnsen, Espen Elvekrok og Magnus Iversen.	
Referent: Oskar	

Agenda

Sak 1: Vil responstida på servoen være rask nok?

Sak 2: Sette opp en ny dag for testing i neste uke.

Sak 3: Lage milestones underveis.

Sak 4: Revidere prosjektplanen med tanke på LabVIEW.

Sak 5: Lage nye teknologidokumenter.

Sak 1

Diskusjon: Krister har ringt til produsenten av servomotoren KA har hos seg og har spurt litt rundt om det faktisk er mulig å gjennomføre oppgaven med det utstyret KA allerede har liggende, eller om vi må kjøpe inn en driver som er programmerbar.

Konklusjon: Vi har ikke nok informasjon om servoen til å gjøre noen fast beslutning på dette tidspunktet.

Ansvarlig person: Hele gruppa

Sak 2

Diskusjon: For å bedre kunne forstå hvor raskt systemet reagerer på for eksempel et sprang eller lignende, snakket vi om å sette av tid i neste uke til en test av nettopp dette og for å finne ut om vi kan bruke Kas utstyr.

Konklusjon: Vi har satt av tid til denne testen på mandag den 24. februar.

Ansvarlig person: Oskar, Kenneth og Martin.

Sak 3

Diskusjon: Tom mente det var lurt om vi lagde oss såkalte «milestones» eller mål underveis som vi følger.

Konklusjon: Vi var alle enige om at dette var et bra forslag og kommer til å gjøre det fra nå av.

Ansvarlig person: Hele gruppa

Sak 4

Diskusjon: Når vi spurte om lisens til LabVIEW, sa Tom at det var lurere å heller gjøre rede for akkurat hva vi skal, hvordan og hva slags utstyr som skal til. Og heller da vente litt med programmeringsfasen.

Konklusjon: Dette ble vi alle enige om.

Ansvarlig person: Hele gruppa

Sak 5

Diskusjon: Vi fikk beskjed om at det var lurt å dokumentere mulige andre løsninger vi ikke valgte.

Konklusjon: Vi ble enige om å lage et nytt teknologidokument som beskriver løsninger som vi valgte bort, siden det er minst like viktig som den metoden vi faktisk gikk for.

Ansvarlig person: Hele gruppa

Møtetype: Veiledningsmøte	Sted: Kongsberg Automotive
Dato: 06.03.2014	Neste møte: 13.03.2014
Deltakere: Kenneth Bergerud, Krister Håkenrud, Martin Ingvaldsen, Oskar Gunbjørnsen, Tom Heistad og Espen Elvekrok.	
Referent: Kenneth	

Agenda

Sak 1: Testing av motstand på motor

Sak 2: Sensorer

Sak 3: Div.

Sak 1

Diskusjon: Det skal utføres en test for å se hvordan motstanden motoren gir virker mot verdier.

Konklusjon: Vi vil lage et enkelt LabVIEW-program og bruke en luftsylinder for å se hvordan motoren oppfører seg med tanke på motstand når den får gitte verdier. Dette er også en nyttig måte å få testet litt responstider, som for eksempel responstid på DO.

Ansvarlig person: Alle

Sak 2

Diskusjon: I utgangspunktet er det tenkt 3 kraftsensorer og 3 posisjonssensorer. Det vil si på arm, aktuator og motor.

Konklusjon: For hastighet velger vi å bruke enkel matematikk istedenfor egen sensor (delta av posisjon). Når det gjelder kraft og posisjonssensorer blir det brukt 3 sett av hver for å måle alle vesentlige krefter og posisjoner. Bestillingstiden er lav (opptil 2 uker), men det kan hende at KA har alle nødvendige sensorer.

Ansvarlig person: Alle

Sak 3

Diskusjon:

- Kan range kortes ned fra +/-50mm?
- Avtale tidspunkt for 2. Presentasjon

Konklusjon: Rangen kan kortes ned til +/-30mm da dette er mer realistisk. Det skal sendes forespørsel om å få holde 2. presentasjon torsdag 03.04.14. Klokkeslett må bli halv 12 og utover.

Ansvarlig person: Alle

Møtetype: Valg av servodriver	Sted: KA lab
Dato: 21.03.14	Neste møte: Man-ons uke 13
Deltakere: Tom Heistad, Espen Elvekrok, Kenneth Bergerud, Krister Håkenrud og Martin Ingvaldsen	
Referent: Martin	

Agenda

Sak 1: Introdusere KA for alternativene for servodriver

Sak 1

Diskusjon: Grappa presenterte de forskjellige alternativene, og kom med den løsningen vi mener er den beste.

Konklusjon: Serco ser ut til å være det beste valget, så vi skal gå dypere inn på det, og finne ut mer om hvordan man styrer systemet, pris og annet vi trenger før vi kan evt sende inn bestilling.

Ansvarlig person: Hele gruppa

Agenda videre:

- Mindre sløyfe
- Kjøre overvåkning og referanser rett inn i NI
- Hva gjør Sercos med signalene?
- Tilkobling mellom NI, Sercos, MLD
- Hvor kan regulatoren implementeres? MLD eller LabVIEW
- Kommunikasjon MLD – LabVIEW
- Via pc?
- Hva er kommunikasjon graf?
- Detaljer kompatibilitet med systemet
- Leveringstid, purre på tilbud!!
- Pugh, graderinger
- Hvorfor er Serco bedre enn Profibus?
- Prosjektmodell
- Hva er uklart/hva ønsker vi?
- Oversikt over utstyr som vi har skal legges ved i dokumentasjon

Møtetype: Møte om valg av servo-driver	Sted: KA lab
Dato: 31.03.14	Neste møte: 04.04.14
Deltakere: Tom Heistad, Kenneth Bergerud, Krister Håkenrud, Martin Ingvaldsen, Oskar Gunbjørnsen og Espen Elvekrok.	
Referent: Oskar	

Agenda

Sak 1: Priser på Profibus alternativene.

Sak 2: Kjøring av programmet via MLD.

Sak 3: Hva skal vi ha med på presentasjon 2?

Sak 1

Diskusjon: Vi la fram hva de forskjellige typene alternativer for Profibus kom til å koste for Tom og Espen.

Konklusjon: I og med at det billigste alternativet kostet i underkant av 36000kr og det dyreste og mest omfattende i underkant av 45000kr, er dette ikke aktuelt. Vi skal derfor se etter andre løsninger der vi ikke må innom Profibus.

Ansvarlig person: Hele gruppa

Sak 2

Diskusjon: Vi vil aller helst ønsker å bruke LabVIEW som MMI, men dette ikke er mulig med Sercos.

Konklusjon: Til fredag skal vi finne ut om kjøringa av programmet kan foregå i MLD og bruke LabVIEW til å vise det. Vi skal også finne ut hvor mange analoge innganger MLD har.

Ansvarlig person: Hele gruppa

Sak 3

Diskusjon: Vi diskuterte også hva vi skal vise fram på 2. presentasjon førstkommende torsdag.

Konklusjon: Det å ta med alle alternativer vi har vurdert i forhold til valg av servo-driver, video klipp og interfacen vi skal bruke i LabVIEW, mente Tom var veldig lurt.

Ansvarlig person: Hele gruppa

Møtetype: Møte om ansvarsområder	Sted: KA lab
Dato: 24.04.14	Neste møte: 04.04.14
Deltakere: Tom Heistad, Kenneth Bergerud, Krister Håkenrud, Martin Ingvaldsen, Oskar Gunbjørnsen og Espen Elvekrok.	
Referent: Oskar	

Agenda

Sak 1: Ansvarsområder

Sak 2: Krav

Sak 3: Sensorer i prinsippskisser

Sak 4: Eget skap til nytt utstyr

Sak 1

Diskusjon: Til den siste delen av prosjektet, har vi delt inn nye ansvarsområder for å bedre disponere tiden. Vi kommer så klart til å hjelpe hverandre likevel om det skulle være noe en av oss står fast på.

Konklusjon:

Kenneth – LabVIEW

Martin – Elektrisk og oppkobling

Oskar – Regulering og IndraWorks

Krister – Regulering og IndraWorks

Ansvarlig person: Alle

Sak 2

Diskusjon: Fra forrige veiledningsmøte den 23.april, la Øyvind fram at vi muligens burde gå over kravene slik at ikke like mange av dem var gradert like høyt.

Konklusjon: Etter å ha tatt opp dette med KA, ble det bestemt at vi skal lage et oppdatert forslag til krav som vi da legger frem ved neste møte. Dette er viktig nettopp fordi det kun er KA som kan endre kravene.

Ansvarlig person: Alle

Sak 3

Diskusjon: Oppdragsgiver har hele veien vært nok så fornøyd med prinsippkissene vi har laget, men mente vi måtte få på plass akkurat hvordan sensorene blir implementert også.

Konklusjon: Vi satte oss ned med dette rett etter møte og kom godt i gang.

Ansvarlig person: Alle

Sak 4

Diskusjon: Siden oppdragsgiver muligens ville bruke riggen til andre formål i fremtiden, var det ønskelig at vi ikke roter til eller endrer for mye i skapet på rigg R-144.

Konklusjon: Det ble bestemt at vi lager et eget sted til alt det nye utstyret slik at vi unngår nettopp dette.

Ansvarlig person: Martin Ingvaldsen

Møtetype: Gruppemøte	Sted: HBV D230
Dato: 13.03.14	Neste møte:
Deltakere: Oskar Gunbjørnsen, Martin Ingvaldsen, Kenneth Bergerud og Krister Håkenrud.	
Referent: Oskar	

Agenda

Sak 1: Status på prosjektet

Sak 2: Oppdatere Gantt-diagram

Sak 3: Frister og milepæler

Sak 4: 2. Presentasjon

Sak 1

Diskusjon: Først diskuterte vi litt rundt statusen på prosjektet og hvordan vi ligger an.

Konklusjon: Var alle enige om at det har vært litt kaos til tider, men at vi fortsatt henger greit med.

Ansvarlig person: Alle

Sak 2

Diskusjon: Gantt-diagrammet har ikke vært like strukturert som det burde være og av den grunn har vi ikke bestandig vært helt klar over hva vi skal gjøre.

Konklusjon: Det ble bestemt at Gantt-diagrammet skulle være ferdig torsdag den 13. mars slik at vi virkelig kunne jobbe etter det.

Ansvarlig person: Alle

Sak 3

Diskusjon: I og med at 2. presentasjon nærmer seg med stormskritt, skrev vi ned datoer som hvert dokument skulle være ferdig.

Konklusjon:

Gantt diagram skal være ferdig i dag (13.03.14)

Presentasjon 2 (torsdag 03.04.14)

Ha en kravspek som er helt ferdig til fredag 04.03

Ha en prosjektplan som er helt ferdig til torsdag 27.03

Ha en testspek som er helt ferdig til mandag 17.03

Ha en testplan som er helt ferdig til torsdag 27.03

Ansvarlig person: Alle

Teknologidokument

GCS – Gir & Clutchsimulator

Forfatter: Kenneth, Krister, Martin og Oskar



KONGSBERG
AUTOMOTIVE

Innledning

I dette dokumentet vil du finne teknisk dokumentasjon som gjelder vårt prosjekt: GCS – Gir & Clutch Simulator. Det består av tre deler, som gjenspeiler de største fasene vi har hatt i løpet av prosjektet, med gjennomgående teknologidokumenter for de forskjellige delene av fasene. Vedlagt ligger aktuell dokumentasjon, dette er mer spesifisert i kapittel 3.6.

Mvh

GCS-teamet

Innholdsfortegnelse

Innledning	2
1 – Forberedelser og valg	5
1.1 – Måling av girkasse	6
1.2 – Testing av eksisterende utstyr (servomotor).....	9
1.3 – Simulering av servo	11
1.4 – Kommunikasjon med leverandør.....	11
1.5 – Valg av løsning for servodriver	12
1.5.1 – Parallell Interface	13
1.5.2 – MLD og Sercos in-line modul	15
1.5.3 – MLD og Profibus	17
1.5.4 – MLD uten buss-kommunikasjon.....	20
1.5.5 – Bestilling	22
2 – Oppkobling og implementasjon	23
2.1 – Klargjøring og oppkobling	24
2.1.1 – Utstyr.....	24
2.1.2 – MLD	25
2.1.3 – Tilkobling av sensorer	26
2.1.4 – Reset-lampe.....	30
2.1.5 – Girspak	31
2.2 – LabVIEW	32
2.3 – Test av sensorkommunikasjon.....	40
2.4 – Test av IndraWorks	42
2.5 – IndraWorks overordnet	46
2.6 – IndraLogic.....	53
2.7 – Regulering	57
2.7.1 – Parametere.....	58
2.7.2 – Parametere, Beregninger & Beskrivelse av Servo-system	61

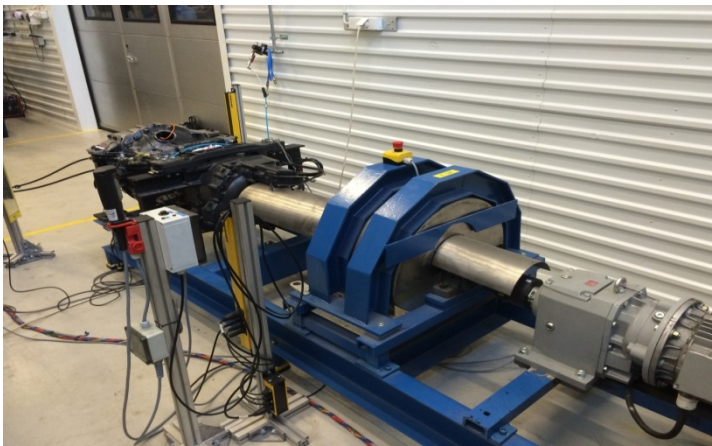
2.7.3 – MathLab/Simscape Simulering av Open-loop respons til Servomotor Bevegelsessystem	64
2.7.4 – Regulerings-Teknisk beregning av system.....	69
3 – Sluttprodukt.....	76
3.1 – Hvordan fungerer simuleringen?.....	77
3.2 – Rigg.....	78
3.2.1 - Girspak.....	79
3.2.2 - Skap.....	81
3.3 – Videreutvikling.....	83
3.4 – Vedlegg.....	84

1 – Forberedelser og valg

I dette kapitlet ligger dokumentene som legger grunnlaget for hvordan vi løste oppgaven. Det inneholder alt det tekniske vi gjorde fra måling av girkasse til og med bestillingen av servodriveren. Valget av servodriver (kapittel 1.5) er det viktigste valget vi har tatt i løpet av prosjektet og er basisen for kapittel 2, der vi starter med sammensetning av systemet. Noe informasjon er vedlagt for dypere forståelse.

1.1 – Måling av girkasse

For å ha et utgangspunkt for girkasse-simulatoren er det nødvendig å vite hvordan kraften øker eller minker, alt etter hvilken posisjon av girsekvensen den er i. Vi hadde en graf til rådighet, men denne var noe utdatert så etter ønske fra KA, skulle vi gjøre nye målinger. Motstanden fra girkassen endrer seg avhengig av hvordan man girer, så vi valgt å måle i tre forskjellige hastigheter; slow, normal, fast. Giringene ble målt fra 5. til 6. til 5. gir på en Scania lastebil-girkasse, dette tilsvarer giring mellom 2. og 3. gir på en vanlig personbil. I tillegg gjorde vi en måling ned til 4. gir, fordi denne gir større krefter, slik at vi fikk et godt overslag.



Figur 1: Girkasserigg

Utstyr

Rigg: R-128

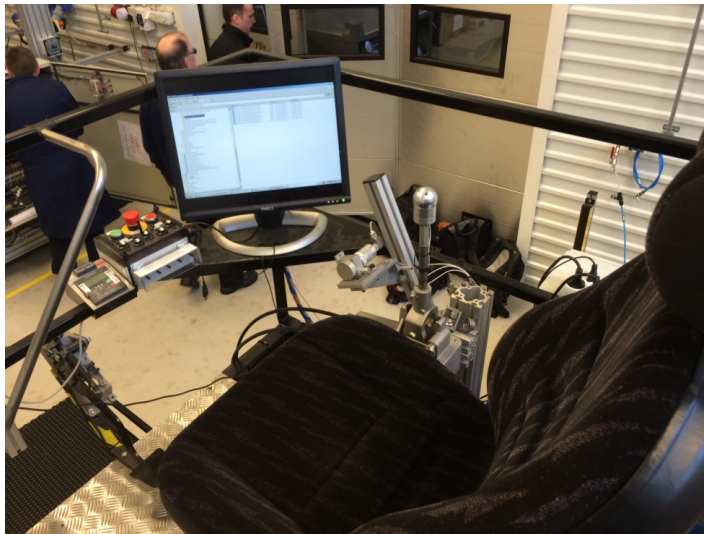
Lastcelle: I-832

Girkule: I-631

Potmeter (tråd): I-734

Posisjonssenor: Har ikke I-nummer, men målt med skyvelære

Skyvelære: I-730

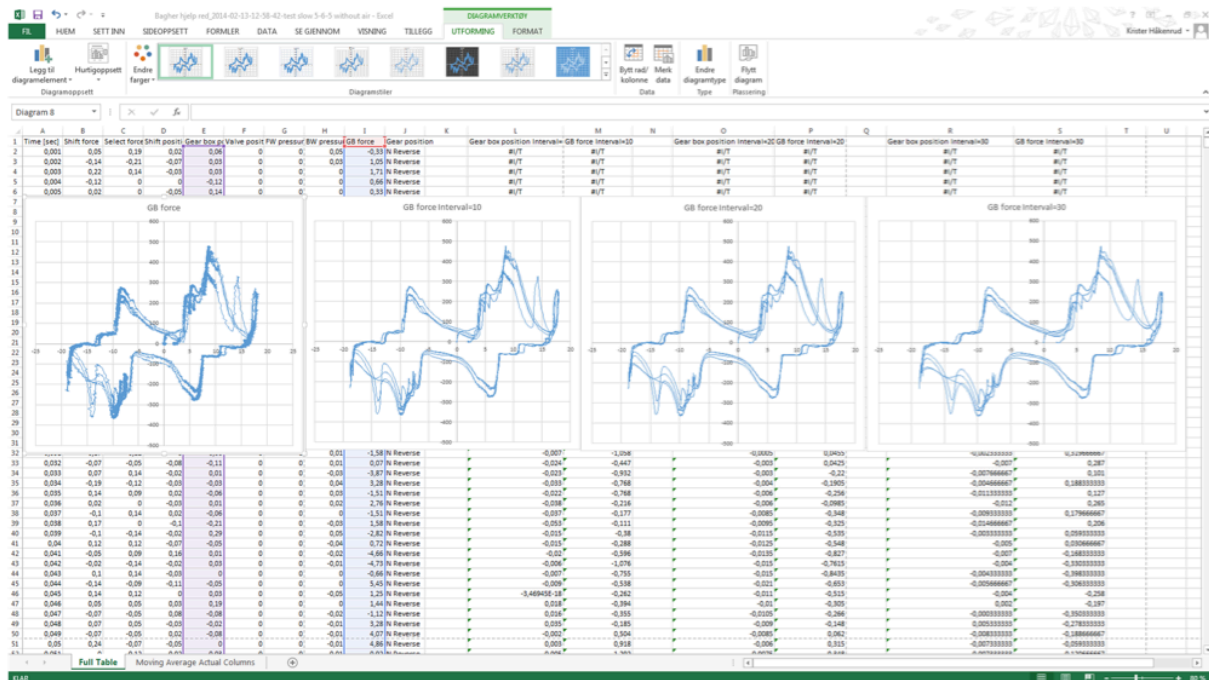


Figur 2: Rigg for manuell giring

Fremgangsmåte

Vi brukte en girkasserigg (R-128) hos KA for å utføre målingene. Dette er en rigg utstyrt med en Scania girkasse og en stor motor for rotasjon. For å måle krefter og posisjon på girkassen, brukte vi En last celle på 1kN og en posisjonsmåler (Datablad). For å gire manuelt, brukte vi en girstang med en spesiell girkule med innebygd kraftsensor, så vi fikk målt kreftene som ble påført girspaken. På girstanga var det festet en trådpotmeter, som målte posisjonen til girstanga. Alle testene ble utført med og uten girskiftsystemet til KA.

Resultater



Figur 3: Kraft/posisjonsgraf etter måling

I bildet over ser man grafen vi fikk etter målingen. Den første grafen (helt til venstre) er den originale grafen, men vi ønsket å fjerne forstyrrelser og lignende. Vi ryddet opp ved hjelp av "moving average" i Excel, for å vise bedre selve grafen.

Hele prosessen ligger i en ppt (Hvordan Importere & Håndtere girmålinger) i mappen: 10 Logging girkasse krefter 13.02.14.

1.2 – Testing av eksisterende utstyr (servomotor)

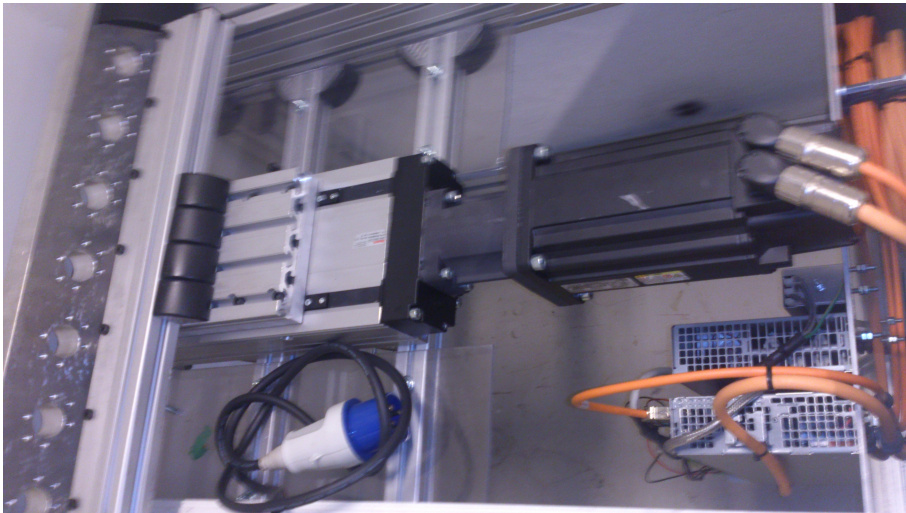
Hensikten med denne testen er å definere input/output med tanke på overføringen for en rotasjon til antall mm, altså posisjonen. Og hvilken hastighet, samt akselerasjon vi klarer å få servomotoren til å oppnå. Dette er nødvendig for å vite om servomotoren KA allerede har liggende hos seg tilfredsstillende kravene vi har fått spesifisert.

Utstyr

Data med IndraWorks DS

Skyvelære

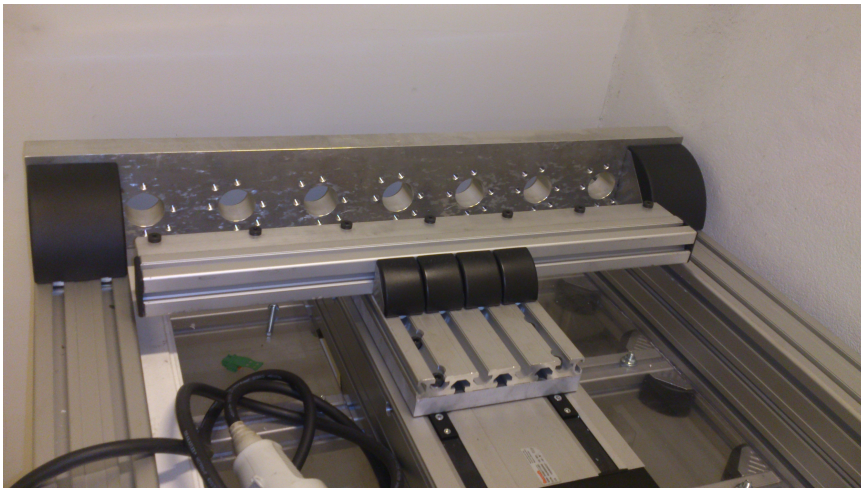
Test-rigg: R-144



Figur 4: Servomotor

Fremgangsmåte

Etter å ha fått et inntrykk av brukergrensesnittet til IndraWorks DS og trykket på reset-bryteren, startet vi med små forflytninger av gangen med høy hviletid for sikkerhets skyld. Etter å ha funnet ut av overføringen til servomotoren, testet vi med høyere hastighet og akselerasjon. Vi ville også hvor langt i millimeter en full omdreining var (360 grader). Dette ble målt ved å kjøre 360 grader i positiv retning og 360 grader i negativ retning, altså totalt 720 grader. Deretter kjørte vi 1440 grader (720 i både positiv og negativ retning), og tilslutt 1800 grader i både positiv og negativ retning. Vi satte merker underveis og målte med skyvelære hvor stor vandring dette ga og fant deretter ut at en omdreining tilsa 10mm.



Figur 5: Rigg

Resultater

Etter nøye måling fant vi ut at 1cm vandring på riggen tilsvarte en hel omdreining på servomotoren, altså 360 grader. Vi testet opp til 5 omdreininger fram og tilbake, dette er det samme utslaget som vi har fått spesifisert krav om nemlig 100mm.

Parametere:

Hastighet: 5000rpm

Akselerasjon: 1050rad/s^2

Dwell time: 0,001s

End position negative: 1800 grader

End position positive: -1800 grader

Limit parametere:

Hastighet: 5100rpm

Akselerasjon: 1100rad/s^2

Dwell time: N/A

End position negative: 2000 grader

End position positive: -2000 grader

1.3 – Simulering av servo

Vi søkte en god del på internett når det gjaldt hvordan angripe og beregne regulerings-kretsen. Vi fant ut at i MathLab med de inkluderte verktøyene Simulink og Simscape, har mulighet til å simulere «open-loop» responsen til nærmest alle typer motorer.

Ved hjelp av Simscape kan man plote grafer som viser responstiden til Hastighet,

Posisjon og Moment (Torque), ved å parametrisere motor i Simscape modellen. Disse «open-loop» grafene kan vi igjen benytte for beregning av regulerings-systemet.

Full oversikt over beregninger og simuleringer ligger vedlagt i dokumentet: Undersøkelser, beregning og koding av regulator.

1.4 – Kommunikasjon med leverandør

Vi kontaktet Boch Rexroth for første gang 14.02.14, der vi ble satt i kontakt med teknisk support avdeling. Vi oversendte utstyrsliste og prinsippskisser på det vi hadde tilgjengelig, og spurte om mulige løsninger. Der fikk vi god hjelp og, sendte mailer frem og tilbake om hvordan vi skulle løse det, pristilbud, hva som fungerer og ikke fungerer, og lignende. Full referat fra mailene ligger vedlagt i dokumentet: Kommunikasjonshistorikk med Bosch Rexroth.

1.5 – Valg av løsning for servodriver

For å kunne drive systemet, trengs det en servodriver. Vi har kommet frem til fire forskjellige løsninger med noen delløsninger som er aktuelle;

Parallell Interface: 16 digitale utganger og innganger som finner sted på baksiden av riggen.

MLD (Motion Logic Drive) og Sercos in-line modul: analog kommunikasjon

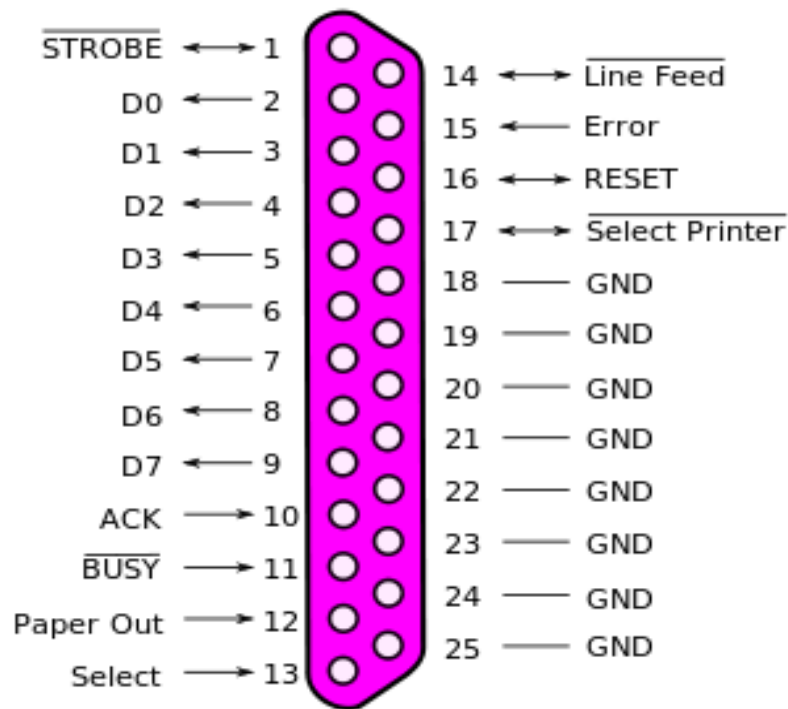
MLD med Profibus: kommunikasjon via Profibus

MLD uten buss-kommunikasjon: unngå buss-kommunikasjon.

Her skal vi beskrive de fire forskjellige alternativene og sette de opp mot hverandre for å komme frem til den løsningen som vil bli aktuelt for vårt system.

1.5.1 – Parallell Interface

På det utstyret vi har i dag sitter det allerede en parallell «interface», med 16 DI (digitale innganger) og 16 DO (digitale utganger). En parallell interface er en kanal som er i stand til å overføre mer enn en bit samtidig. De fleste PCer kommer med minst én parallell interface.

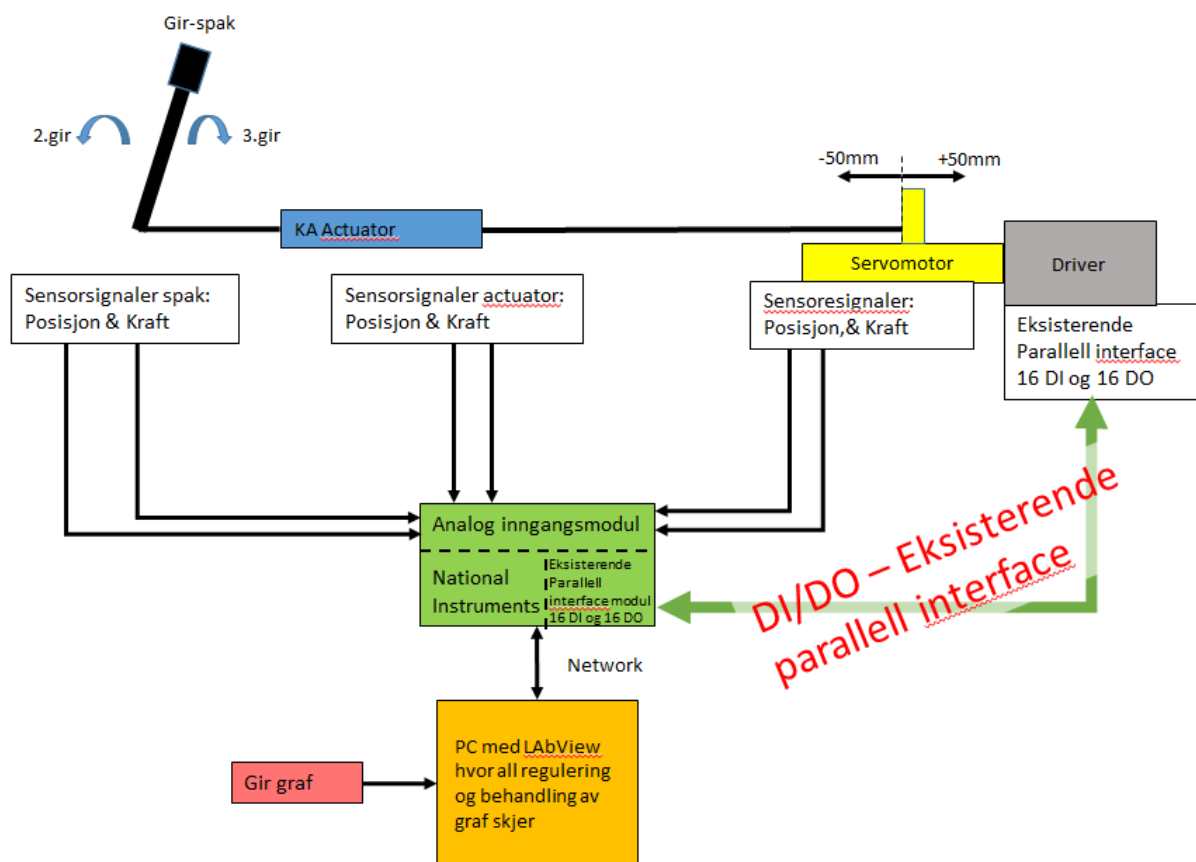


Figur 6: Pinne-nummer og funksjon på parallell interface

Spørsmålet er om responstidene er raske nok for vår applikasjon, siden kraftgrafene må innom en PC som er koblet til National Instruments sin CompactRIO og deretter måles opp mot et referanse signal som da gir et pådrag avhengig av hvor stort avviket er.

Konklusjon:

Etter samtale med Rexroth har vi fått beskjed om at parallell interface ikke er godkjent for de kravene vi stiller til driveren. Dette kommer av at vi ønsker å kommunisere i «real-time» og analogt. Det har også lavere bit-rate enn Profibus alternativet. I tillegg fant vi ut at en slik løsning ikke er veldig god når vi ønsker å benytte en dynamisk kraft som endres ut ifra hvilken posisjon gir-spaken er i. Om vi bare hadde behøvd enten ingen kraft eller full kraft, som for eksempel på et samlebånd som dytter esker rundt omkring, ville dette fungert.



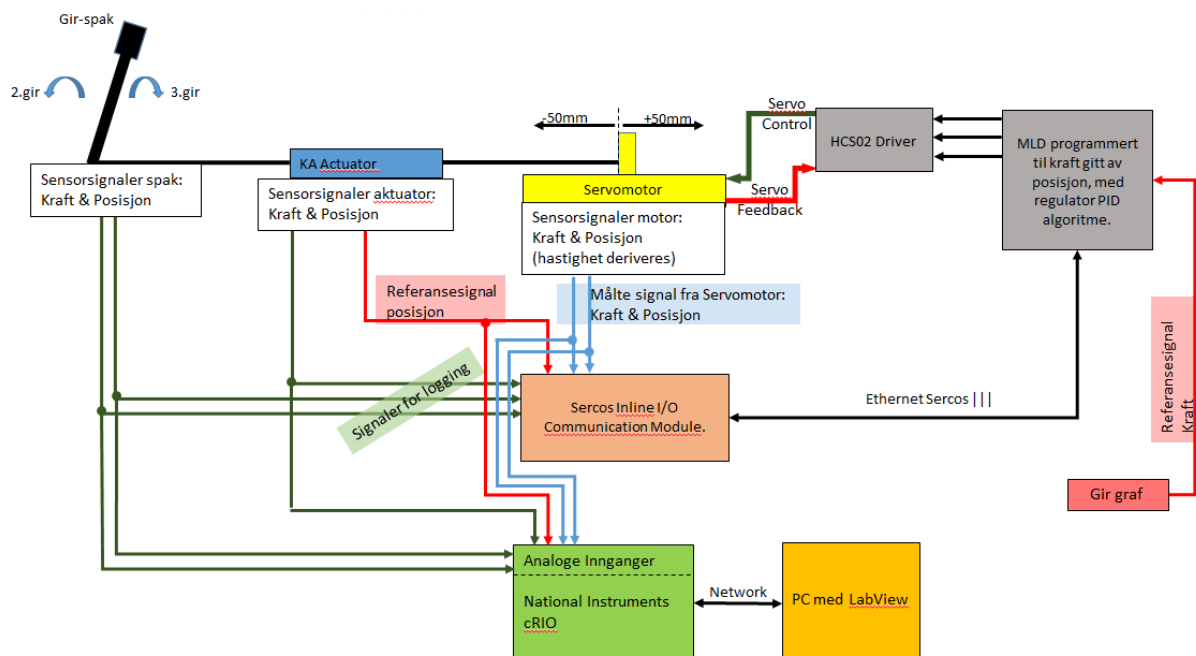
Figur 7: Prinsippkisse alternativ 1

1.5.2 – MLD og Sercos in-line modul

I denne løsningen foregår kommunikasjonen analogt via Sercos in-line modul (kommunikasjonsbuss). MLD (Motion Logic Drive) er en Rexroth servodriver med integrert PLS som kan programmeres etter IEC 61131-3 standard, til å følge kraftgrafen vår ut ifra kraft og posisjon. Den kjører servomotor i samarbeid med HCS02 (power section) som vi allerede har.

Grunner for og imot dette alternativet:

- Fra 10-100 ganger raskere enn «fieldbus» alternativene, inkludert Profibus.
- Sercos er en real-time kommunikasjons bus.
- Sercos er standard bus system på Bosch Rexroth sine komponenter, det vil si ingen kompatibilitetsproblemer.
- Cycle-time ned mot 32,25 μ s (ref. manual/flyer).
- Kostnad på tilleggsutstyr er lavest av de to aktuelle alternativene.
- Det er imidlertid ikke kompatibelt med LabVIEW.



Figur 8: Prinsippkisse alternativ 2

Cycle-time: Syklisk oppdateringstid til enheten så lavt som 31,25 μ s(ref. manual/flyer).

Industrielt Ethernet har blitt den nye standarden for informasjons-nettverk i produksjon, og markedet krever nettopp dette for servo-drivere. Som sagt åpner dette muligheter for 10-100 ganger raskere fieldbus alternativer.

Kostnader servo-tillegg:

Sercos in-line modul: 13 565 kr

MLD driver: 6 775 kr

Firmware: 3 487 kr

IndraWorks Software til Programmering: 3 010 kr

Totalt ca.: 26 838 kr.

Analog inngangsmodul til NI cRio:

NI 9215 4-Channel, 100 kS/s/ch, 16-bit, ± 10 V Analog Input Module, kostnad: 4 045 kr

Totalt med Sercos enkel- og AI- modul til NI cRio ca.: 30 883 kr.

Leveringstider:

Vanlig leveringstid på dette alternativet er rundt én måned, men siden vi forklarte at vi hadde dårlig tid, skulle de se om de klarte fremskyve bestillingen slik at det tok tre uker isteden.

Konklusjon:

Løsning ikke gjennomførbar siden National Instruments ikke har Sercos moduler til sitt utstyr, de bruker fremdeles Profibus. (ref.e-post og samtale: Make Nguyen | Technical Support | National Instruments Finland Oy).

MLD med Sercos III, kan da ikke kommunisere med NI cRIO.

1.5.3 – MLD og Profibus

I denne løsningen foregår kommunikasjon ved hjelp av Profibus. Dette er en kommunikasjonsstandard innen automatikk. Målesignaler henter signaler til LabVIEW hvor det blir behandlet og regulert. Deretter sendes posisjon, hastighet og kraft til MLD-drive via Profibus. Profibus ikke er en real-time bus, derfor vil signalbehandling ta lenger tid.

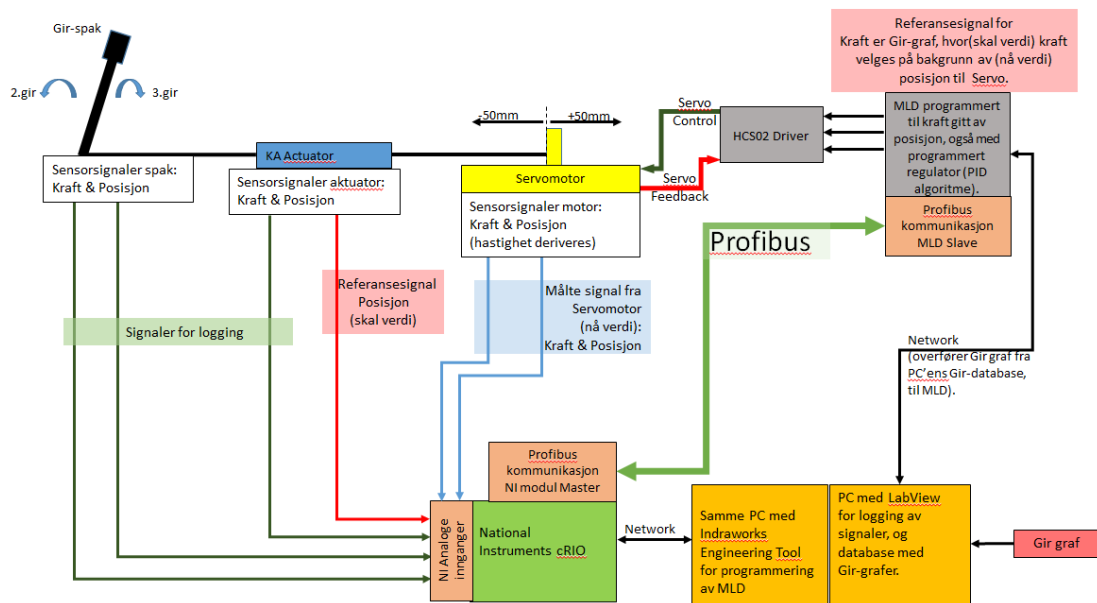
Rexroth tilbakemelding:

«Det utstyret dere har, kan ikke styres via Profibus. Dere trenger ny drive med Profibus-kort på. Jeg foreslår MLD drive med Profibus + programvare for programmering.»

MLD (Motion Logic Drive) er en Rexroth servo- driver med integrert PLS som kan programmeres etter IEC 61131-3 standard. Den kjører servo-motor i samarbeid med HCS02 som vi allerede har.

Grunner for og imot dette alternativet:

- Profibus er ikke en real-time bus, tregere av den grunn.
- Profibus er ikke standard bus system på Bosch Rexroth sine komponenter, men de har egne Profibus-kort til sine drivere derfor ingen kompatibilitetsproblemer.
- Bit-rate: 9,6 kbit/s → 12 Mbit/s, avgjort av Bit-transmission layer method.
- Cycle-time rundt 1ms (ref. Bosch Rexroth).
- Må ha tilleggsutstyr i form av Profibus-modul til NI cRio.
- Kostnad på tilleggsutstyr er høyest av de to aktuelle alternativene.



Figur 9: Prinsippskisse alternativ 3

Bit-rate: 9,6 kbit/s → 12 Mbit/s, avgjort av Bit-transmission layer metode.

Cycle time: “Frame Checking Sequence” bestemmer dette basert på bit-rate og lengden til dataserien.

Kostnader Servo tillegg:

Tilleggs kostnader ref. første tilbud feilaktig inkludert utstyr vi har: *Totalt ca.: 33 853 kr*

Tilleggs kostnader ref. korrigeret tilbud ekskludert utstyr vi har: *Totalt ca.: 21 478 kr*

Kostnader NI tillegg:

Må her i tillegg muligens ha Profibus Master/Slave Modul til NI cRIO, kostnad: 15 000 kr

Eller en av de enklere versjonene av Profibus moduler til NI cRIO hvis de er tilstrekkelig, kostnad: 5-6 000 kr

Analog inngangsmodul til NI cRio:

NI 9215 4-Channel, 100 kS/s/ch, 16-bit, ±10 V Analog Input Module, kostnad: 4 045 kr

Totalt med Profibus enkel- og AI- modul til NI cRio ca.: 31 600 kr.

Totalt med Profibus S/M- og AI-modul til NI cRio ca.: 40 600 kr.

Leveringstider:

Vanlig leveringstid på dette alternativet er rundt én måned, men siden vi forklarte at vi hadde dårlig tid, skulle de se om de klarte fremskyve bestillingen slik at det tok tre uker isteden.

Konklusjon

Etter samtale med oppdragsgiver nede på KA, fikk vi beskjed om at dette alternativet ikke er aktuelt til den prisen vi la frem for dem (fra 36 000 – 45 000kr). Men siden da har vi fått et oppdatert tilbud som ikke er like ille med tanke på pris. Dette kommer av at i det originale tilbudet ble det tatt med i betraktningen deler og utstyr KA allerede var i besittelse av og ting vi ikke trengte.

Denne løsningen er av de siste gjenværende løsningene om vi har lyst til å benytte LabVIEW som MMI (Man Machine Interface). Det var dette KA originalt ønsket seg, siden det var nettopp dette de hadde mest erfaring og kompetanse med. Men samtidig så er vi nødt til å benytte MLD uansett, noe som krever kurs og ekstra kompetanse likevel.

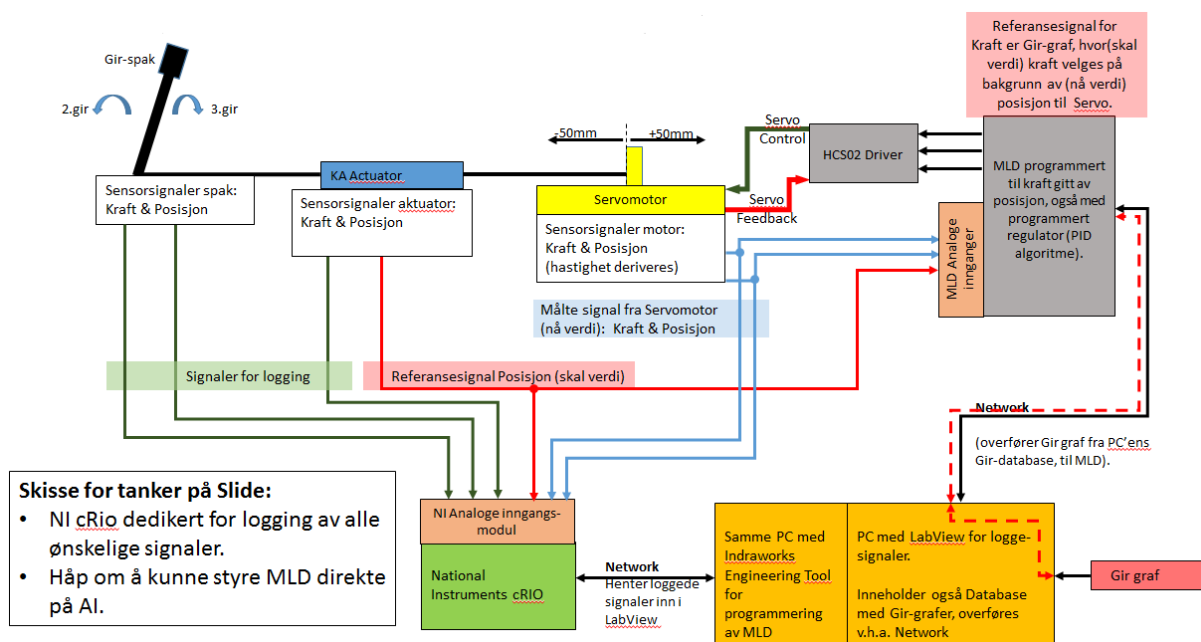
1.5.4 – MLD uten buss-kommunikasjon

Dette er den aller nyeste løsningen vi har kommet over, og det skjedde under møte med KA mandag den 31. mars. Ideen er at vi går utenom buss-kommunikasjon i det hele tatt, om dette måtte være enten Sercos eller Profibus, og heller går direkte fra MLD til PC med LabVIEW som vist på figur 5.

Med dette alternativet ønsker vi å utføre all reguleringsteknisk styring på MLD ved hjelp av IndraWorks. Grafene lastes direkte inn i MLD fra LabVIEW via TCP/IP (kommunikasjonsmetode) som sett på figur 5. LabVIEW vil da fungere som master og IndraWorks som slave.

Grunner for og imot dette alternativet:

- Det absolutt billigste alternativet.
- Det går utenom kommunikasjonsbussen og forenkler derfor systemet en god del.



Figur 10: Prinsippskisse alternativ 4

Kostnader servo-tillegg:

MLD (control unit)	<i>Totalt: 9 321,85 kr</i>
Firmware	<i>Totalt: 3 657,75 kr</i>
Software (IndraWorks)	<i>Totalt: 2 656,50 kr</i>
<i><u>Totalt med MLD, firmware og software: 15 636,1 kr</u></i>	

Leveringstider:

Leveringstid på softwaren, altså IndraWorks tok kun tretten dager fra vi bestilte den 11. april til den 24. april. Men på grunn av komplikasjoner med lisens, blir MLD og firmware forsinket til den 9. mai. Dette er kun fjorten dager før alle dokumenter skal leveres inn, derfor er det veldig kritisk at det ikke blir flere forsinkelser.

Konklusjon:

Dette alternativet er helt klart det mest gunstige med tanke på pris og oppkobling, siden vi som sagt hovedsakelig trenger MLD-driveren, firmware og software. Det er derfor vi har valgt å gå for dette alternativet over de andre. Og selv om vi ikke kun bruker LabVIEW som MMI lenger, er den nye løsningen med LabVIEW som master og IndraWorks som slave fullt realiserbar.

Pugh Matrix:

Pugh Matrix Servodriver					
Kriterier	Viktighet (1-10)	Alternative Konsepter			
		Parallell int.	Serco	Profibus	MLD uten buss-kommunikasjon
Responstid	9	2	8	6	10
Kompabilitet med utstyret	10	10	2	10	2
Pris	9	8	5	3	10
Enkelhet med hensyn på regulering	8	0	8	6	10
Kompabilitet med systemet	10	0	2	10	2
Leveringstid	7	9	6	6	6
Ferdigstillingsmulighet	10	0	4	2	8
Sum		29	35	43	48
Vekted sum		253	303	391	422

Figur 11: Pugh Matrix for valg av servodriver

1.5.5 – Bestilling

Vi valgte å gå for alternativ 4 og sendte inn bestillingen inn 11.04.14.

Pos.	Material/betegnelsen	Ant. Enhet	Bruttopris/enhet		Nettopris/enhet	Total
40	R911335358 SWA-IWORKS-MLD-12VRS-D0-DVD**-COPY SOFTWARE Varekode: 85234925 Opprinnelsesland: DE Vekt pr stk: 0,050 KG Leveringsdato: 24.04.2014	1 STK	3.542,00	-25,00 %	2.656,50	2.656,50
Total pris uten m.v.a.					NOK	2.656,50

Figur 12: Ordrebekreftelse del 1

Pos.	Material/betegnelsen	Ant. Enhet	Bruttopris/enhet		Nettopris/enhet	Total
50	R911315731 CSH01.1C-S3-ENS-NNN-MA1-NN-S-NN-FW CONTROL UNIT Varekode: 85049099 Opprinnelsesland: DE Vekt pr stk: 0,628 KG Leveringsdato: 09.05.2014	1 STK	12.429,13	-25,00 %	9.321,85	9.321,85
51	R911334746 FWA-INDRV-MPH-08VRS-D5-1-NNN-ML FIRMWARE Varekode: 85423275 Opprinnelsesland: DE Vekt pr stk: 0,010 KG	1 STK	4.877,00	-25,00 %	3.657,75	3.657,75
Total pris uten m.v.a.					NOK	12.979,60

Figur 13: Ordrebekreftelse del 2

Vi mottok software 24.04 og control unit med firmware 06.05.

2 – Oppkobling og implementasjon

I dette kapitlet tar vi for oss sammensetningen og testingen av systemet. All regulering, testing av kommunikasjon for data og sensorer, elektriske og mekaniske forberedelse blir beskrevet her. Mye av informasjonen vil bli linket til forskjellige vedlegg, som diverse manualer og videre utlegg om reguleringer og kodinger.

2.1 – Klargjøring og oppkobling

Denne delen består av forberedelser og oppkobling til den siste systemtesten, hvor systemet skal testes i sin helhet, det vil si den dynamiske delen av testingen. Her kobler vi opp alt nødvendig utstyr og legger opp alle kabler og ledninger. Hovedsakelig vil dette dreie seg om å få på plass ei lampe som lyser når reset bryter er trykket inn, få kontakt med sensorer til pc via National Instruments, sette inn MLD i HSC202 «power section» på baksiden av riggen og da få kontakt med IndraWorks via denne.

2.1.1 – Utstyr

Grønn lampe: BACO 33EAGL

Posisjonssensorer: Penny+Giles SLS130 og BEI SENSORS CD50 (trådpotmeter)

Lastcelle: I-916 og I-698 og girkule-lastcelle

Signalforsterker: Clip AE 101

Rigg: R-144

MLD: CSH01C

Power section: HSC202

Koblingskontakter og bananplugg

Diverse ledninger

Diverse verktøy

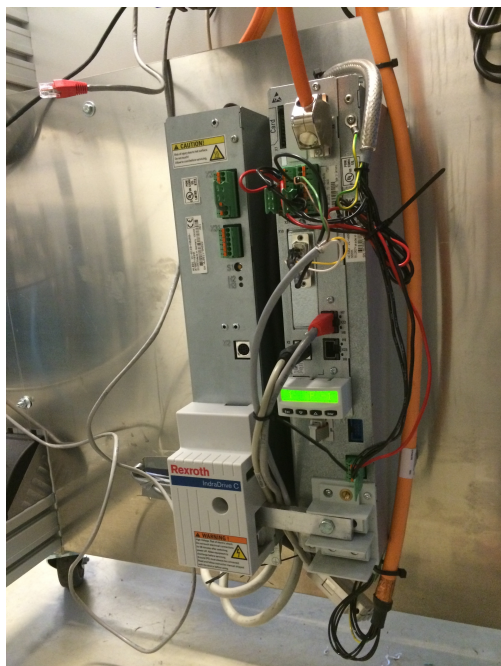
Microsoft Visio

2.1.2 – MLD

Installasjonen av MLD i HSC202 kan sammenliknes med et «motherboard» hvor vi skal bytte grafikkortet eller lignende. Som man ser på det første bildet har vi simpelthen fjernet den eksisterende kontroll-enheten, og satt inn den nye i HSC02 (power section). Vi har også tatt vekk multipluggen og koblet til noen ekstra ledere fra sensorer som skal leses i IndraWorks.



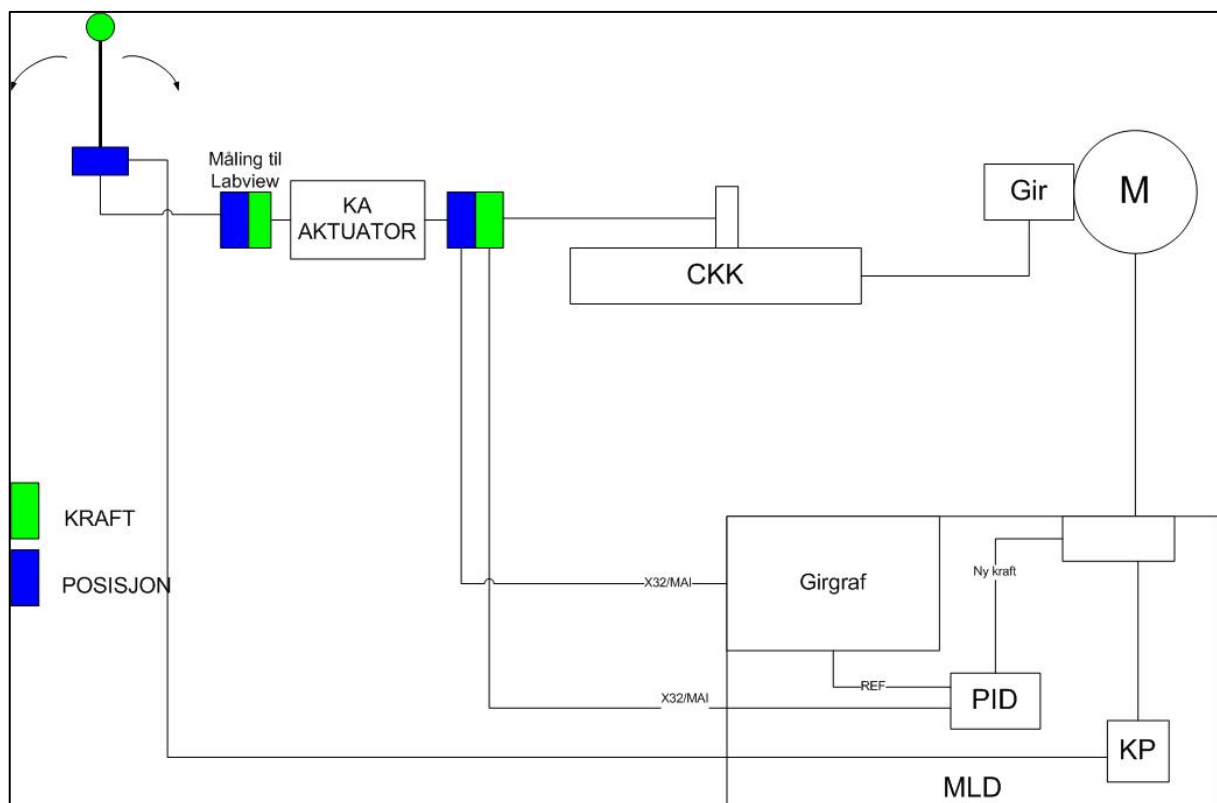
Figur 14: Opprinnelig modul



Figur 15: Ny MLD

2.1.3 – Tilkobling av sensorer

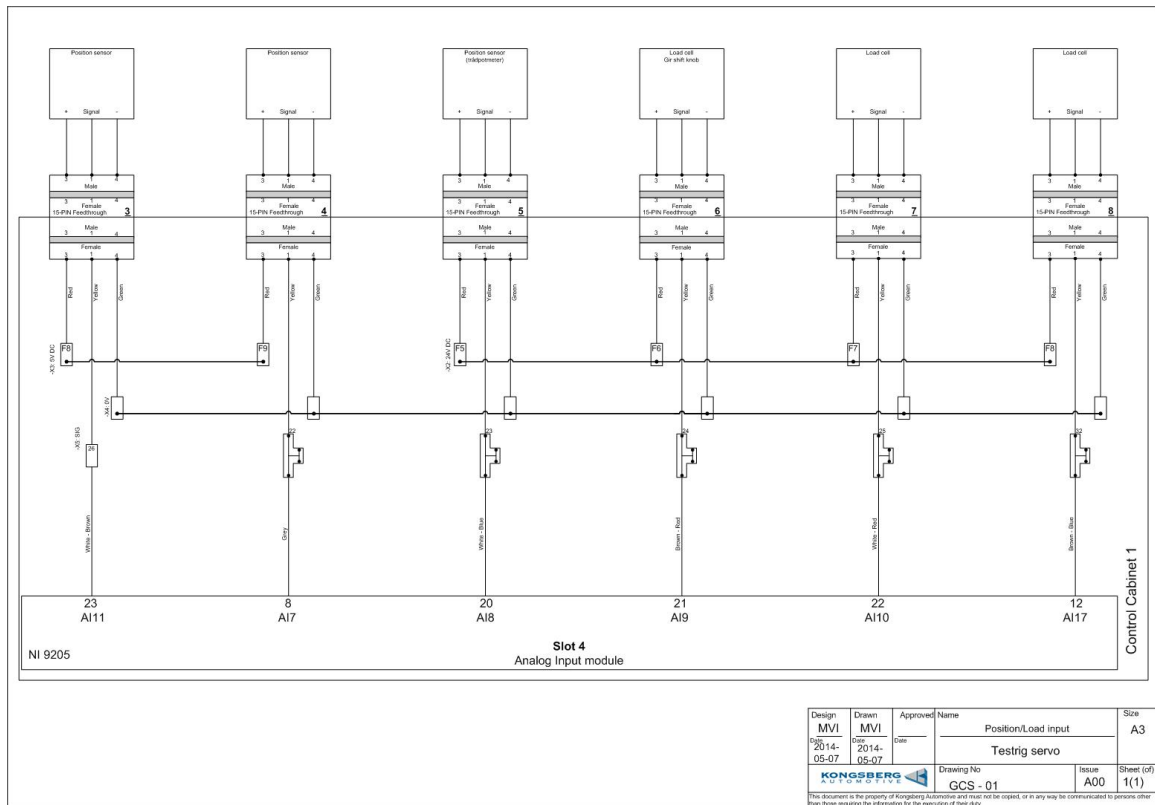
Vi har til sammen seks sensorer (tre last celler og tre posisjonsgivere) i systemet vårt som skal leses av og gi referanser til reguleringen. I første omgang bruker vi bare tre for å få lest av de viktigste målingene. Disse er; posisjonsgiver for servo, trådpotmeter for girspak og lastcelle ved servo. Alle sensorene skal leses i LabVIEW, i tillegg skal sensoren mellom aktuator og servo, posisjonsgiver mellom aktuator og servo, og posisjonsgiver ved girspak inn i IndraWorks.



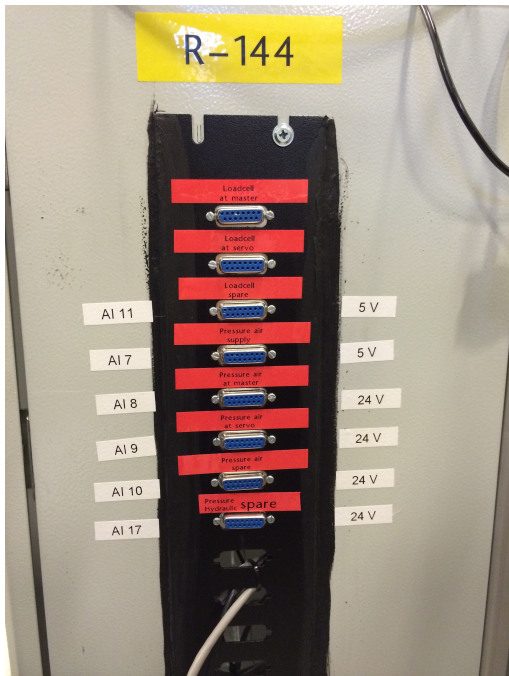
Figur 16: Tilkobling sensorer outside

For å koble til de sensorene vi trenger måtte vi ha lett tilgjengelige koblingspunkter slik at vi lett kunne koble til og fra sensorene. Disse sto allerede på skapet, men vi måtte koble til to ekstra slik at vi fikk seks innganger. Disse inngangene er koblet til cRio, 5V/24V og 0V. I tillegg er tre av inngangene også koblet til den nye MLD-modulen for avlesning i IndraWorks. Det elektriske systemet for sensorene ble også tegnet i Microsoft Visio.

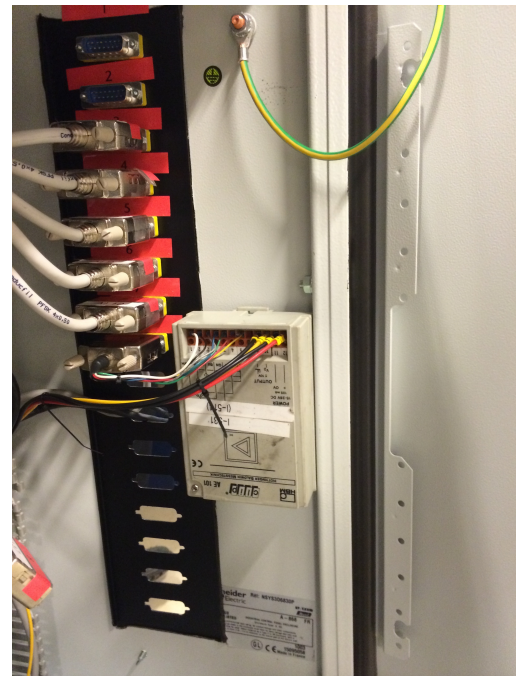
GCS – Gir & Clutchsimulator



Figur 17: Visio-tegning av sensorer

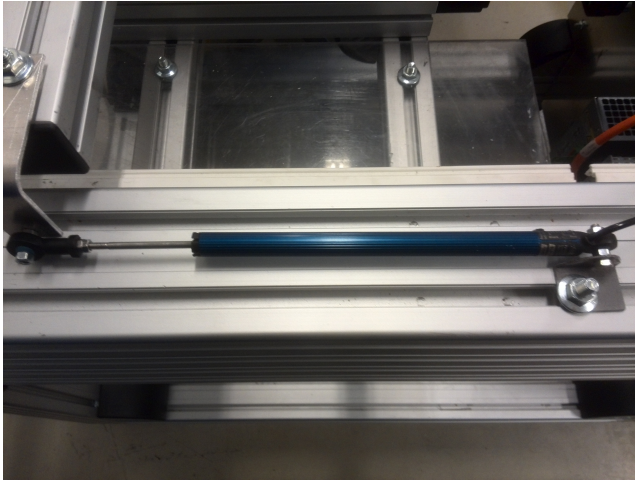


Figur 18: Tilkobling sensorer utside

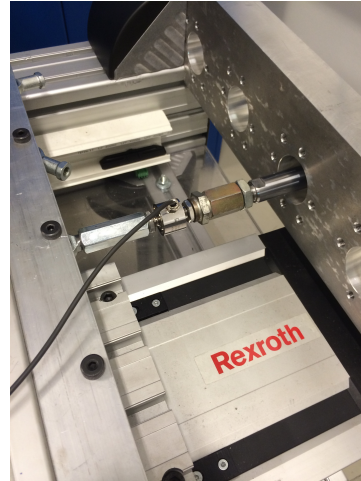


Figur 19: Sensor-inngang innside og signalforsterker

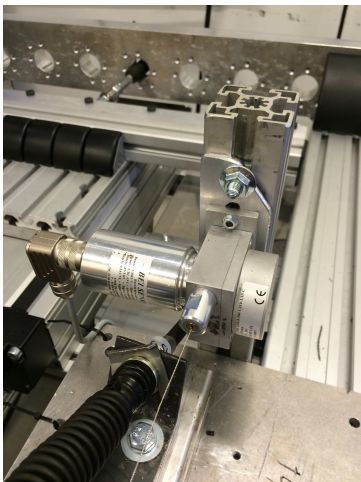
Den ene sensoren (I-698) som skulle ved servoen har mindre støy og krever ekstern forsterker (Clip AE 101). Denne forsterkeren er koblet til inngang 8 og krever egen strømtilførsel.



Figur 20: Posisjonsgiver for servo



Figur 21: Lastcelle for servo



Figur 22: Trådpotmeter ved girspak

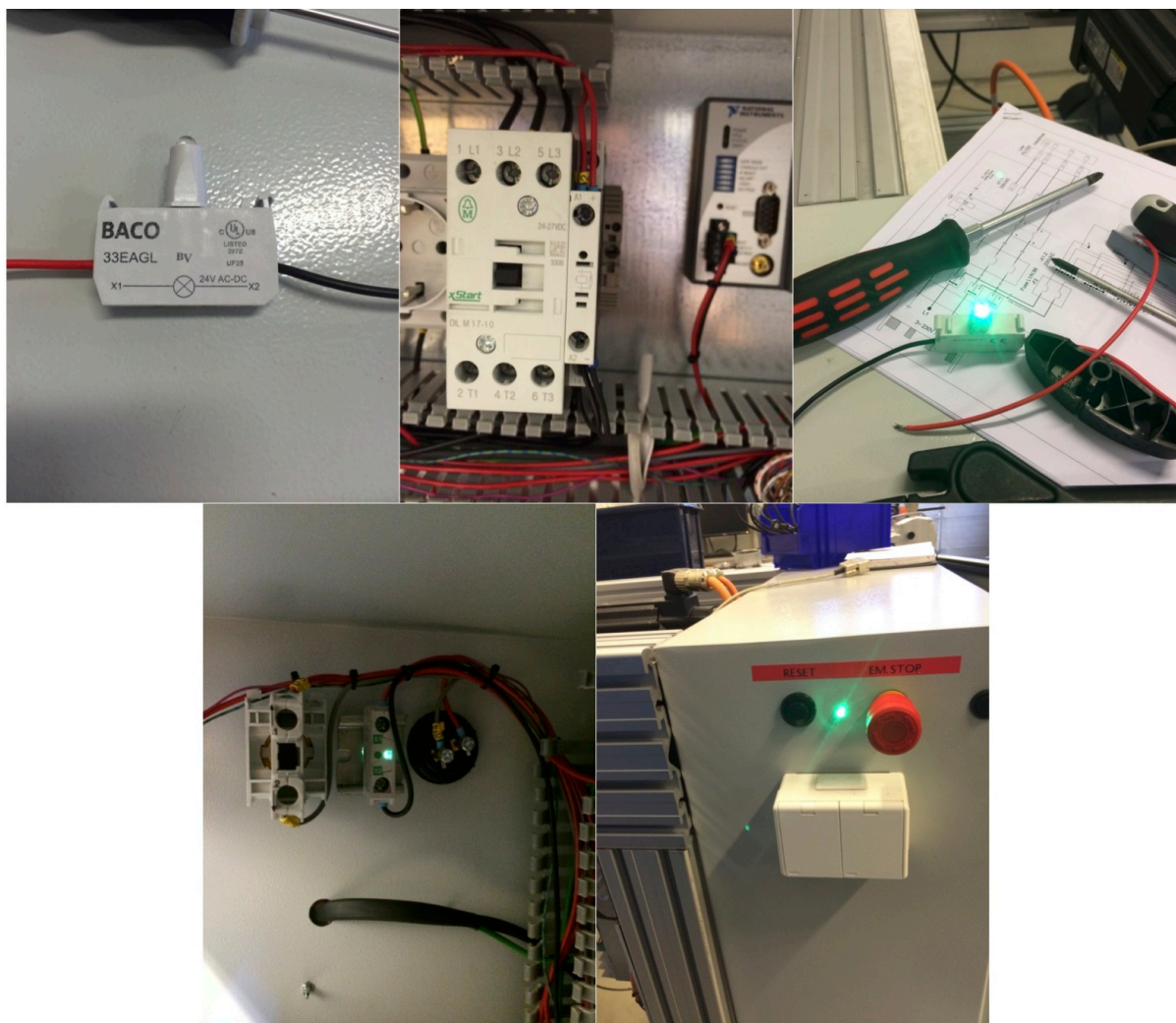
Posisjongiveren for servoen (Penny+Giles SLS130) er festet til profilen til riggen og profilen på servomotoren ved hjelp av en vinkel. Den følger bevegelsen til servomotoren og signalet sendes til LabVIEW og IndraWorks.

Lastcellen ved servoen (I-698) sitter mellom profilen på servoen og girkabelen. Den måler kraften nærmest servoen og signalet sendes til LabVIEW og IndraWorks.

Trådpotmeteret (BEI SENSORS CD50) er festet til selve girspaken, ved hjelp av et stativ for å få riktig vinkel på tråden. Den måler posisjonen til og signalet sendes til LabVIEW og IndraWorks.

2.1.4 – Reset-lampe

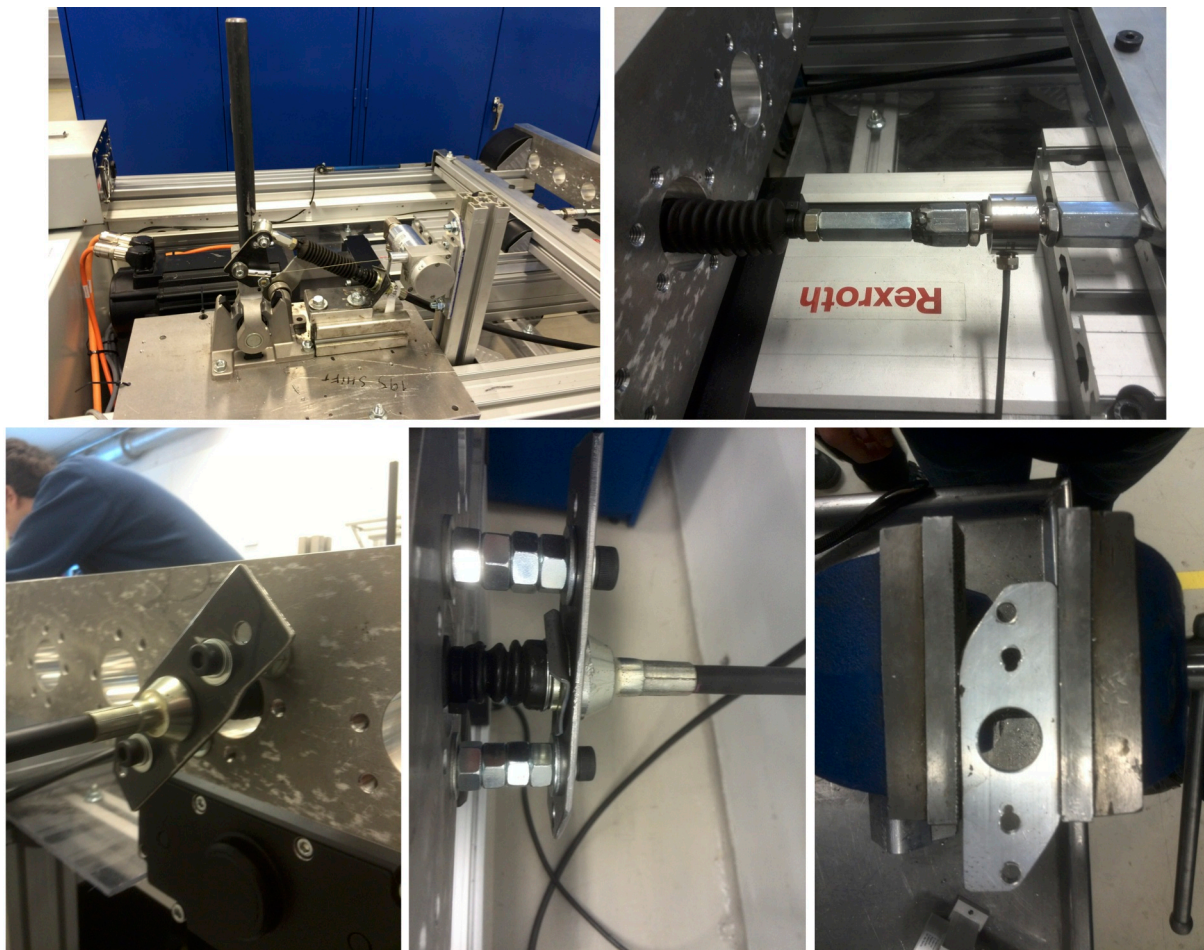
På grunn av at vi glemte å trykke på reset-knappen etter vi hadde trykket inn nød-stoppen, fant vi ut av vi skulle montere en lampe som lyste grønt når systemet var resatt og klar til bruk. Denne lampen er koblet på tidsreleet som igjen er koblet til reset-bryteren.



Figur 23: Tilkobling av reset-lampe

2.1.5 – Girspak

Vi skulle egentlig koble til en KA aktuator på servoen, men for å ha noe å kjøre mot, koblet vi først opp en luftsylinder, som skulle kobles til trykkluft for gjøre det enklere i første omgang. Før vi fikk koblet til trykkluften, fikk vi tak i girspaken vi egentlig skulle koble til, så da byttet vi til denne. Girspaken er koblet via en girwire rett til servoen, så vi må gire manuelt uten hjelp, til vi får reguleringen til å fungere ordentlig. Girkabelen har litt kortere vandrings enn spaken, så vi får ikke kjørt full giring enda. For å få festet kabelen, måtte vi lage en brakett, slik at vi fikk festet den til riggen. Kabelen ble så ført igjennom braketten til lastcellen og lastcellen ble så koblet til servoen.



Figur 24: Tilkobling av girspak

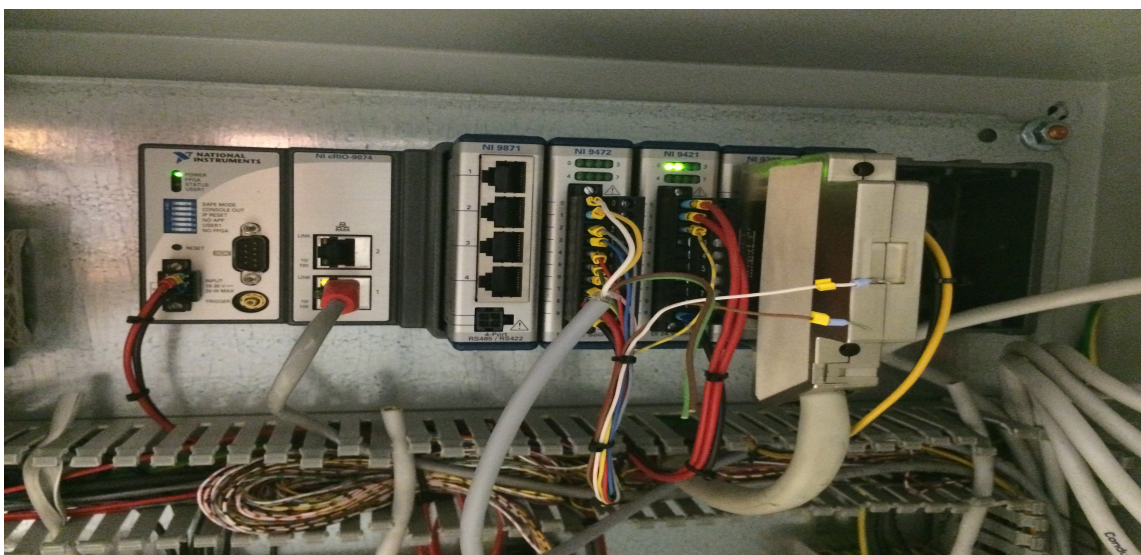
2.2 – LabVIEW

LabVIEW (Laboratory Virtual Instrument Engineering Workbench) er et grafisk programmeringsverktøy med i all hovedsak 2 deler, front-diagram og block diagram. Front-diagrammet viser et brukerinterface med knapper, grafer, lamper og lignende, og brukeren styrer programmet derfra. Block diagrammet er der selve programmeringen skjer. I denne oppgaven har LabVIEW vært en av de største delene og en nøkkelrolle for overvåking, logging, kraftberegning og kalibrering av sensorer. Programmet har blitt bygd fra bunn og alt er eget verk, bortsett fra kalibreringsdelen. Den har vi fått men vi har måtte gjøre store modifikasjoner for å få det til å funke til vårt program.

cRIO

CompactRIO (cRIO) er en embedded kontrollenhet fra National Instruments. Den er selve leddet mellom komponentene og PC. I denne oppgaven ble det brukt en cRIO-9074. I chassiset på denne enheten er det flere I/O moduler. NI 9871 (parallellport som ikke er i bruk for denne oppgaven), NI 9772 (Digitale utganger), NI 9421 (Digitale innganger), NI 9263 (Analoge utganger) og NI 9205 (Analoge innganger).

Kommunikasjon mellom LabVIEW og PC skjer ved hjelp av en ethernet-kabel på Link 1 porten på cRIO. IP og nettverksmaske er satt slik at kommunikasjonen fungerer. Det brukes seks analoge innganger for inputs fra sensorer, en analog utgang for å sende signal fra LabVIEW til MLD og en digital utgang for å sende et styresignal for start og stopp til MLD.



Figur 25: cRio med moduler

Lagring til fil og vise inputs på graf

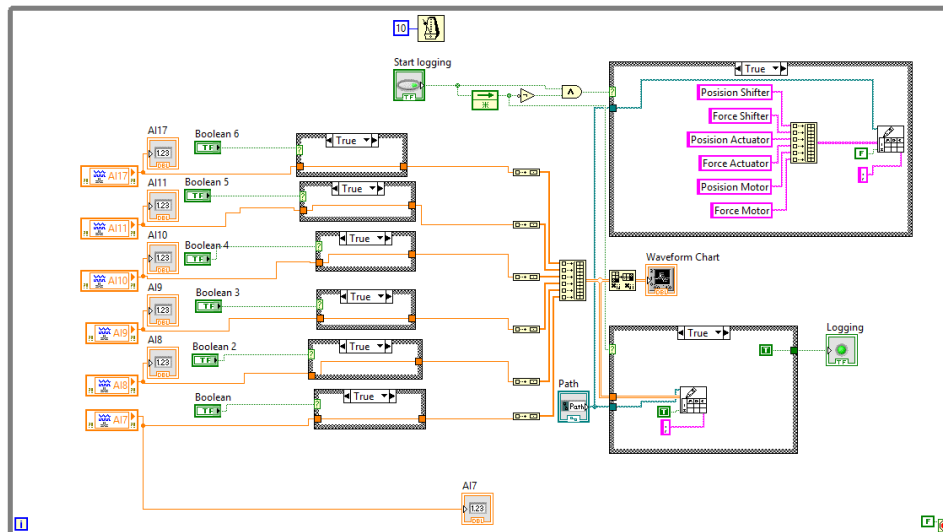
Sensorenes verdier skal lagres til en .csv fil (comma-separated values). Dette er for å logge hva slags krefter som har blitt brukt på ulike posisjoner. De analoge signalene kommer inn fra cRIO og vises i en numerisk indikator (display på front-diagram).

Alle signalene går igjennom en case-structure, der de får passere hvis argumentet er "true". Argumentet er en booleansk knapp i front-diagrammet som man kan huke av for å vise signalet (sette true). Hvis knappen ikke er huket av vil case-structure bli "false" og sender en konstant 0 i stedet.

Hvis signalene kommer igjennom sine case-structures, blir de omgjort til hvert sitt array. Alle separate arrays samles så inn og bygger et felles array, som nå blir et 2D array. Ut fra arrayet sendes signalet videre til en waveform chart, der de vises som grafer i front-diagrammet.

Logging aktiveres ved å trykke på knappen, og da settes to nye case-structures til true. I den ene skrives et string-array til en csv fil ved hjelp av "write to spreadsheet" funksjonen. Pathen er valgfri og skrives inn i front-diagrammet (den spør deg hvor du vil lagre hvis du ikke skriver noe). Denne case-structuren kjører kun en gang for man vil bare ha overskriftene printet en gang. For å få den til kun å kjøre en gang bruker vi en "feedback node".

Den andre case-structuren skal kjøre hele tiden, så lenge den er true. Den skriver kontinuerlig de samme verdiene som blir vist på grafen til samme fil som den andre case-structuren. Vi valgte å ha en egen knapp for logging fordi den skriver veldig fort og det er ikke nødvendig med alle verdier. Det lyser også en grønn lampe mens logging er aktiv.



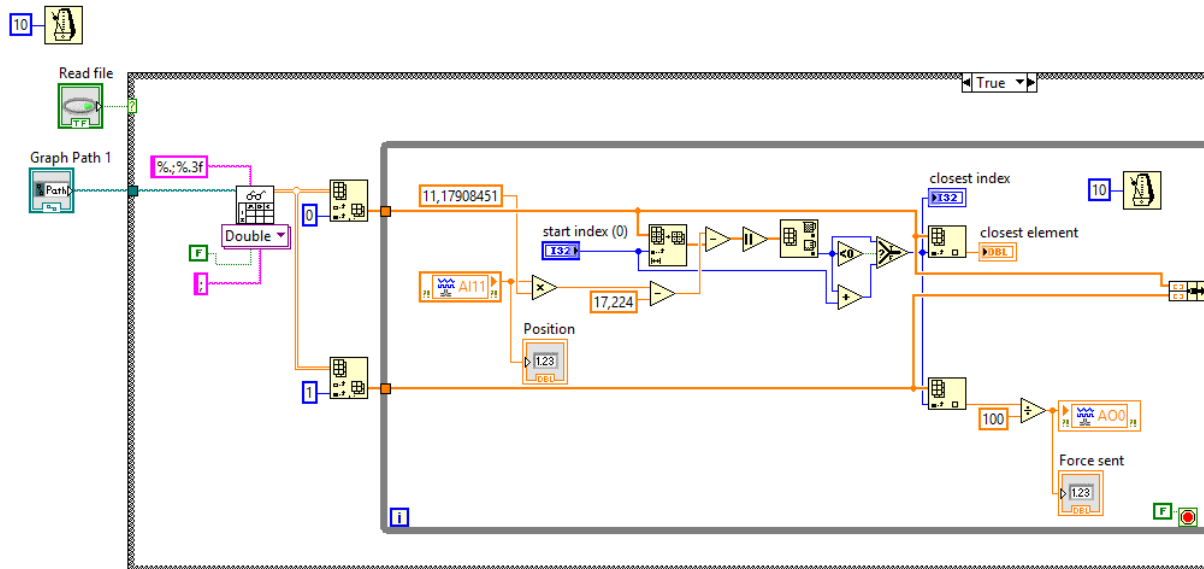
Figur 26: LabVIEW blockdiagram for å skrive til graf

Beregning av referansekraft

For å få en PID regulator i IndraWorks trengs to kraftmålinger, aktuell kraft og referansekraft. Referansekraften er den kraften vi målte på girkassen i starten av prosjektet. Denne kraften ligger i en csv fil på 2920 rader og to kolonner, en for kraft og en tilhørende posisjon (dette er en av to retninger for å få en komplett kurve). LabVIEW leser inn filen som double verdier og omdanner disse verdiene til et 2D array. Deretter splittes dette til to 1D array ved hjelp av "index array", et for posisjon og et for kraft. Arrayet for posisjon skal sammenlignes med aktuell posisjon fra posisjonssensor. Posisjonssensoren gir et signal mellom 0 og 2,84 V, mens posisjonsverdiene i csv filen er 14,5246 og -17,224 mm. For at signalet fra posisjonssensor skal få samme verdiområde som arrayet ganges først signalet med 11,17908451 og deretter trekkes det fra 17,224. Programmet prøver å finne en tilsvarende verdi i arrayet til det som kommer inn fra sensor, hvis den ikke finner en eksakt verdi så velges det nærmeste elementet. Indexen til dette arrayet, altså radnummeret blir da sendt til arrayet for kraft som sender kraftverdien på dette radnummeret videre. Verdiene som kan sendes mellom NI 9263 (analog output) og MLD må være mellom +10 og -10 V.

Kraftverdiene overstiger 120 så dette er skalert ned ved å dele verdien på 100 i LabVIEW og ganges opp igjen når verdien er i MLD. Alt dette skjer i en while-loop som ikke stopper med mindre tilhørende case-structure blir satt til false ved hjelp av knappen i front-diagrammet. Programmet kjører ikke før knappen blir trykket og setter case-strukturen til en true tilstand.

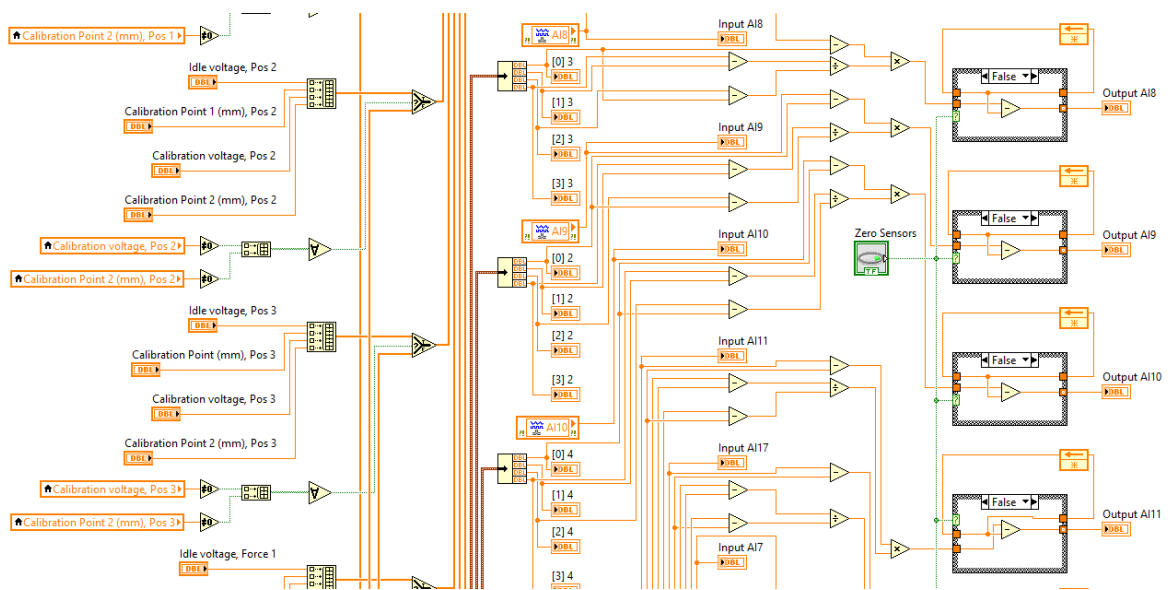
Verdiene fra csv filen skrives også direkte til en XY graf der man tydelig ser forholdet mellom posisjon og kraft fra vår tidligere måling av girkasse.



Figur 27 : Kode for å beregne referansekraft

Kalibrering av sensorer

Kalibreringsbiten av programmet er noe vi har fått og modifisert til vårt program. Poenget med denne biten er at vi kan kalibrere sensorene for at de skal få en nøyaktig måling. Dette er viktig fordi det er ganske små marginer i girgrafen, så det er kritisk at tallene stemmer. Det leses opp et kalibreringsark med default verdier til sensorene. Dette arket leses opp ved oppstart av programmet. Når man trykker "Update Parameters" vil de nye verdiene man har skrevet inn skrives over de gamle verdiene i kalibreringsdokumentet. Man kan også sette verdiene til null ved hjelp av "Zero Sensors". I skrivende stund fungerer ikke kalibreringen helt som den skal. Den leser opp fil og oppdaterer men det er noe feil med indexene. Dette er en liten feil men med tanke på tiden er dette ikke helt fikset.



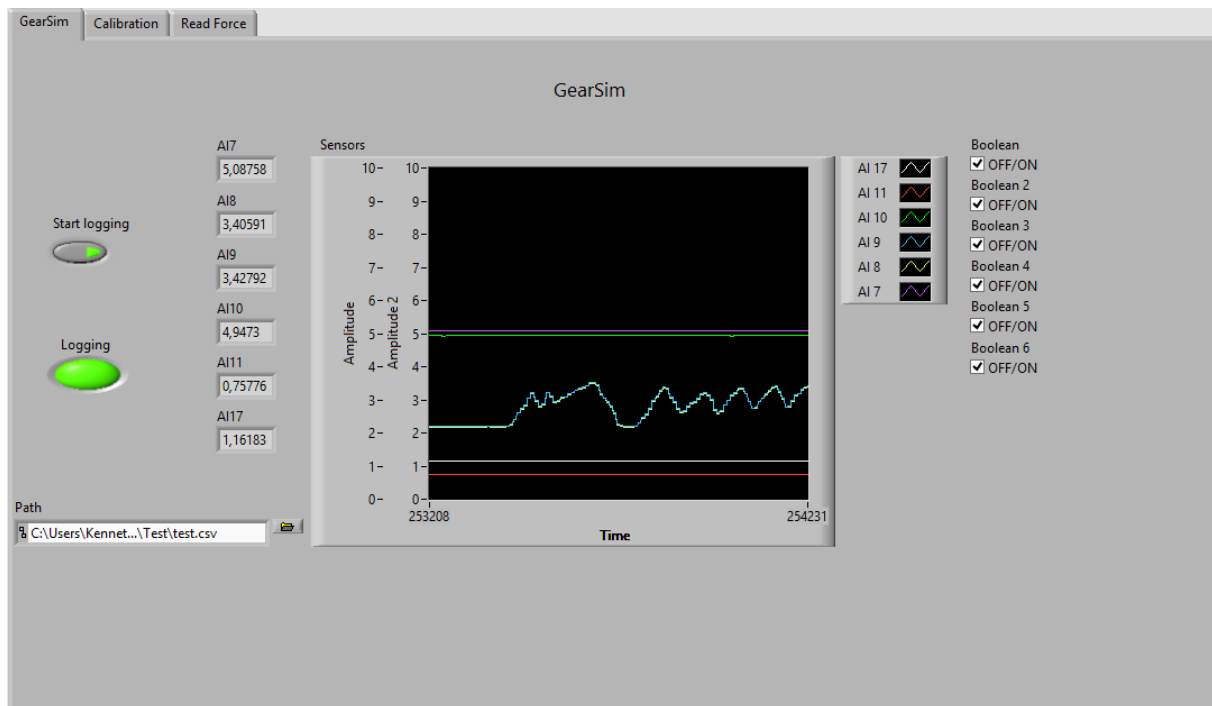
Figur 28 : Utdrag av kode for kalibrering av sensorer

Slik fungerer kalibreringen for eksempelvis med en lastcelle:

Man leser av hvor mange V det er ved 0 N og skriver inn verdien. Deretter tilfører man en last (typisk 1000 N ved en 2000 N lastcelle), og leser av hvor mange V det er ved denne lasten og skriver inn dette.

Interface

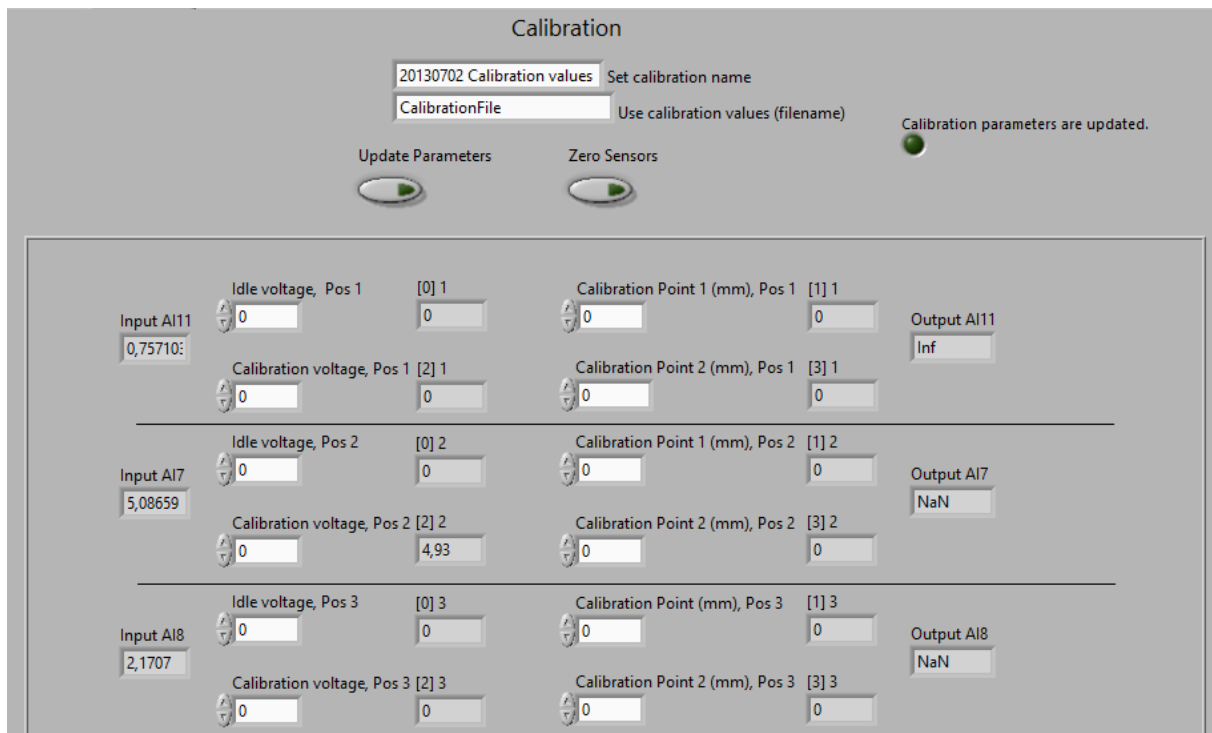
LabVIEW brukerinterface er ganske enkelt. Det blir laget i front diagrammet og med tilhørende kode i block diagrammet blir det et komplett program. I vårt program har vi en tabcontrol som gir oss tre forskjellige faner for å gjøre interfacet mer brukervennlig. Disse fanene er "GearSim", "Calibration" og "Read Force".



Figur 29 : Front diagram for GearSim

I "GearSim" fanen skjer:

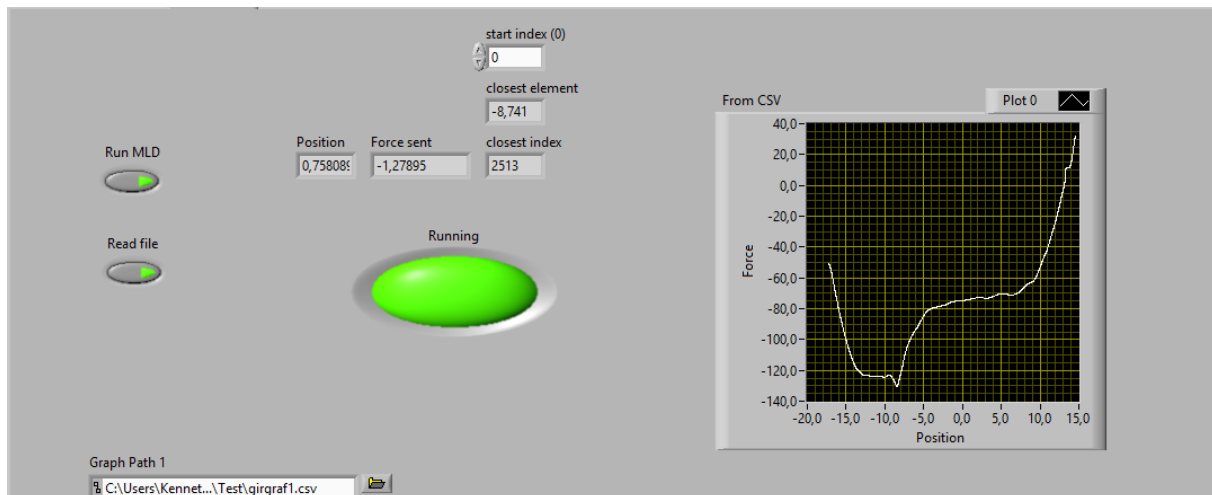
- Styrer hvilke analoge inputs (sensorer) man vil vise på charten.
- Numeriske indikatorer på verdiene fra analoge inputs.
- Knapp for å starte logging.
- Lampe som lyser mens logging er aktiv.
- Path for hvor loggefilen skal lagres.
- Selve charten.



Figur 30: En del av front diagrammet for Calibration

I "Calibration" fanen skjer:

- Setter verdier for sensorene i bestemt last.
- Path for hvor kalibreringsfilen skal leses fra.
- Knapp for å oppdatere verdier.
- Knapp for å nullstille sensorer.



Figur 31: Front diagrammet for Read Force

I "Read Force" skjer:

- Knapp for å sende et digitalt signal til MLD.
- Lampe som viser at det digitale signalet er satt høyt.
- Knapp som leser opp en girgraf og gjør det om til arrays.
- Indikatorer som viser aktuell posisjon, hvilken posisjon som er nærmest aktuell posisjon i csv fil, nærmeste index (radnummer) og hvilken kraft dette tilsvarer (kraften er delt på 100).
- En kontroller man kan plote inn hvilken index man vil starte søket etter nærmeste posisjon i csv fil.
- XY graf for hvordan grafen ser ut.

Vedlegg: Bilder av kode på CD.

2.3 – Test av sensorkommunikasjon

Denne testen gikk ut på å få kommunikasjon mellom sensorer og PC, om det måtte være IndraWorks eller LabVIEW. Vi har seks sensorer totalt som skulle kobles til. Tre lastceller og tre posisjonsgivere. I og med at vi ikke skal teste med aktuator i første omgang, har vi kun koblet opp to posisjonsgivere (på servo og girspak) og en lastcelle (på servo) på riggen. Vi testet disse sammen med en posisjonsgiver og en lastcelle løst ved siden av, for å sjekke om vi fikk riktig signal på alle inngangene.

Utstyr

Rigg: 144

PC med LabVIEW/IndraWorks

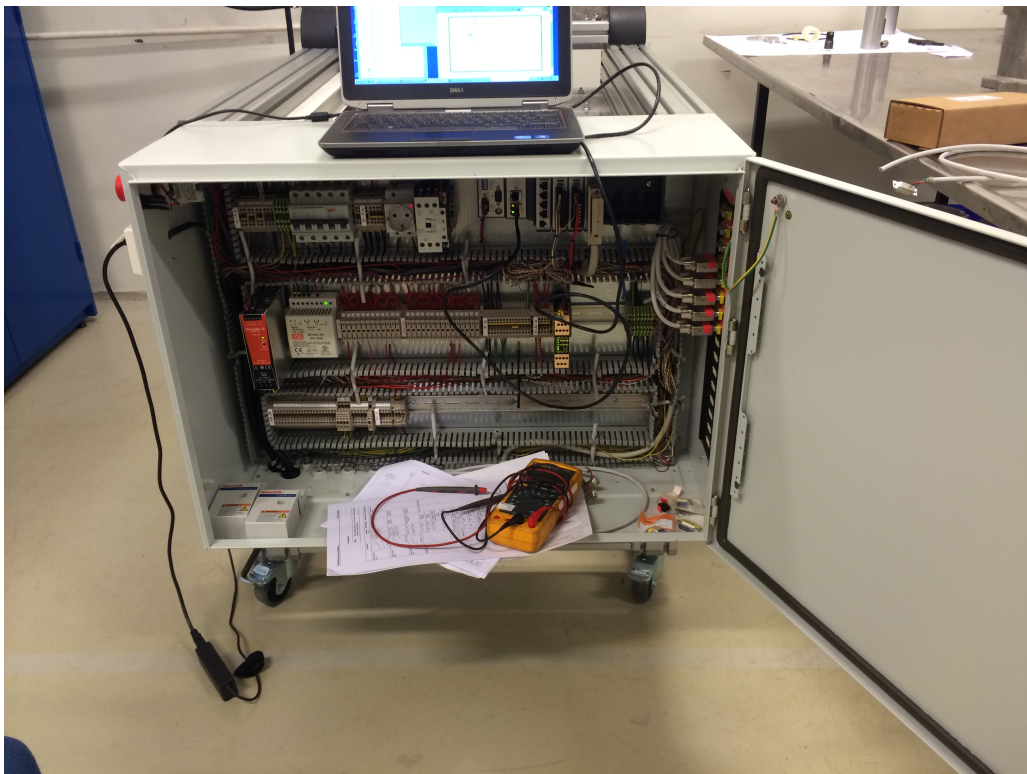
Lastceller: I-916 og I-698

Posisjonsgivere: Penny+Giles SLS130 og BEI SENSORS CD50 (trådpotmeter)

Koblingskontakter og bananplugger

Diverse ledninger

Diverse verktøy



Figur 32: Skap på rigg R-144

Fremgangsmåte

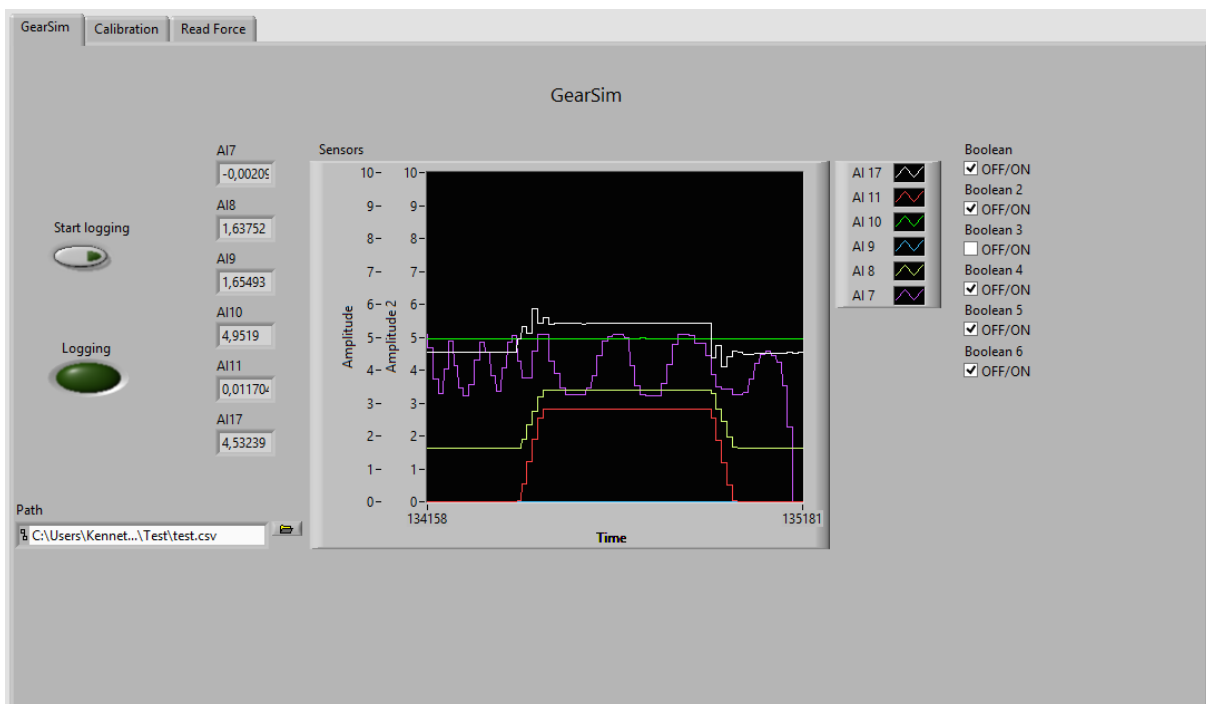
Før vi kunne starte testen oppsto det et problem med et av tilkoblingspunktene på riggen, og systemet hadde derfor ingen spenning. Dette førte til en del feilsøking med multimeter.

Alle kablene til sensorene er koblet til kontakter som passer rett inn på siden av skapet, slik at vi kan koble de lett til og fra. Disse kablene måtte vi lage selv, slik at det passet vårt system.

Signalene fra sensorene går inn på rekkeklemmer i skapet, deretter inn på analog input på cRio, som igjen er koblet til PC med LabVIEW.

Resultat

Når vi startet systemet beveget vi på sensorene og fikk utslag i LabVIEW ved hjelp grafer (se figur). På denne figuren er fem av seks sensorer koblet inn, fordi vi ikke hadde den siste tilgjengelig. For å få teste alle inngangene bytte vi bare mellom kontaktene. Sensorene gir et output-signal som måles i volt, og det er volt over tid som vises i grafen. Disse signalene brukes til å velge hvilken motkraft servoen skal gi avhengig av hvor den er. Alle sensorene fungerte som de skulle, og vi fikk lest av slik vi ville i LabVIEW. Inputs og outputs i LabVIEW er nærmere beskrevet i forrige kapittel.



Figur 33: Sensorgrafer i LabVIEW

2.4 – Test av IndraWorks

Introduksjon

Poenget med denne testen er å få kommunikasjon mellom MLD (CSH01) og PC med IndraWorks. I IndraWorks setter vi all parametere for hastighet, posisjon, akselerasjon og kraft. Dette vil foregå enten med «serial communication» via X2-porten eller med Ethernet via x22 eller x23 portene. Ethernet har den fordelen at det er det raskeste alternativet, derfor ønsker vi jo å bruke denne mest.

Utstyr

- PC med IndraWorks
- MLD
- Overgang til USB fra Serial Communications
- Ethernet kabel
- Posisjonssensor: Penny+Giles SLS130 og trådpotmeter
- Lastcelle: I-698

Fremgangsmåte

Først prøvde vi kun med Serial Communications og etter en god del lesing i manualene, fikk vi opprettet kommunikasjon. Problemene dukket opp mer hyppig når vi prøvde med Ethernet imidlertid. Vi klarte ikke å endre IP-adressen i programmet og heller ikke få kommunikasjon.

Panelet på MLDen skal også ha mulighet til å sette IP, nettverksmaske og default gateway. Nedenfor står prosedyren vi fulgte ut i fra en av manualene våre. Med denne metoden var det ingen mulighet til å sette noen av disse parameterne ut i fra vår erfaring. Når vi prøvde oss frem litt på egenhånd fant vi fram til der man kan skrive inn parametere for IP og så videre, men de ble ikke lagret.

- Slo på spenning til driver.
- Holdt «Esc» og «Enter» inne i ca. 8 sekunder.
- Bladde opp med tastene til vi fant «2. Comand» og trykket Enter.
- Valgte «2.3 Ethernet» og trykket Enter, hvor 2.3.1, 2.3.2 og 2.3.3 befant seg.

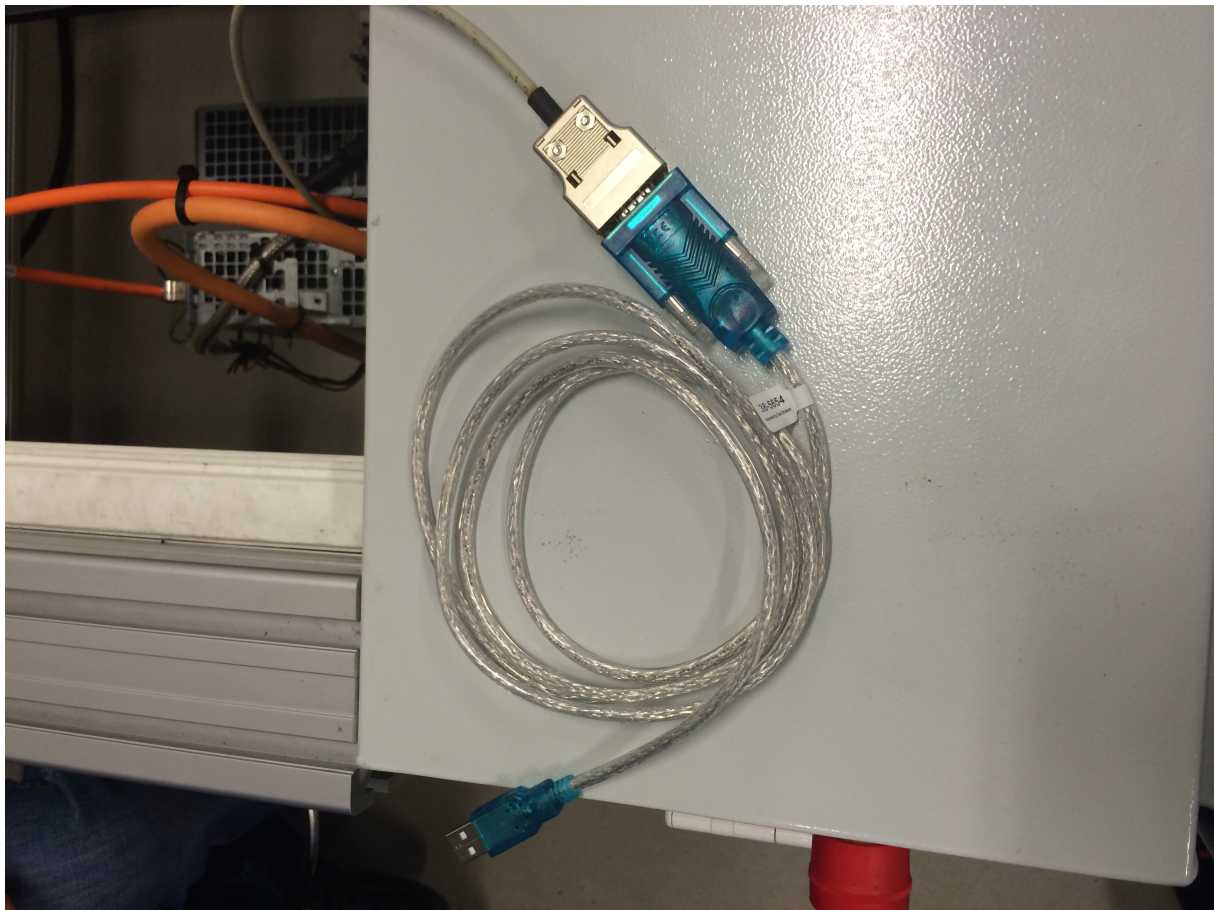
GCS – Gir & Clutchsimulator

Under «2.3.1 P-0-1531», «2.3.2 P-0-1532» og «2.3.3 P-0-1533» kan man i følge manualen (BoschRexroth - Indramotion MLD_Getting Started_Summary) sette IP-adresse, nettverksmaske og default gateway henholdsvis.

Her var det ikke mulig å skrive inn noen av disse.

Vi prøvde derfor å bytte til «parameter mode», ved å gå velge 2.11 > PM, men denne innstillingen sto MLDen allerede på.

Etterhvert fastslo vi at vi ikke kunne bruke mer tid på nettopp dette, og at vi heller kunne se mer på det om vi så skulle få tid. Vi har som sagt ikke fått noe krav om dette, men ville helst benytte det på grunn av hastighet.

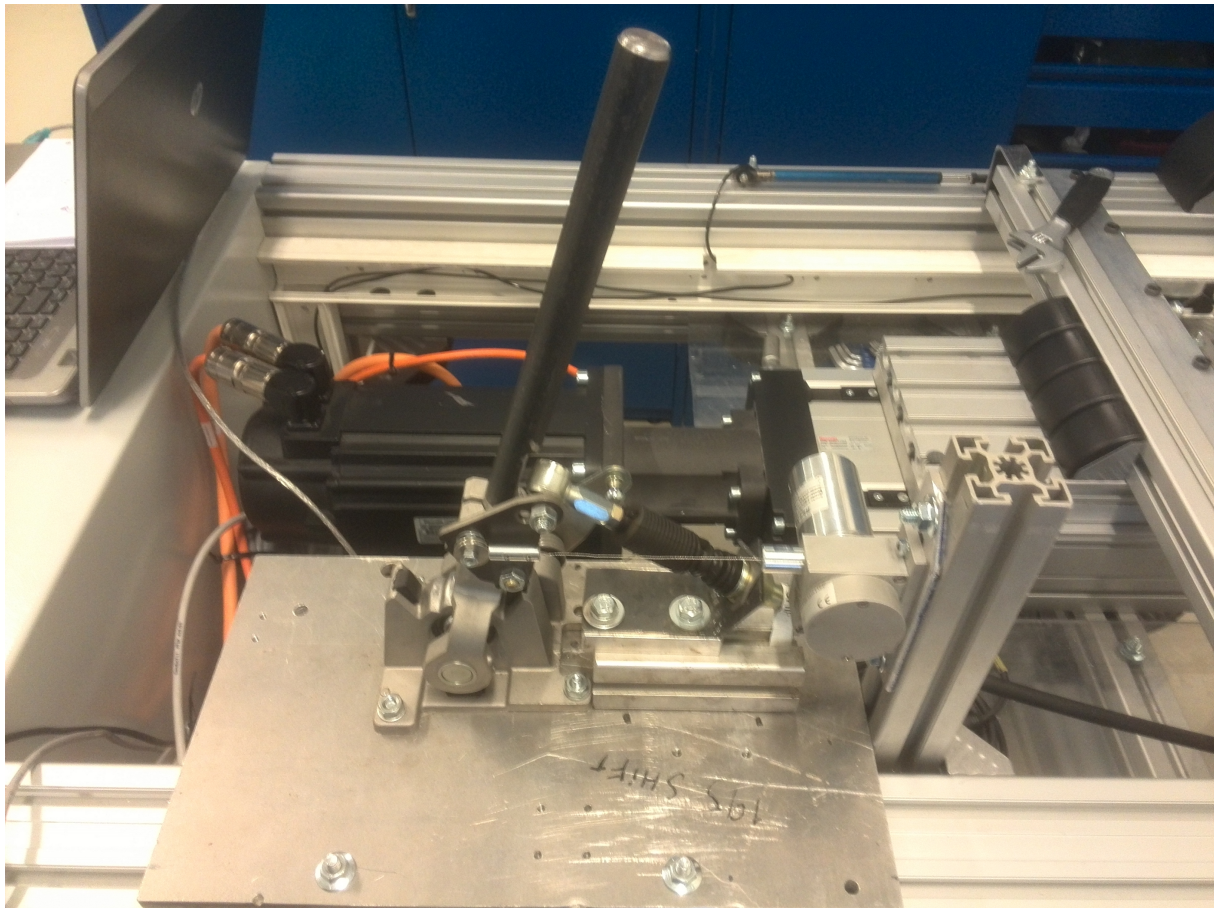


Figur 34: Overgang til USB fra Serial Communications

GCS – Gir & Clutchsimulator

Avlesning av «limit-values» fant sted først når luftsynderen var satt fast i et av hullene på braketten helt i enden av riggen. Total vandring ble satt til 4 cm eller 1440 grader, dette så langt som sylindren ville gå altså ± 2 cm. Under «Commissioning» i IndraWorks fantes «Easy Startup Mode», her var det mulig å ta små forflytninger av gangen med såkalt «Jogging». Denne sylindren ble etter hvert koblet fra og erstattet med en girspak.

Når girspaken var skrudd på plass og koblet til last cellen istedenfor luft-sylindren, var vi nødt til å lese av nye limit-values og et nytt nullpunkt i IndraWorks. Det foregikk på samme måte i «easy startup mode» hvor vi benyttet «Jog+» og «Jog-» fra 1,0 rpm til 20,0 rpm bare for å være veldig forsiktig.



Figur 35: Girspak med trådpotmeter koblet til.



Figur 36: Lastcelle festet til rigg og kabel fra girspaken

Resultater

I første omgang satte vi endepunktene til å være 1300 grader og -140 grader. Dette ga et midtpunkt på 580 grader.

Med girspaken koblet til endte vi opp på 971,14 grader i positiv retning og -1628,85 grader i negativ retning. Total vandring tilsvarer da 2600 grader som betyr at nullpunktet i forhold til spaken er -328,86 grader. Dette betyr igjen at total vandring i cm blir 7,22 cm, eller $\pm 3,61$ cm.

Disse verdiene ble endret til 960 grader i positiv retning og -1620 grader i negativ retning for å ha litt margin. Total vandring ble da 2580 grader eller 7,167 cm, altså $\pm 3,583$ cm.

2.5 – IndraWorks overordnet

Vi ønsker her å beskrive hvordan man med Rexroth softwaren "IndraWorks Engineering Tool" starter opp, lager et Nytt Prosjekt og finner driver-enhet. Når driver-enheten er funnet vil firmwaren pre- installert fra produsentens side, på bakgrunn av vår bestilling, fortelle IndraWorks hvilke muligheter, funksjoner og utvidelser vi har bestilt. Deretter beskrives hvordan innstillinger og parametere settes, på grunnlag av data funnet under IndraWorks oppstart beskrevet i Teknologidokumentet "Test av IndraWorks". Dette gjøres for å forberede og definere IndraWorks for å bli tatt over, og fullstendig kontrolleres av Servo Driverens PLS.

Utstyr

- PC med IndraWorks Engineering Tool
- MLD
- Serial Communications (RS232) to USB adapter.
- (Ikke alle PCer er utstyrt med RS232 I/O port)
- Ethernet kabel
- Posisjonssensor: Penny+Giles SLS130 og trådpotmeter
- Lastcelle: I-698

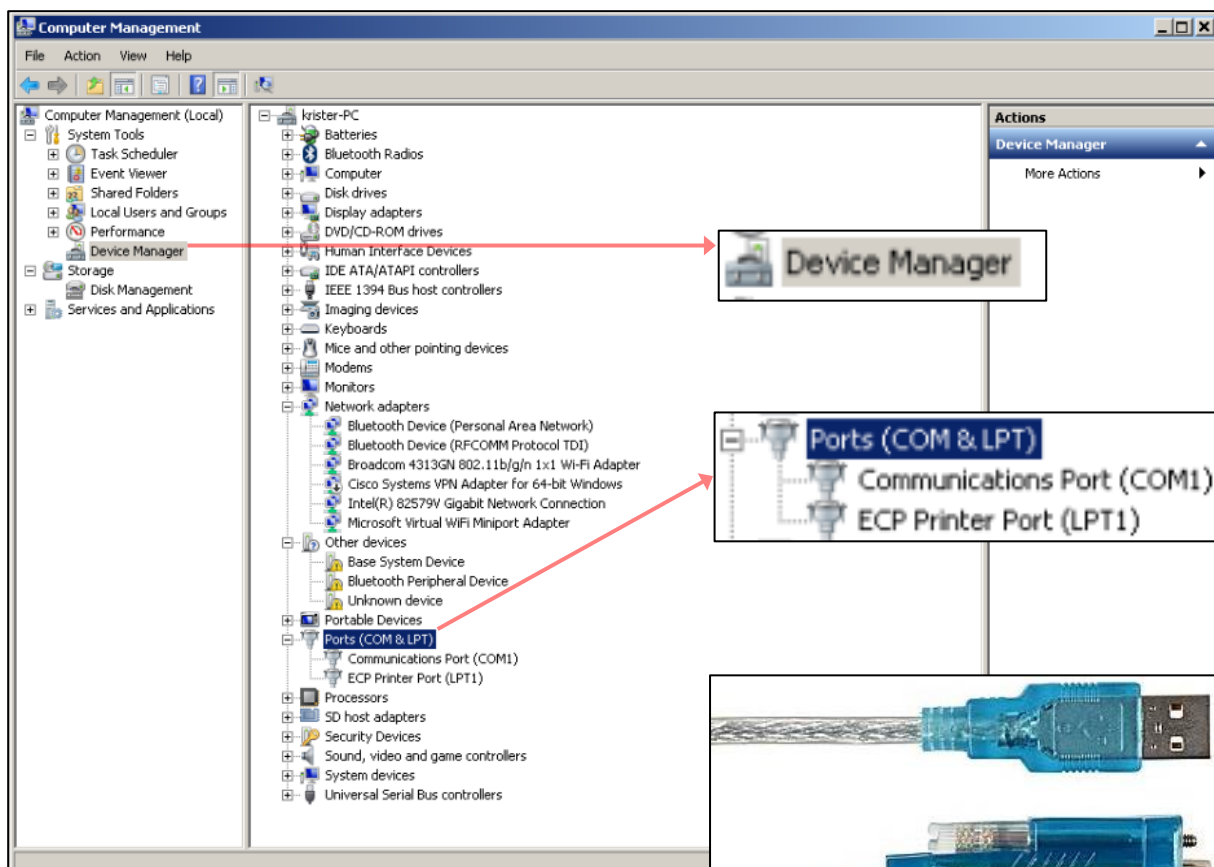
Resultater fra Teknologidokument: "Test av IndraWorks"

Med girspaken koblet til endte vi opp på 971,14 grader i positiv retning og -1628,85 grader i negativ retning. Total vandring tilsvarer da 2600 grader som betyr at nullpunktet i forhold til spaken er -328,86 grader. Dette betyr igjen at total vandring i cm blir 7,22 cm, eller $\pm 3,61$ cm.

Disse verdiene ble endret til 960 grader i positiv retning og -1620 grader i negativ retning for å ha litt margin. Total vandring ble da 2580 grader eller 7,167 cm, altså $\pm 3,583$ cm.

Fremgangsmåte

Serial Communications må kobles til USB port på PC, men før man kobler til er det viktig å legge merke til hvilken "Com-port" tilkobling blir tildelt. Dette gjøres i Windows7 ved å trykke "Start", deretter høyre-klikke "Computer". Velg "Management", så vil "Computer Management" åpne. Velg "Device Manager (kolonne med liste på venstre side av vinduet), deretter "Ports (Com & LPT)" –seksjon i lang liste over computerens innhold. Når du da setter i USB-Serial RS232 kabel, så vil det dukke opp en ny com-port med et nummer. Dette nummeret "COMx" brukes til å angi hvilken kanal IndraWorks vil kommunisere og utveksle data med MLD på. Hvilken nummer porten blir tildelt avhenger av hvilken USB-port man velger til adapter, antall USB-porter som HMI-computer er utstyrt med og hvor mange USB-porter som allerede er i brukt.

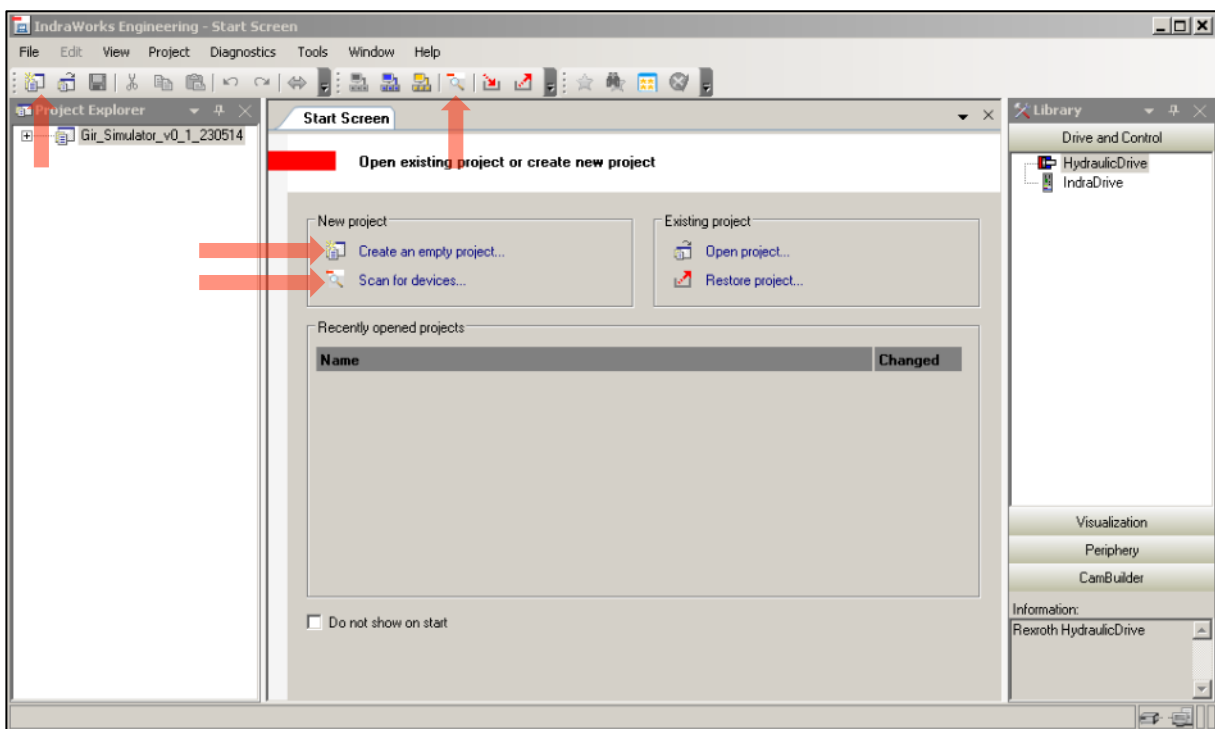


Figur 37: Computer Management \ Device Manager \ Ports

Figur 2: Overgang til USB fra Serial Communications

GCS – Gir & Clutchsimulator

- Slo på spenning til system ved å sette i kontakt, og vippet opp sikringene som var av.
- Trykket inn sort knapp på venstre side av styreskapet til systemet for å resette nødstop- krets. Grønn lampe kom på, og styrestrøms relét slo inn ved system resett.
- Startet "IndraWorks Engineering Tool".
- Lagde nytt prosjekt, og valgte ny prosjektmappe og norsk som språk. Følger veiviser.
- Trykket på "Scan for devices", og fulgte veiviseren.



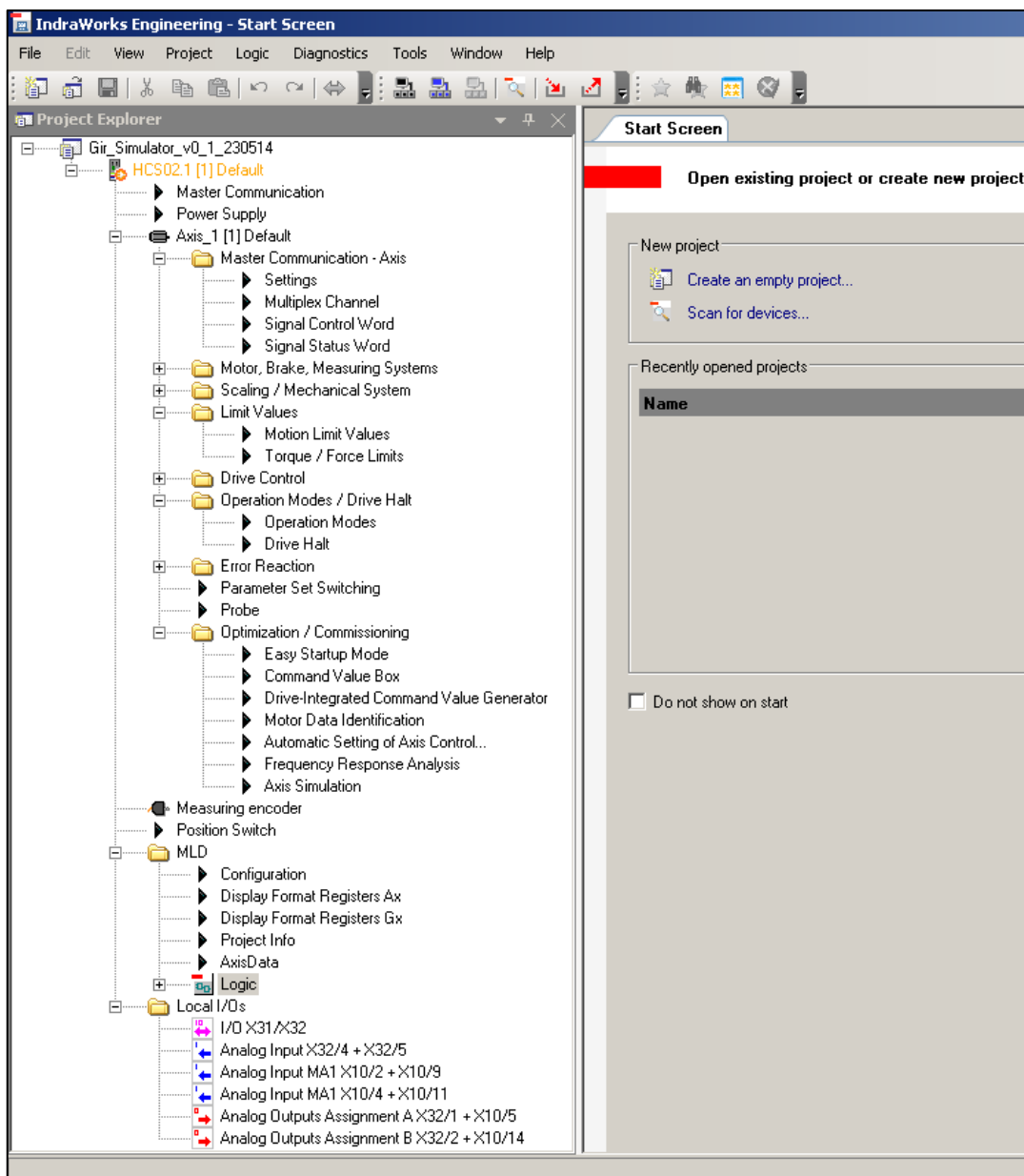
Figur 38: Overgnag til USB fra Serial Communications

- Valgte type system som er IndraDrive for vår del, med kommunikasjon over RS232, fulgte veiviser. Valgte også å merke søk på standard kommunikasjons-bus Sercos3, men Sercos3- bus bruker IP-adresser som må bli aktivisert i IndraWorks via seriellkommunikasjon først. Først da kan man finne utstyret via Ethernetkabel, og IP kommunikasjonskanalen på Sercos3-bus.
- Neste i veiviser for utstyrssøk valgte vi tildelt com nummer.
- Flytting av ønsket com-port til veiviser, gjøres ved bruk av pillene mellom valgboksene. Fra boks med samtlige com-porter funnet, til søke-boks for com-porter vi ønsket å søke etter utstyret på.

GCS – Gir & Clutchsimulator

Ved å hake av vår driver Power Section HSC02.1 og fullføre veileder. Blir Aksen som skal kontrolleres Axis1, MLD Control Section ved navn CSH01.1 og tilgjengelige Lokale innganger på disse lagt til prosjekt- browser. Dette fordi de komplette numrene til systemets komponenter i kombinasjon beskriv systemets kapabiliteter og parametere:

- Power Section: HSC02.1E-W0028-A-03-NNNN
- Control Section MLD: CSH 01.1C-S3-ENS-NNN-MA1-NN-S-NN-FW
- Firmware bestilt: FWA-INDR*-MPH-08VRS-D5-1-NNN-ML



Figur 39: IndraWorks Browser etter at driver er lagt til i prosjektet

Vi har i figuren IndraWorks Browser på forrige side uthevet de områdene som vi jobbet med under parametersettingen.

- Master Communications.
- Easy Start up mode.
- Command Value Box.
- Motion Limits.
- Torque/Force Limits.
- I/O interface X31/32
- Analoge innganger. Standard: X32.4/5, Extensions MA1: X10.2/10 & X10.4/11
- Axis Data
- Operation Modes

Lokale I/O:

- Få utslag på de analoge og digitale inngangene fra givere i systemet.
- Tildele signalene oppgaver.
- Skalere og forbehandle.

I/O interface X31/32:

- Navngi signal funksjonene med funksjonsparametere.

I/O konfigurasjon i MLD: (Synes ikke pga. offline bilde)

- Gi signalfunksjon en mottakerkode i PLS med signalbeskrivelse.

Axisdata:

- Motta og utføre signalfunksjon på bakgrunn av mottakerkoden ved hjelp av logikk gitt av PLS-program.

Master Communications:

- Tildele Sercos 3 IP og Gateway adresse for IP kommunikasjon.

Easy Startup mode & Command Value Box:

- For manuell kontroll, tilpassing og sekvensiell kjøring.

Motion & Torque/Force limits:

- Setter inn min. og max. posisjonsverdier aksene kan gå til i Motion Limits, verdier hentet fra Teknologidokumentet "Test av IndraWorks".
- Standard verdisetting, antagelig det motor parameterne tillater av overbelastning.

Operation Modes:

- Settes fra PLS med koder.
- Primary Operation Mode: Positioning block mode lagless, encoder 1.
- Secondary Operation Mode 1: Satt til Torque Control. PLC og driver overtar kontroll ved å aktivisere denne operasjonsmodusen under automatisk kjøring.

Liste over I/O, Signal & Parameter-koder

Navn:	Beskrivelse:	Signal Range:	Perifer I/O navn:	Assignment:	Signal koder:	Parameter koder:	I/O Bit
Styrestrøms-relé MLD	24V inn for aktivisering	24Vinn. 24Vut.	X31.1, X31.2	NA	NA	P-0-0861 Power Supply Status Word	9
Start/Stopp prosess signal fra LabVIEW	Enable / Disable TRUE / FALSE	Low- High / 0- 1	X31.3	NA	NA	P-0-1390 Input Word IB0	1
Kraft-celle	+10V max trykk-kraft. -10V max dra-kraft.	+10V / -10V	X31.4	A	S-0-0084 Torque / Force Feedback Value	P-0-0308	-
Referanse signal Kraft fra LabVIEW	Valg av referanse kraft fra tabell, REAL	NA	X10.4	NA	S-0-0080 Torque / Force Command Value	P-0-0038 Torque / Force Command Value	-
Posisjongiver Spak	Tråd potmeter Lite spennings-område.	NA	Ikke i bruk.	NA	NA	NA	-
Posisjongiver Servo-system	Stang-Potmeter Penny+Giles	0V - 3,143 V	X10.2	B	S-0-0048 Additive Position Command Value	P-0-0686 Additive Position Command	-
Tillegg:							
Position Block Mode Lagless, Encoder1	Primary Operation Mode	-	-	-	S-0-0032	-	-
Torque / Force Control	Secondary Operation Mode	-	-	-	S-0-0033	-	-

Tabell 1: Liste over I/O, signal og parameter-koder

2.6 – IndraLogic

IndraLogic er et komplett utviklingsverktøy for bruk til PLS. Her er det flere IEC61131 språk det er mulig å benytte, disse inkluderer;

IL (Instruction List) består av en rekke instruksjoner som alle har sin egen linje og inneholder en operator og, avhengig av hva slags type operasjon, en eller flere operander adskilt av kommandoer.

ST (Structural Text) består av flere instruksjoner som styres av «IF», «THEN», «ELSE» type setninger og «WHILE», «DO» looper. Dette språket sammen med FBD kommer vi til å bruke aler mest.

SFC (Sequential Function Chart) er et grafisk basert språk som gjør det mulig å beskrive den kronologiske rekkefølgen til forskjellige handlinger i et program.

FBD (Function Block Diagram) er også et grafisk programmeringsspråk. Det benytter en liste med nettverk hvor hvert nettverk inneholder en hvis struktur som representerer et logisk eller aritmetisk uttrykk, en funksjonsblokk, et «jump», eller en «return» instruksjon.

LD (Ladder Diagram) er også et grafisk programmeringsspråk som nærmer seg strukturen til en elektrisk krets. Det er nyttig for å konstruere logiske porter og kan også lage nettverk slik som i FBD. Det består av en rekke nettverk som er begrenset på venstre og høyre side av en vertikal «strøm-linje». I midten er det et krets skjema som består av såkalte «contacts», «coils» og linjer i mellom.

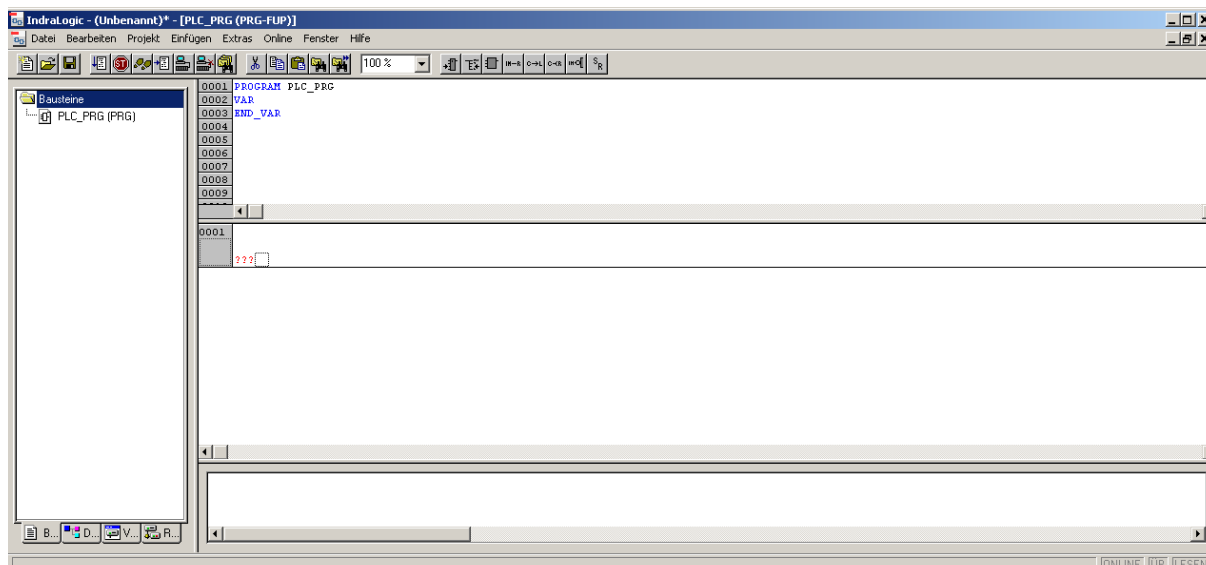
CFC (Continuous Function Chart) jobber ikke med nettverk slik som FBD for eksempel, men heller med fritt plasserbare elementer.

Disse kan igjen deles inn i flere såkalte «POU's». Det står for Project Organization Unit og er delt inn i tre forskjellige kategorier, da «program», «function block» og «function».

Fremgangsmåte

Når man er ferdig med parameter setting i IndraWorks. Kan man i MLD seksjonen av IndraWorks- browser se Indralogic ikonet. Dobbelt klikk på dette så vil tilleggs-softwaren IndraLogic åpnes, og man får spørsmål om hvilken måte hovedprogram skal skrives. Det er her man kan velge mellom forskjellige fremstillingsmåter av PLC program.

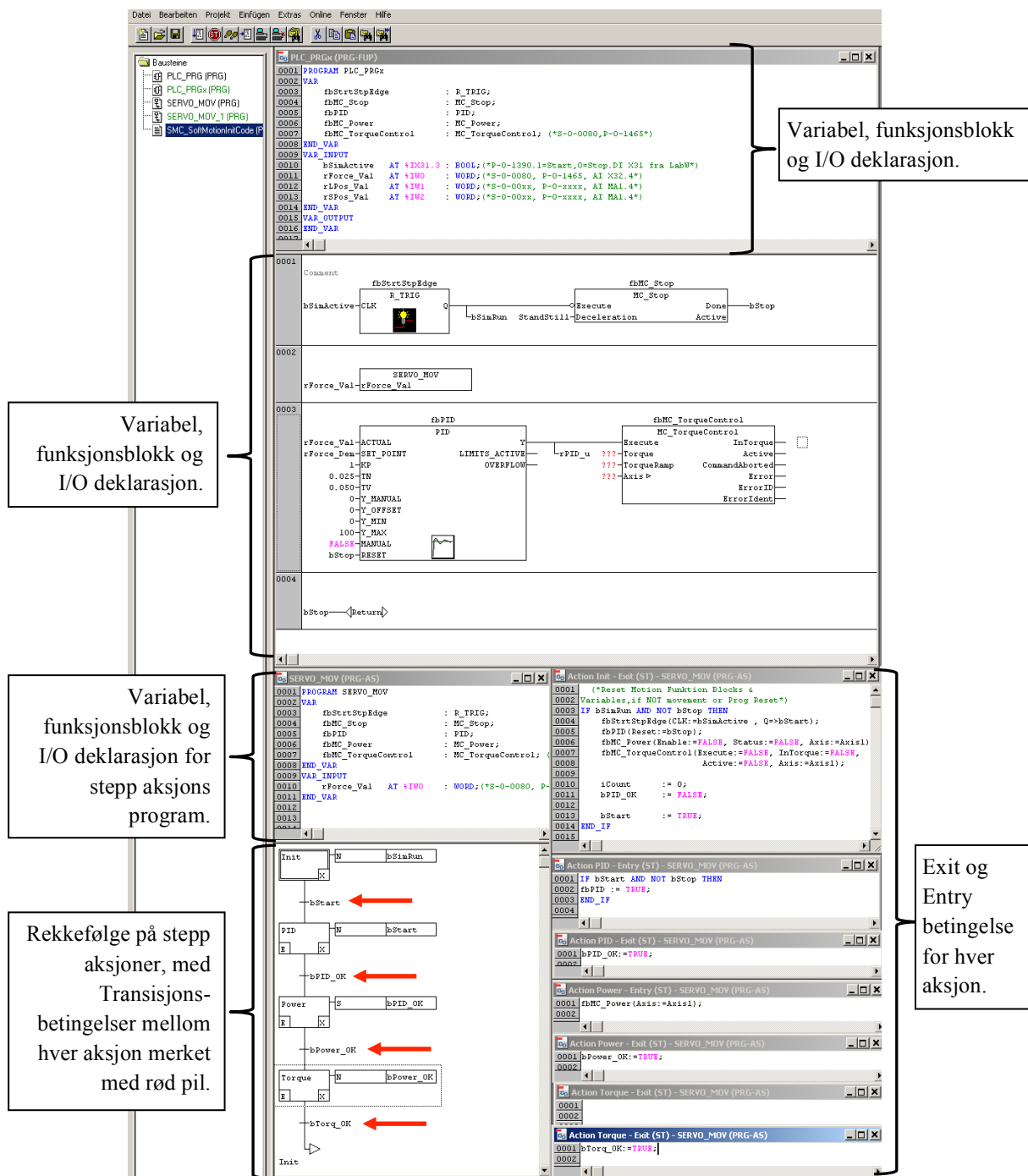
Fremstillingsmåtene er beskrevet på foregående side.



Figur 40 IndraLogic åpnet første gang, med standard funksjonsblokk oppsett på hovedprogram

Resultater

Her vises vårt første forsøk på å bygge PLC program, etter eksempel funnet i MLD Application Examples Manual. Vi gjorde om eksempel funksjonene til vårt behov. Her er standard funksjonsblokk Diagram (FBD) fremstilling valgt på hovedprogram. I underprogram er fremstilling valgt til Sekvensiell Funksjons Oversikt (SFC), men hver aksjons "Entry" og "Exit" betingelse i "SFC" er fremstilt i Strukturell Tekst (ST). Hvor vi gjorde om eksempel funksjoner tilpasset våre ønskede funksjoner. Vi syntes dette var vanskelig og tung måte å forstå logikk og bygge program på.

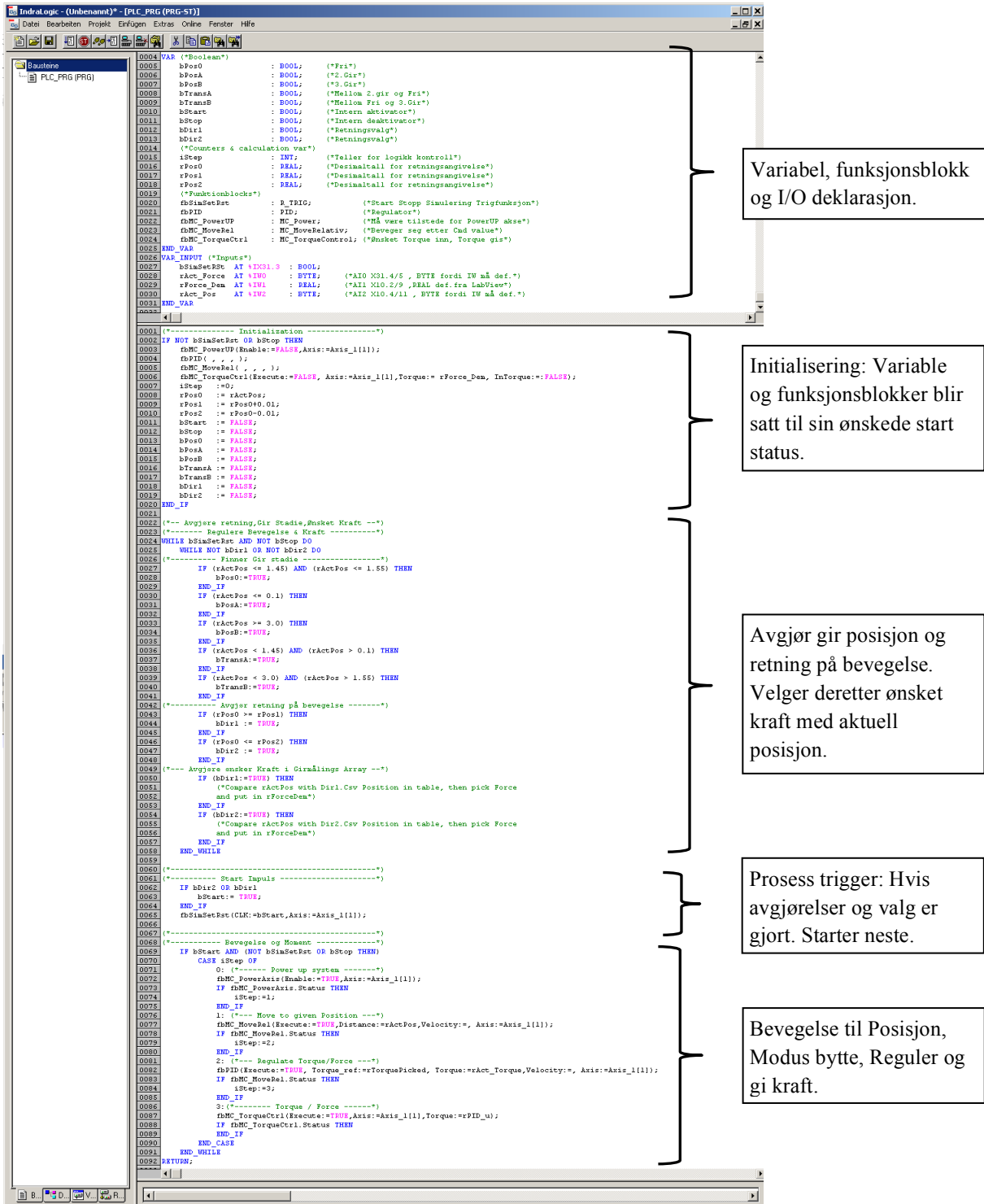


Figur 41: Første forsøk på PLC-program

Etter et nedslående første forsøk på PLC programmering, gikk vi løs på andre eksempler i håp om at forståelsen kom ved å øve. Det var da vi kom til et eksempel skrevet i ren Strukturell Tekst (ST), hvor logikken begynte å gi mening. ST-oppbygning av PLC program blir til slutt lik FBD-oppbygning, men med lite erfaring fra PLC ble blokkfremstillingen av logikk vanskeligere å forstå. Blokkfremstilling av program er uvant, men ved å ha mulighet til å

GCS – Gir & Clutchsimulator

skrive program i tekst med kjent syntaks fra programmering på andre plattformer ble logikken mulig å forstå. Strukturell Tekst program for vår applikasjon:



Variabel, funksjonsblokk og I/O deklarasjon.

```

0004 VAR ("Boolean")
0005   bPos0      : BOOL; ("Fri")
0006   bPosA      : BOOL; ("2. Gir")
0007   bPosB      : BOOL; ("3. Gir")
0008   bTransA    : BOOL; ("Mellom 2. gir og Fri")
0009   bTransB    : BOOL; ("Mellom Fri og 3. Gir")
0010   bStart      : BOOL; ("Inntre aktivator")
0011   bStop      : BOOL; ("Inntre deaktivator")
0012   bDir1      : BOOL; ("Retningsvalg")
0013   bDir2      : BOOL; ("Retningsvalg")
0014   ("Counters & calculation var")
0015   iStep      : INT; ("Teller for logikk kontroll")
0016   rPos0      : REAL; ("Desimaltall for retningsangivelse")
0017   rPos1      : REAL; ("Desimaltall for retningsangivelse")
0018   rPos2      : REAL; ("Desimaltall for retningsangivelse")
0019   ("Funktionblokker")
0020   fbSimSetRst : F_TRIG; ("Start Stepp Simulering Trigfunksjon")
0021   fbPID       : PID; ("Regulator")
0022   fbMC_PowerUP : MC_Power; ("Må være tilstede for PowerUP aksje")
0023   fbMC_MoveRel : MC_MoveRelativ; ("Beveger seg etter Cad value")
0024   fbMC_TorqueCtrl : MC_TorqueControl; ("Brenker Torque inn, Torque gir")
0025 END_VAR
0026 VAR_INPUT ("Inputs")
0027   bSimSetRst AT I1X31.3 : BOOL;
0028   rAct_Force AT I1V0 : BYTE; ("AI0 X31.4/5 , BYTE fordi IW må def.")
0029   rForce_Dem AT I1V1 : REAL; ("AI1 X10.2/9 , REAL def.fra LabView")
0030   rAct_Pos AT I1V2 : BYTE; ("AI2 X10.4/11 , BYTE fordi IW må def.")
0031 END_VAR
0032
0001 (*----- Initialization -----*)
0002 IF NOT bSimSetRst OR bStop THEN
0003   fbMC_PowerUP(Enable:=FALSE,Axis:=Axis_1[1]);
0004   fbPID( , , );
0005   fbMC_MoveRel( , , );
0006   fbMC_TorqueCtrl(Execute:=FALSE, Axis:=Axis_1[1],Torque:= rForce_Dem, InTorque:=FALSE);
0007   iStep := 0;
0008   rPos0 := rActPos;
0009   rPos1 := rPos0+0.01;
0010   rPos2 := rPos0-0.01;
0011   bStart := FALSE;
0012   bStop := FALSE;
0013   bPos0 := FALSE;
0014   bPosA := FALSE;
0015   bPosB := FALSE;
0016   bTransA := FALSE;
0017   bTransB := FALSE;
0018   bDir1 := FALSE;
0019   bDir2 := FALSE;
0020 END_IF
0021
0022 (*-- Avgjøre retning,Gir Stadie,Ønsket Kraft --*)
0023 (*----- Reguleres Bevegelse & Kraft -----*)
0024 WHILE bStart AND NOT bStop DO
0025   WHILE NOT bDir1 OR NOT bDir2 DO
0026     (*----- Finnes Gir stadie -----*)
0027     IF (rActPos <= 1.45) AND (rActPos <= 1.55) THEN
0028       bPos0:=TRUE;
0029     END_IF
0030     IF (rActPos <= 0.1) THEN
0031       bPosA:=TRUE;
0032     END_IF
0033     IF (rActPos >= 3.0) THEN
0034       bPosB:=TRUE;
0035     END_IF
0036     IF (rActPos < 1.45) AND (rActPos > 0.1) THEN
0037       bTransA:=TRUE;
0038     END_IF
0039     IF (rActPos < 3.0) AND (rActPos > 1.55) THEN
0040       bTransB:=TRUE;
0041     END_IF
0042     (*----- Avgjør retning på bevegelse -----*)
0043     IF (rPos0 >= rPos1) THEN
0044       bDir1 := TRUE;
0045     END_IF
0046     IF (rPos0 <= rPos2) THEN
0047       bDir2 := TRUE;
0048     END_IF
0049     (*-- Avgjøre ønsket Kraft i Cirralls Array --*)
0050     IF (bDir1:=TRUE) THEN
0051       ("Compare rActPos with Dir1.Csv Position in table, then pick Force
0052       and put in rForceDem")
0053     END_IF
0054     IF (bDir2:=TRUE) THEN
0055       ("Compare rActPos with Dir2.Csv Position in table, then pick Force
0056       and put in rForceDem")
0057     END_IF
0058   END_WHILE
0059 END_WHILE
0060
0061 (*----- Start Impuls -----*)
0062 IF bDir2 OR bDir1
0063   bStart:= TRUE;
0064 END_IF
0065 fbSimSetRst(CLK:=bStart,Axis:=Axis_1[1]);
0066
0067 (*----- Bevegelse og Moment -----*)
0068 IF bStart AND (NOT bSimSetRst OR bStop) THEN
0069   CASE iStep OF
0070     0: (*----- Power up system -----*)
0071       fbMC_PowerAxis(Enable:=TRUE,Axis:=Axis_1[1]);
0072       IF fbMC_PowerAxis.Status THEN
0073         iStep:=1;
0074       END_IF
0075     1: (*-- Move to given Position --*)
0076       fbMC_MoveRel(Execute:=TRUE,Distance:=rActPos,Velocity:=, Axis:=Axis_1[1]);
0077       IF fbMC_MoveRel.Status THEN
0078         iStep:=2;
0079       END_IF
0080     2: (*-- Regulate Torque/Force --*)
0081       fbPID(Execute:=TRUE, Torque:=rAct_Torque,Velocity:=, Axis:=Axis_1[1]);
0082       IF fbMC_MoveRel.Status THEN
0083         iStep:=3;
0084       END_IF
0085     3: (*----- Torque / Force -----*)
0086       fbMC_TorqueCtrl(Execute:=TRUE,Axis:=Axis_1[1],Torque:=rPID_u);
0087       IF fbMC_TorqueCtrl.Status THEN
0088         iStep:=4;
0089       END_IF
0090   END_CASE
0091 END_WHILE
0092 RETURN;

```

Initialisering: Variable og funksjonsblokker blir satt til sin ønskede start status.

Avgjør gir posisjon og retning på bevegelse. Velger deretter ønsket kraft med aktuell posisjon.

Prosess trigger: Hvis avgjørelser og valg er gjort. Starter neste.

Bevegelse til Posisjon, Modus bytte, Reguler og gi kraft.

Figur 42: Strukturell tekst

2.7 – Regulering

Vi søkte en god del på internett når det gjaldt hvordan angripe og beregne regulerings-kretsen. Vi fant ut at i Mathlab med de inkluderte verktøyene Simulink og Simscape, har vi mulighet til å simulere «open-loop» responsen til nærmest alle typer motorer.

Ved hjelp av Simscape kan man plotte grafer som viser responstiden til Hastighet,

Posisjon og Moment (Torque), ved å parametrisere motor i Simscape modellen. Disse «open-loop» grafene kan vi igjen benytte for beregning av regulerings-systemet.

2.7.1 – Parametere

Motor MSK076C Parametere:

MSK 076C fra BoschRexroth som vi allerede har tilgang på, er en Servomotor av typen «3-phase permanent magnet motor». Beskrivelser og informasjon om denne typen servomotor finnes i mappen «Støttedokumentasjon», som er en undermappe av «20 Regulering av løsning» . Dokumentnavnet er «Description and Basics of 3-ph PM Servomotors.pdf».

Electrical data						
Type	Maximum speed	Torque at standstill	Maximum torque	Continuous current at standstill	Maximum current	Moment of inertia
	n_{Max} [1/min]	M_0 [Nm]	M_{Max} [Nm]	I_0 [A]	I_{Max} [A]	J [kgm ²]
MSK076C-0300	4700			7,2	32,4	
MSK076C-0450	5000	12	43,5	12,2	54,9	0,0043

All specifications relate to the basic version of the motor with encoder S1 and without holding brake.
 1) At 750V DC bus voltage.
 2) Values without/with holding brake.

Figur 43: Utdrag av MSK076C Technical data



Figur 44: Deklarasjonsmerke på KA's servomotor


Parametere lest av Motormerke:		Verdi:	Enhet:
Torque Constant	- $K_m = K_t$	1,14	Nm / A
Proporsjonalitetsfaktor	- $K_E(\text{eff})$	70,5	V / min ⁻¹
Maximum speed	- n_{max}	5000	min ⁻¹
Maximum Torque	- $M_{d0\ 60k} = M_{\text{max}}$	12	Nm
Maximum Current	- $I_{d0} = I_{\text{max}}\ 60k(\text{eff})$	12,2	A
Servo Motor Type: 3-Phase Permanent Magnet Motor			
Motor Control Signals: VPWM (Variable(frequency) Pulse Width Modulation)			

Figur 45: Parametere lest av motormerke

GTE160 Gir med transmisjonsratio 1:3

Economical and compact

- ▶ Ideal for standard applications
- ▶ Low backlash
- ▶ Open choice of mounting positions
- ▶ Silent operation
- ▶ Lifetime lubrication



The simple step-by-step guide to ordering your GTE planetary gearboxes

GTE060-NN1-004B-NN03

Gears

- ▶ Size (e.g. "040")
- ▶ Gear range
- 1 - Single stage
- 2 - Two stage
- ▶ Output shaft and backlash
- A - With keyway
- B - Plain shaft

Motor/gearbox combination

Motor	GTE40	GTE60	GTE90	GTE120	GTE160
MSK	030	-	NN02	NN02	-
	040	-	NN03	NN03	-
	043	-	NN03	NN03	-
	050	-	-	NN20	NN20
	060	-	-	NN21 ¹⁾	NN21
	061	-	-	NN05 ¹⁾	NN05
	070	-	-	-	NN16
	071	-	-	-	NN16
	075	-	-	-	NN16
	076	-	-	-	NN06
	100	-	-	-	NN09 ¹⁾
MSM	041	-	NN03	NN03	-
	061	-	-	NN05 ¹⁾	NN05
	071	-	-	-	NN16
	076	-	-	-	NN06
MSM	019A	NN45	-	-	-
	019B	NN45	-	-	-
	021B	-	NN42	-	-
	021C	-	-	NN43	-
	041B	-	-	NN44	NN44

¹⁾ Combination only possible with single stage gearboxes

Gear	Transmission ratio	Maximum average input speed		Maximum input speed ⁽¹⁾	Maximum output speed ⁽¹⁾	Nominal input torque ^(1,2,3)	Nominal output torque ^(1,2,3)	Maximum input torque	Maximum output torque ^(1,2,3,4)	Backlash	Torsion stiffness	Efficiency ⁽²⁾	Mass moment of inertia	Weight	
		at 50% M OUT norm ⁽¹⁾	at 100% M OUT norm ⁽¹⁾												
Stage	i	[rpm]	[rpm]	[rpm]	[rpm]	[Nm]	[Nm]	[Nm]	[Nm]	[arcmin]	[Nm/arcmin]	[%]	[kgcm ²]	[kg]	
GTE160	Single-stage	3	1,350	900	6,500	2,167	133.3	400	213.3	640	< 6	38	98	12,14	18
		4	1,450	1,000	6,500	1,625	112.5	450	180	720	< 6	38	98	7,78	18
		5	1,650	1,150	6,500	1,300	90	450	144	720	< 6	38	98	6,07	18
	Two-stage	8	2,150	1,550	6,500	813	56.3	450	90	720	< 6	38	97	4,63	18
		12	1,550	1,000	6,500	542	66.7	800	106.7	1,280	< 10	41	96	12,37	22
	20	2,050	1,400	6,500	325	40	800	64	1,280	< 10	41	96	6,65	22	
	40	2,950	2,300	6,500	163	17.5	700	28	1,120	< 10	41	94	5,28	22	

Figur 46: Parametere gir

CKK 25-200 Lineær omformer

Technical data											
General technical data											
Load capacities and moments											
Size	Number of carriages	Ball screw $d_s \times P$	Dynamic load capacity C (N)			Dynamic moments		Planar moment of inertia		Maximum length L_{max} (mm)	Moved mass m_b (kg)
			Guideway	Ball screw	Fixed bearing	M_t (Nm)	M_L (Nm)	I_y (cm ⁴)	I_z (cm ⁴)		
CKK 25-200	1	32 x 5	55000	21500	26000	3570	540	612.00	3008.0	2200 (with SPU 5500)*	3.18
		32 x 10		31700							
		32 x 20		19700							
		32 x 32		19500							
	2	89340	21500	26000	5800	7810	612.00	3008.0	2200	5.20	
	($l_m = 175$ mm)	32 x 10		31700						2200 (with SPU 5500)*	
		32 x 20		19700							
		32 x 32		19500							

l_m = Center-to-center distance of carriages * = See section "Screw support for CKK 25-200" for lengths of 2,200 to 5,500

Maximum permissible loads						
Size	Number of carriages	Maximum permissible forces (N)			Maximum permissible moments (Nm)	
		F_{z1max}	F_{z2max}	F_{umax}	M_{tmax}	M_{Lmax}
CKK 25-200	1	42 200	21 100	12 230	1 372	209
	2	68 550	34 270	19 880	2 228	2 999

Acceptable loads
(recommended from experience)

With respect to the desired service life, loads up to about 20% of the characteristic dynamic values (C , M_t , M_L) have proven to be acceptable.

At the same time, the following may not be exceeded:

- maximum permissible loads,
- permissible drive torque,
- permissible travel speed.

Figur 47: Parametere lineær omformer

2.7.2 – Parametere, Beregninger & Beskrivelse av Servo-system

Motor MSK076C parametere:

$n_{\max} = n_m = 5000 \text{ min}^{-1}$; Rotasjonshastighet

$M_{\max} = M_m = 12 \text{ N}_m$; Torque (kraftmoment)

$J = J_m = 0,0043 = 4,3 \cdot 10^{-3} \text{ kgm}^2$; Treghetsmoment

$J_{Br} = 360 \cdot 10^{-6} \text{ kgm}^2$; Motor bremsemoment

Gir GTE160 parametere:

$i = 3$; Transmisjonsratio 3:1

$\eta = 98 \%$; Girets virkningsgrad

$J = J_g = 0,01214 \text{ kgm}^2$; Treghetsmomentet til giret

Lineær omformer CKK 25-200 parametere:

$L = 500 \text{ mm}$; Lengde på enhet
Carriage Option no# : 11	; Konfigurering av Compact Module
No# of carriages : 2	; Antall slider
$P = 10 \text{ mm}$; Ball screw lead
$d_0 = \text{diameter motor shaft} = 32 \text{ mm}$; Ball screw diameter
$M_{RS} = M_R = 1 \text{ Nm}$; Friksjonskraft CKK(negativ)
$k_1 = 84,211$, $k_2 = 0,64$, $k_3 = 2,533$; Konstanter, side 20 i link
$M_{perm} = 18 \text{ Nm}$; Permissible drive torque tabel
$l_m = 175 \text{ mm}$; Avstand mellom slidene
Excess Travel $32 \times 10 = 10 \times 2 = 10 \text{ mm}$; Overflødig gang
$v_{perm} = 30 \text{ m/min}$; Permissible speed tabel
$J_k = 200 \times 10^{-6} = 0,2 \times 10^{-3} \text{ kgm}^2$; Koblings treghetsmoment

Beregninger av system:

$M_{tot} = (M_m * i * \eta) - MRS = (12*3*0,98) - 1 = \underline{34 \text{ Nm}}$; Nominelt moment i systemet.

$n_{tot} = n_m / i = 5000 / 3 = \underline{1666,67 \text{ min}^{-1}}$; Maksimalrotasjonshastighet til systemet

$m_{fr} = 0,0299 * L * 8,7 = 0,0299 * 500 * 8,7 = 23,65 \text{ kg}$; Vekt på ekstern last CKK.

$J_s = (k_1 + (k_2 * L) + (k_3 * m_{fr})) * 10^{-6} =$

$(84,211 + (0,64 * 500) + (2,533 * 23,65)) * 10^{-6} =$

$\underline{J_s = 464,11645 * 10^{-6} = 0,46411645 * 10^{-3} \text{ kgm}^2}$; Treghetsmoment til CKK.

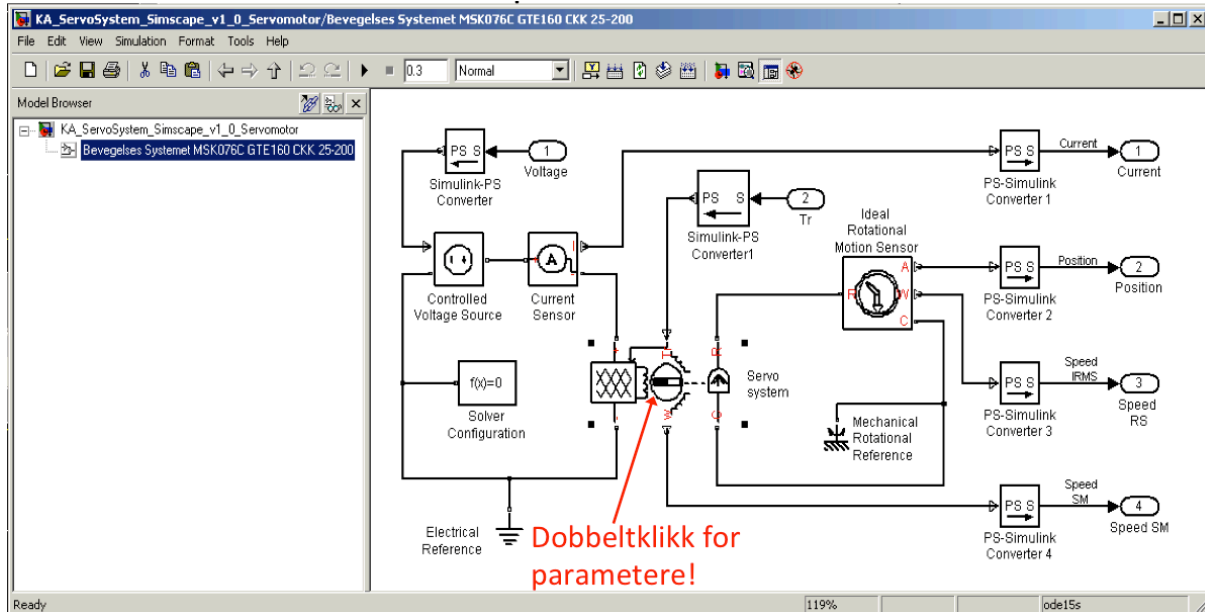
$J_{tot} = J_s + J_k + J_g + J_{Br} + J_m =$

$(0,46411645 + 0,36 + 12,14 + 0,2 + 4,3) * 10^{-3} =$

$\underline{J_{tot} = 17,46411645 * 10^{-3} \text{ kgm}^2}$; Treghetsmoment for system

$r_{bs} = d_0 / 2 = 32 / 2 = 16 \text{ mm} = 0,016 \text{ m}$; Radius til CKK «ball screw»

$F_{tot} = (M_{tot} / r_{bs}) = (34 / 0,016) = \underline{2125 \text{ N}}$; Nominell Kraft levert horisontalt av systemet



Figur 48: Simscape Sub-modell

Block Parameters: Servo system

Servomotor

This block represents a servomotor and driver operating in torque-control mode. Based on the assumption that torque is proportional to current, this is equivalent to the driver controlling current. The motor's permissible range of torques and speeds is defined by the manufacturer torque-speed envelope, and the output torque is assumed to track the torque reference demand T_r with time constant T_c . If the motor manufacturer does not define an envelope, then set the vector of maximum torque values to $[T_{max} \ T_{max} \ 0 \ 0]$ and the vector of rotational speed values to $[0 \ (1-eps)*w_{max} \ (1+eps)*w_{max} \ 2*w_{max}]$ where T_{max} is the rated maximum continuous torque, w_{max} is the maximum speed and eps is a small positive number e.g. 0.01.

The controlled servomotor should be connected to a DC supply. Electrical losses are assumed to be the sum of a torque-independent term plus a term proportional to the square of the torque. In addition, a resistor in series with the supply can be included to model transmission losses between power supply and servomotor driver.

The block produces a positive torque acting from the mechanical C to R ports.

Parameters

Electrical Torque	Mechanical
Vector of rotational speeds:	$666.67*(1-0.01) \ 1667.67*(1+0.01) \ 2*1666.67$ rpm
Vector of maximum torque values:	$[\ 34 \ 34 \ 0 \ 0]$ N*m
Torque control time constant, T_c :	0.02 s
Motor and driver overall efficiency (percent):	100
Speed at which efficiency is measured:	1667.67 rpm
Torque at which efficiency is measured:	34 N*m
Torque-independent electrical losses:	0 W
Supply series resistance:	0 Ohm

OK Cancel Help Apply

Block Parameters: Servo system

Servomotor

This block represents a servomotor and driver operating in torque-control mode. Based on the assumption that torque is proportional to current, this is equivalent to the driver controlling current. The motor's permissible range of torques and speeds is defined by the manufacturer torque-speed envelope, and the output torque is assumed to track the torque reference demand T_r with time constant T_c . If the motor manufacturer does not define an envelope, then set the vector of maximum torque values to $[T_{max} \ T_{max} \ 0 \ 0]$ and the vector of rotational speed values to $[0 \ (1-eps)*w_{max} \ (1+eps)*w_{max} \ 2*w_{max}]$ where T_{max} is the rated maximum continuous torque, w_{max} is the maximum speed and eps is a small positive number e.g. 0.01.

The controlled servomotor should be connected to a DC supply. Electrical losses are assumed to be the sum of a torque-independent term plus a term proportional to the square of the torque. In addition, a resistor in series with the supply can be included to model transmission losses between power supply and servomotor driver.

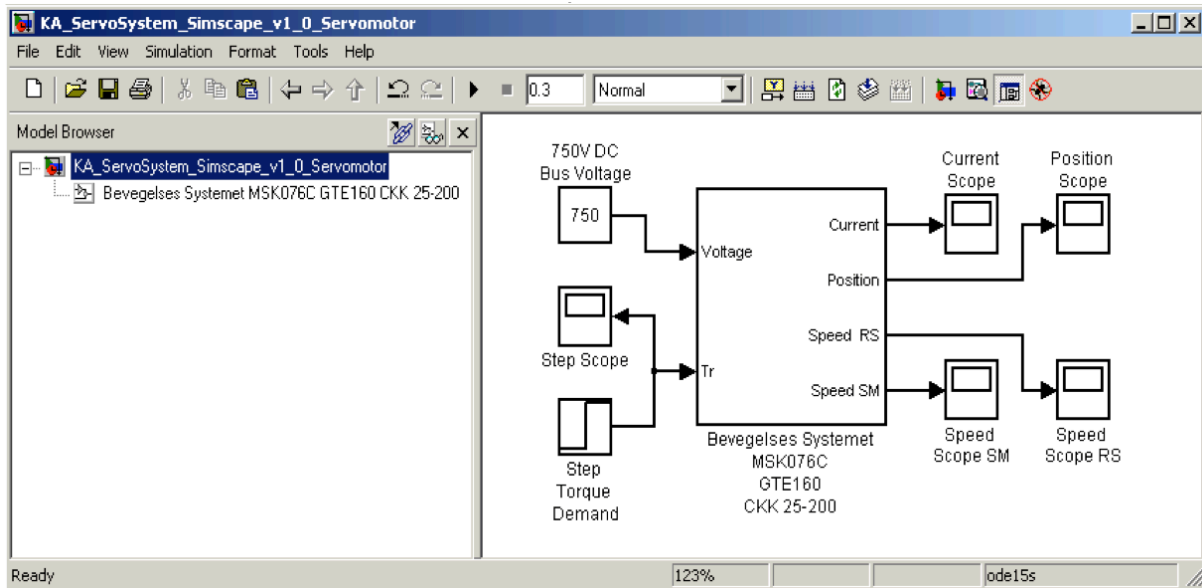
The block produces a positive torque acting from the mechanical C to R ports.

Parameters

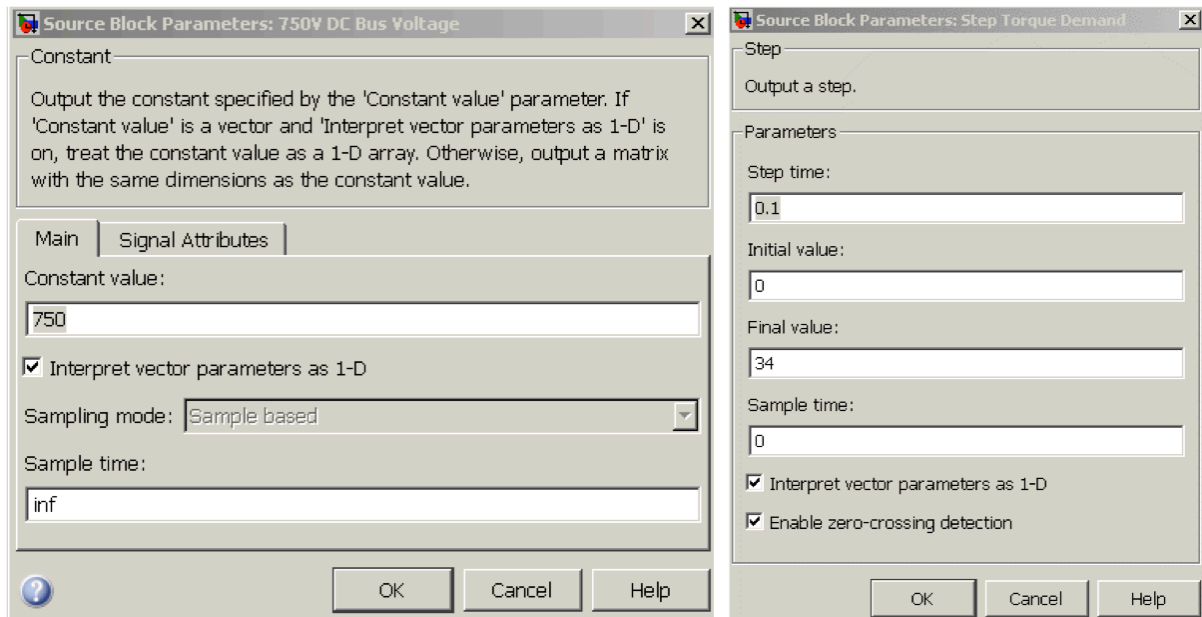
Electrical Torque	Mechanical
Rotor inertia:	$17.46411645e-3$ kg*m ²
Rotor damping:	$1e-5$ N*m/(rad/s)
Initial rotor speed:	0 rpm

OK Cancel Help Apply

Figur 49: Servo-system parametere



Figur 50: Simscape modell



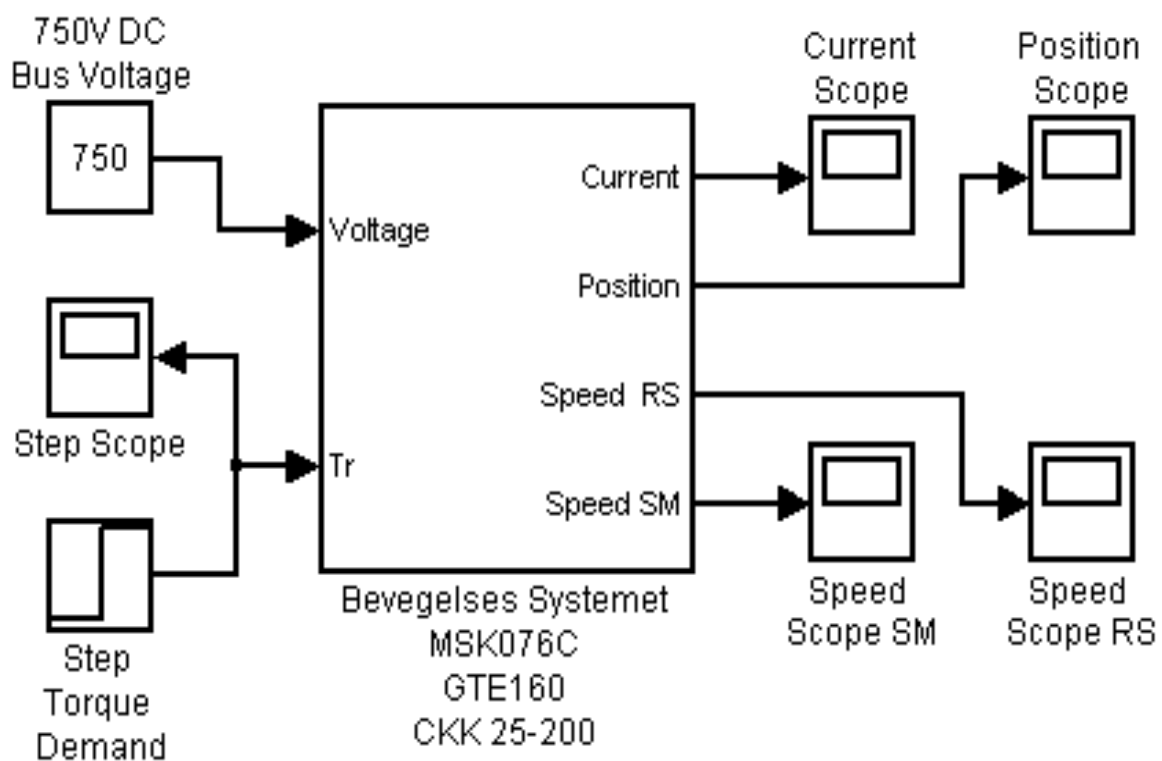
Figur 51: « Source Block» Parametere

Plotting av grafer

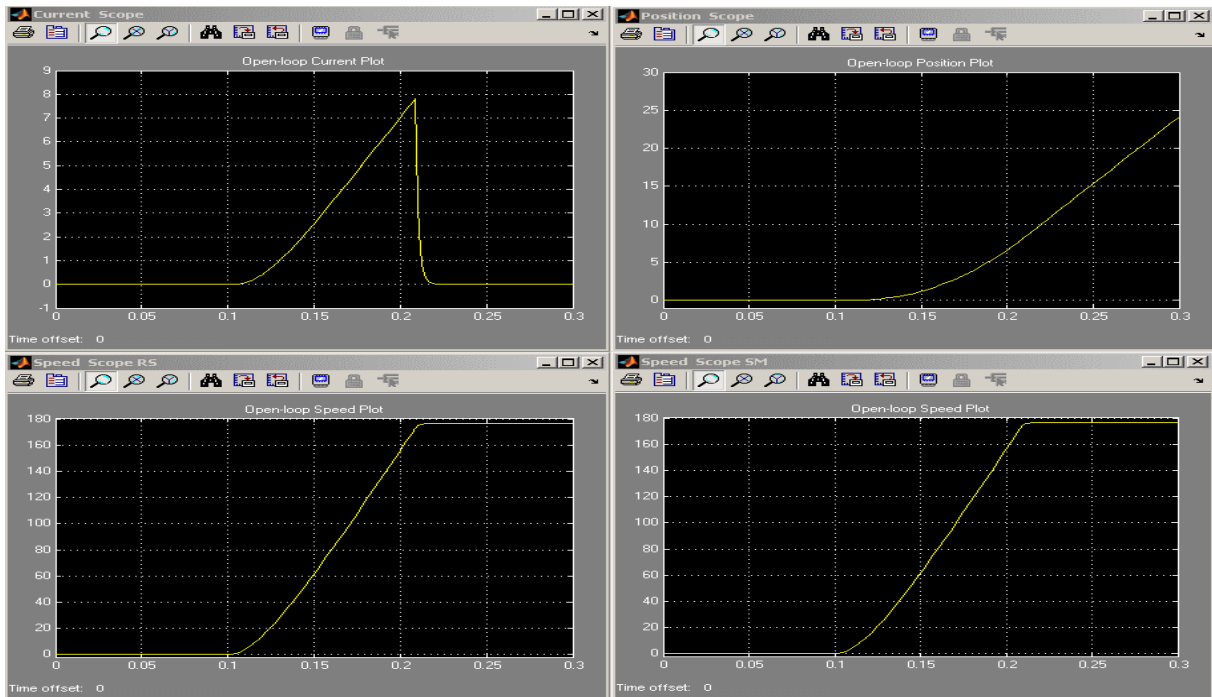
For å kunne Simulere hele modellen med overvåkning, gjør vi om DC-motor modellen til en Sub-modell i en ny modell. Modellen har inngang for å kunne påføre motor spenningsprang, med oscilloskop for å kunne vise plott av spranget.

Oscilloskop er også tilkoblet utgangene for å plotte respons kurver ved sprang på inngang:

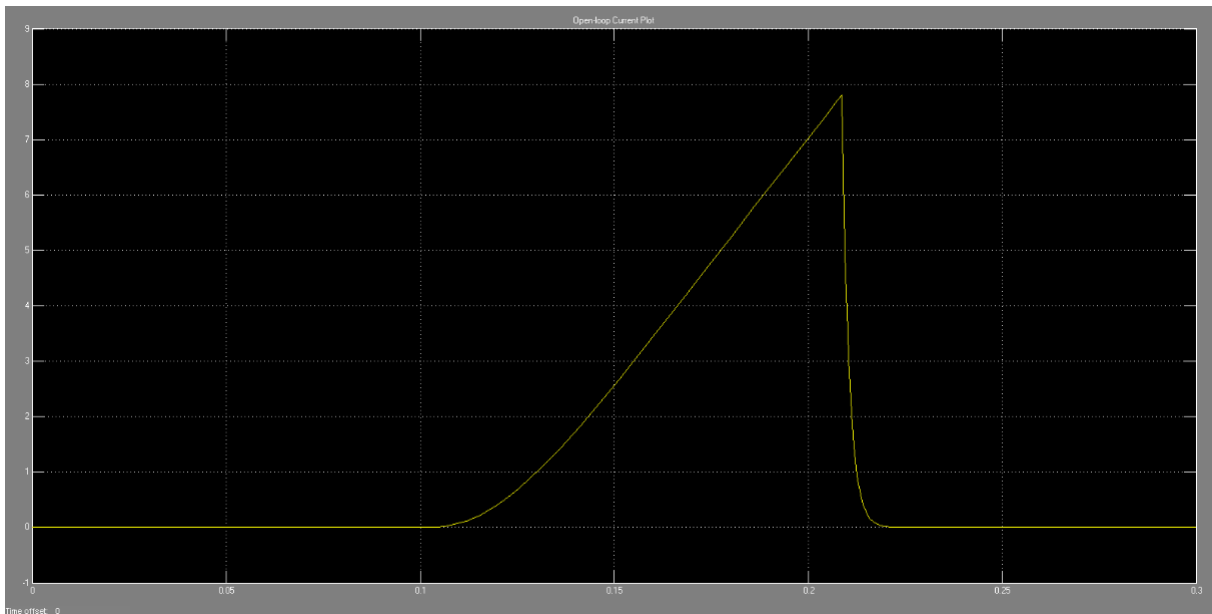
- Current Scope gir «open-loop» graf på strøm(A), som er proporsjonalt med kraft(N) avgitt fra motor. Vi beregner regulator med Current/Torque plott.
- Position Scope gir «open-loop» graf på posisjonen til motor, hvor mange rotasjoner den gjør.
- Speed Scope SM gir «open-loop» graf på hastighet direkte fra motor.
- Speed Scope RS gir «open-loop» graf på hastighet målt av rotasjons sensor.



Figur 52: Sub-modell av DC-motor



Figur 53: Servo-systemets Open-Loop Plots



Figur 54: Current «open-loop» plot

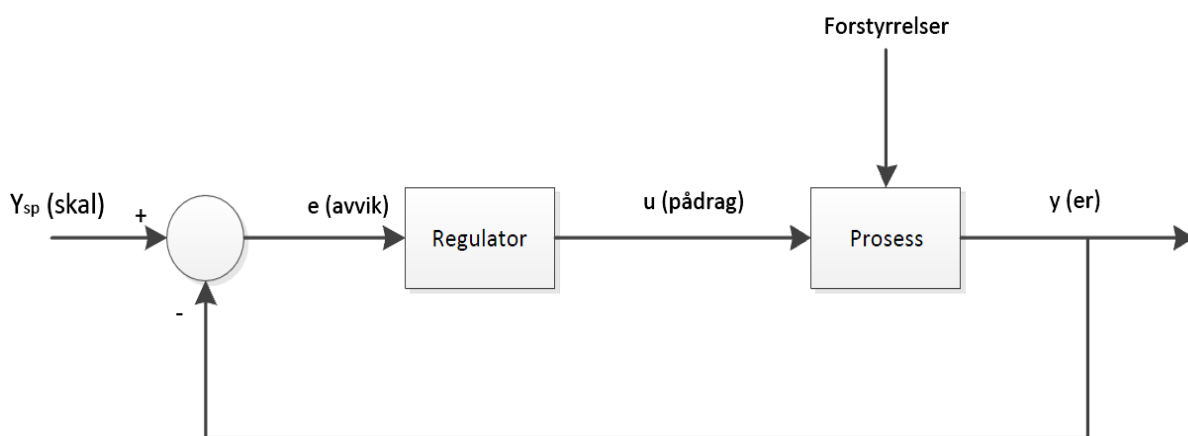
2.7.4 – Regulerings-Teknisk beregning av system

Teori PID-regulatoren

Virkemåte

PID-regulatorens funksjon i et system er å holde prosessen stabil selv ved forstyrrelser utenifra. Dette gjøres med en tilbakekobling som vist i *figur*. Verdien av utsignalet til prosessen, trekkes fra som er settpunktet. Avviket som defineres som, blir så lest inn i regulatoren. Regulatoren foretar nødvendige justeringer på pådraget.

Signalet sendes til prosessen og skal motvirke forstyrrelser. Regulatoren har flere parametere som kan justeres. Ved feil innstilling av disse verdiene kan regulatoren gjøre systemet ustabil, og det er derfor viktig med en grundig analyse av prosessen før verdiene settes.



Figur 55: PID-regulator

Standard matematisk fremstilling av en PID-kontrollerer er vist i *formel*:

$$u(t) = K_p \left(e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt} \right)$$

Figur 56: Formel for standard matematisk fremstilling av PID-kontroller

I *formel* inngår regulering parameterne K_p , T_i og T_d , hvor K_p er proporsjonalt pådrag. Det proporsjonale leddet fører til hurtigere regulering ved $K_p > 1$.

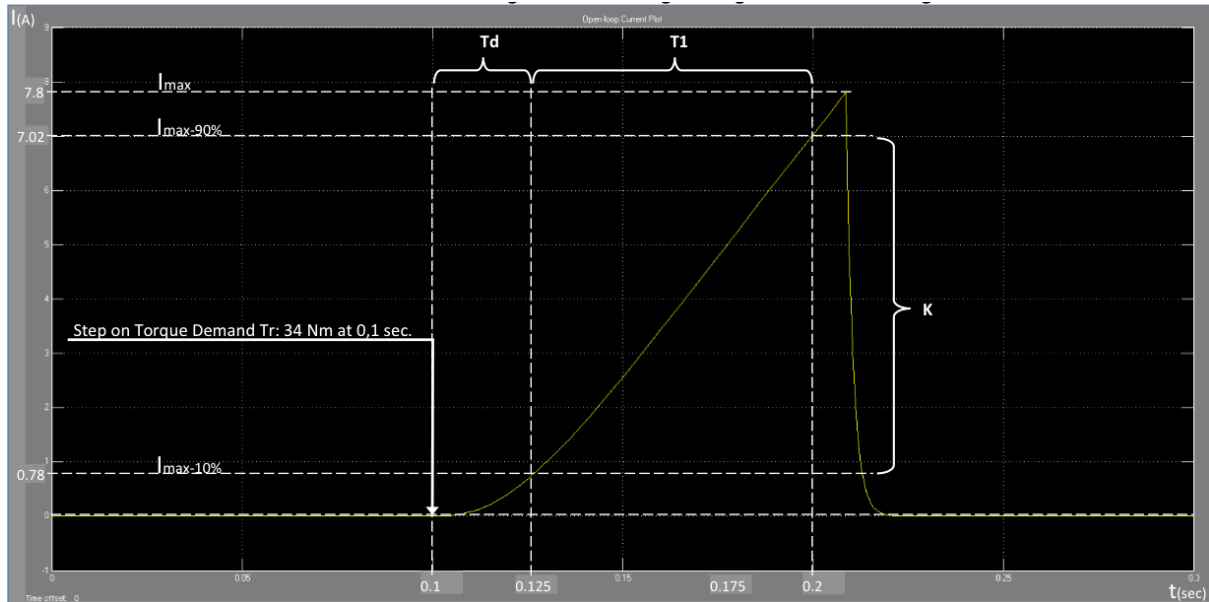
T_i er integraltiden som blir justert med hensyn på tiden det tar å regulere prosessen til settpunktet. Ved stort sprik mellom gjeldende verdi og settpunktet vil I-leddet gi et stort pådrag. Når gjeldene y -verdi nærmer seg settpunktet vil pådraget fra I-leddet minske. Dette har sammenheng med at I-leddet er proporsjonalt med arealet under avviket .

T_d er derivatdelen av kontrolleren, den forholder seg til avviket . Derivatoren er godt egnet for å takle raske endringer. Et problem med derivatleddet er at høyfrekvente forstyrrelser på målesløyfen kan føre til unødvendig regulering og slitasje på utstyret. Ved systemer som responderer tregt løses dette ved å sette $T_d = 0$ og bruke regulatoren kun som en PI-regulator.

Ziegler-Nichols

Current «open-loop» plot:

Her merket med interessante Ziegler-Nichols regulerings-tekniske tider og størrelser!



Figur 57: Current «open-loop» plot med Ziegler-Nichols

Avleste og beregnede regulator variabler:

Verdier avlest fra plott:

- $I_{max} = 7,8$ A
- $I_{min} = 0$ A
- $T_{step} = T_0 = 0,1$ s
- $T_{10\%} = 0.125$ s
- $T_{90\%} = 0,2$ s

Plott beregninger:

$$I_{\max 10\%} = 7,8 * 0,1 = \underline{0,78 \text{ A}}$$

$$I_{\max 90\%} = 7,8 * 0,9 = \underline{7,02 \text{ A}}$$

$$K = I_{\max 90\%} - I_{\max 10\%} = 7,02 - 0,78 = \underline{6,24}$$

$$T_d = T_{10\%} - T_{\text{step}} = 0,125 - 0,1 = \underline{0,025 \text{ s}} = \underline{25 \text{ ms}}$$

$$T_1 = T_{90\%} - T_{10\%} = 0,2 - 0,125 = \underline{0,075 \text{ s}} = \underline{75 \text{ ms}}$$

Ziegler Nichols - «Open-loop» metode:***P:***

- $K_p = T_1 / (K * T_d)$

PI:

- $K_p = (0,9 * T_1) / (K * T_d) = (0,9 * 75) / (6,24 * 25) = \underline{0,325}$

- $T_i = 3,3 * T_d = 3,3 * 25 = \underline{0,0726 \text{ s}} = \underline{72,6 \text{ ms}}$

PID:

- $K_p = (1,2 * T_1) / (K * T_d) = (1,2 * 75) / (6,24 * 25) = \underline{0,577}$

- $T_i = 2 * T_d = 2 * 25 = \underline{50 \text{ ms}}$

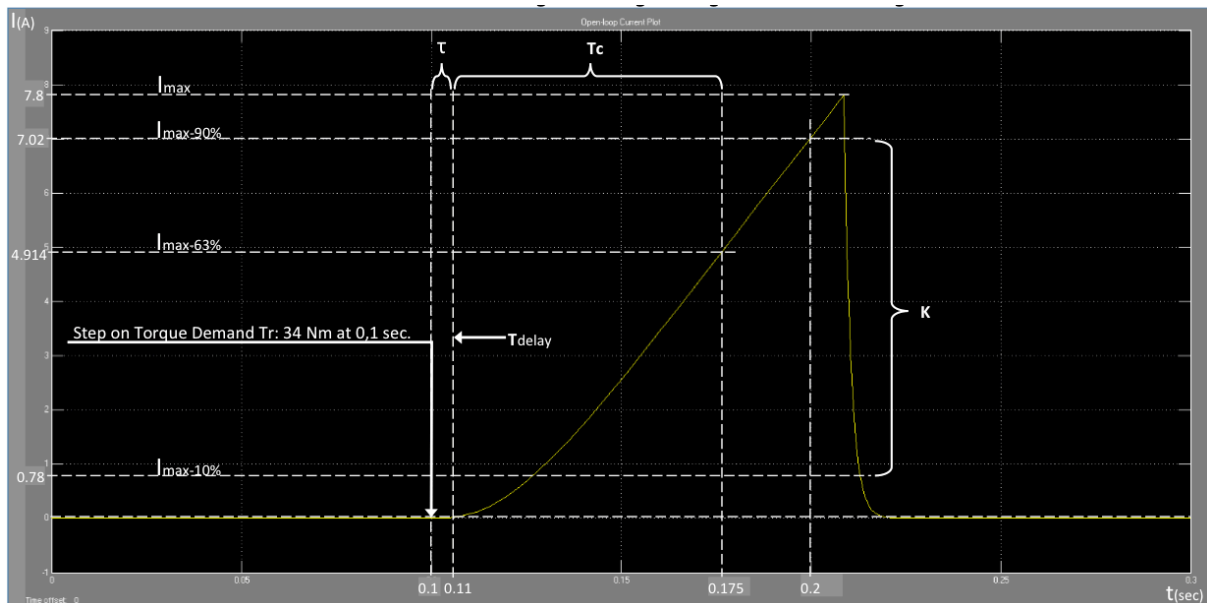
- $T_d = 0,5 * T_d = T_i / 4 = \underline{12,5 \text{ ms}}$

T_d er avlest tid, og T_d er Ziegler Nichols beregnet derivat tid.

Skogestad

Current / Torque «open-loop» plot:

Her merket med interessante Skogestad reguleringstekniske tider og størrelser!



Figur 58: Current «open-loop» plot med Skogestad

Avleste og beregnede regulator variabler:

Verdier avlest fra plott:

- $I_{max} = 7,8$ A
- $I_{min} = 0$ A
- $T_{step} = T_0 = 0,1$ s
- $T_{10\%} = 0.125$ s
- $T_{90\%} = 0,2$ s

Plott beregninger:

$$I_{\max 10\%} = 7,8 * 0,1 = \underline{0,78 \text{ A}}$$

$$I_{\max 90\%} = 7,8 * 0,9 = \underline{7,02 \text{ A}}$$

$$I_{\max 63\%} = 7,8 * 0,63 = \underline{4,914 \text{ A}}$$

$$K = I_{\max 90\%} - I_{\max 10\%} = 7,02 - 0,78 = \underline{6,24}$$

$$\tau = T_1 - T_{\text{step}} = 0,11 - 0,1 = \underline{0,01 \text{ s}} = \underline{10 \text{ ms}}$$

$$T_c = T_{63\%} - T_1 = 0,175 - 0,11 = \underline{0,065 \text{ s}} = \underline{65 \text{ ms}}$$

$$T = T_c + \tau = 65 + 10 = \underline{75 \text{ ms}}$$

Skogstad - «Open-loop» metode:**Forklaring av formler:**

$k = 1,5$, for hurtig kompensering for forstyrrelse i prosessen.

T_1 er den største , og T_2 er den minste tidskonstanten i prosessen. $T_1 = T_c$, $T_2 = \tau$.

T = Total tid, $T_1 + T_2$.

$T_c = \tau$, så lenge det ikke er spesielle grunner til å beholde verdi på T_c .

***PI* (integrator):**

- $K_p = 1 / (K * (T_c + \tau)) = 1 / (6,24 * (0,01 + 0,01)) = \underline{8,013}$

- $T_i = k * (T_c + \tau) = 1,5 * (0,01 + 0,01) = \underline{0,03}$

***PI* (tidskonstant):**

- $K_p = T / (K * (T_c + \tau)) = 0,075 / (6,24 * (0,01 + 0,01)) = \underline{0,601}$

- $T_i = \min[T, k(T_c + \tau)] = \min[0,075, ((1,5) * (0,01 + 0,01))] = \min[0,075, 0,03] = \underline{0,03}$

PI(D) (integrator+tidskonstant):

- $K_p = 1 / (K * (T_c + \tau)) = 1 / (6,24 * (0,01 + 0,01)) = \underline{8,013}$
- $T_i = k * (T_c + \tau) = 1,5 * (0,01 + 0,01) = \underline{0,03}$
- $T_d = T = \underline{0,075}$

PI(D) (To tidskonstanter):

- $K_p = T / (K * (T_c + \tau)) = 0,075 / (6,24 * (0,01 + 0,01)) = \underline{0,601}$
- $T_i = \min[T_1, k * (T_c + \tau)] = \min[0,065, ((1,5) * (0,01 + 0,01))] = \min[0,065, 0,03] = \underline{0,03}$
- $T_d = T_2 = \tau = \underline{10 \text{ ms}}$

PI(D) (Dobbel integrator):

- $K_p = 1 / (4 * K * ((T_c + \tau)^2)) = 0,075 / (6,24 * (0,01 + 0,01)) = \underline{0,601}$
- $T_i = 4 * (T_c + \tau) = 4 * (0,01 + 0,01) = \underline{0,08}$
- $T_d = 4 * (T_c + \tau) = 4 * (0,01 + 0,01) = \underline{0,08}$

PID Algoritme:

$T = 1$; Tidskonstant

K_p = Proporsjonal-ledd , K_i = Integral-ledd , K_d = Derivat- ledd

- $U(k) = U(k-1) + (K_p * T / T_i) * e(k) + K_p * e(k-1) + K_p * T_d * 2e(k-1)$
- $U(k) = U(k-1) + K_i * e(k) + K_p * e(k-1) + K_d * (e(k) - e(k-1))$
- $U(k) = U(k-1) + A * e(k) + B * e(k-1) + C * e(k-2)$
- $\Delta u = A * e_p + B * e_i + C * e_d$

3 – Sluttprodukt

I dette kapitlet vil vi vise det endelige produktet, og hva vi har fått til, og hvordan produktet fungerer. I og med at vi ikke fikk ferdigstilt reguleringen, har vi beskrevet en del om hvordan systemet skal fungere når det er ferdigstilt. Vi kommer til å jobbe videre frem til siste presentasjon, og forhåpentligvis få et fullstendig produkt da.

3.1 – Hvordan fungerer simuleringen?

Simuleringen er bygd opp av hardware, software, mekaniske og elektriske komponenter. Når det gjelder software er det to programmer som brukes, LabVIEW og IndraWorks. Disse har et "slave-master" forhold ved at LabVIEW sender et digitalt signal til MLD som starter hele prosessen. IndraWorks styrer servomotor og regulering, mens LabVIEW styrer logging, sensoravlesning, sensorkalibrering og beregning av referansekraft utifra aktuell posisjon. Alle sensorverdier går inn til LabVIEW via analog input modulen i cRIO. Simuleringen fungerer slik at man starter med å laste opp en girgraf i form av csv format til labview. Deretter begynner man å sende referansekraft fra denne filen til MLD via analog output modulen i cRIO. Der behandles signalet i form av at det må ganges med 100 fordi rekkevidden på analog output er +/- 10 V så signalet måtte skaleres ned før sending fra LabVIEW.

I IndraWorks er det underprogram som heter IndraLogic. IndraLogic er PLC programmering i form av ladder diagrammer eller structural text (det finnes også flere måter i programmere PLS i IndraLogic, men det er disse to som er relevant for oss). Referansesignalet blir mottatt og sammenlignet med aktuell kraft målt på lastcelle og blir deretter regulert via en PID regulator i PLS programmet. Deretter blir den nye kraften sendt til servomotor som bruker denne kraften som brems. PLS delen av dette fungerer ikke helt som vi vil i skrivende stund (se resultat i prosjektrapport).

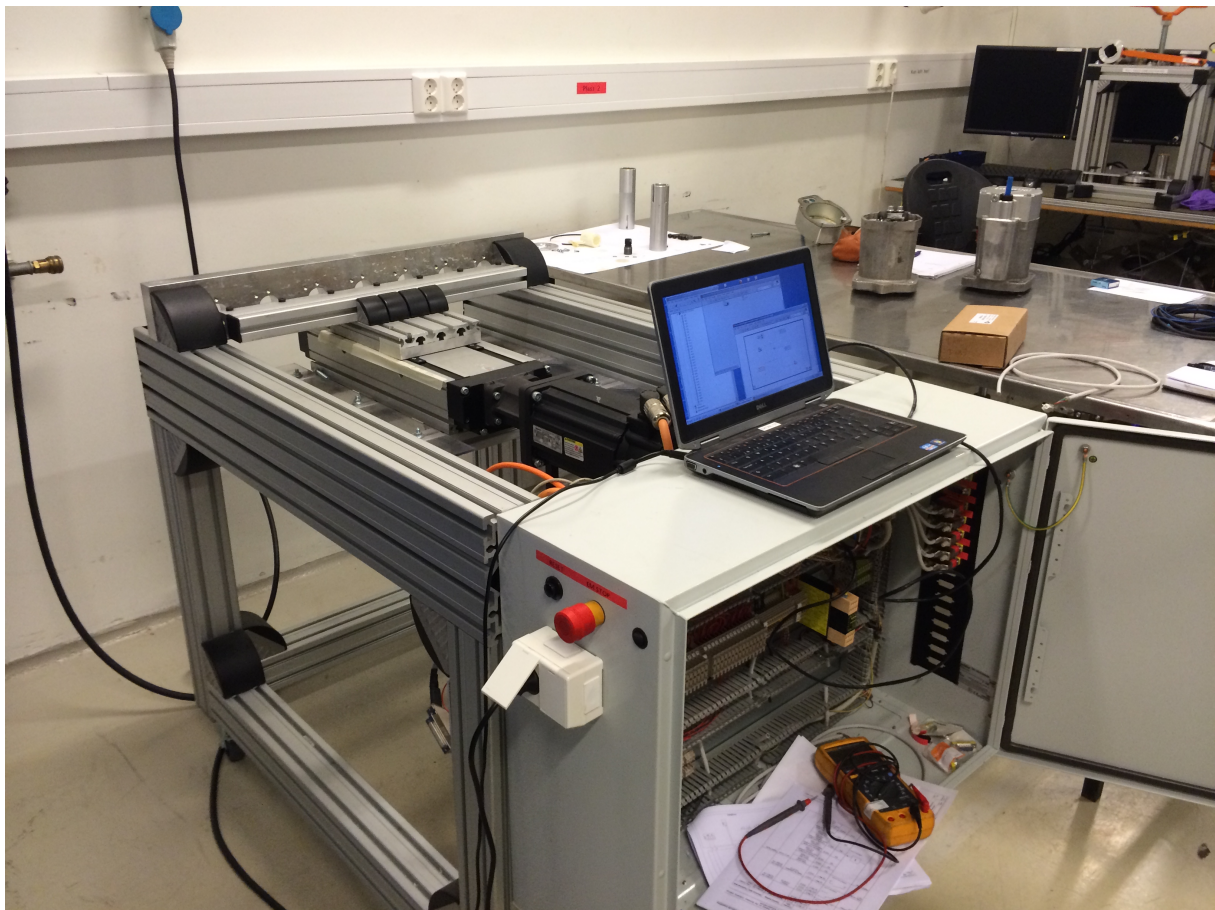
Optimalt ville den aktuelle motkraften og aktuelle posisjonen også blitt vist i en XY graf i LabVIEW, der vi kunne sammenlignet grafen vi skulle følge og den grafen vi faktisk kjører. Grunnet at vi mangler PLS programmet så får vi ikke den aktuelle grafen slik at vi får sammenlignet.

3.2 – Rigg

I dette avsnittet skal vi fortelle litt om oppsettet på rigg R-144, hvilke komponenter den inneholder, hva slags utstyr som sto der fra før og hva vi har tilført.

Riggen i seg selv er utstyrt med profiler for feste av sensorer, girspaker og lignede så lenge det festes med skruer. Den er ca. 105cm høy, 83cm bred og 140cm lang.

På baksiden av riggen sitter det en kontroll enhet eller MLD (CSH01) og en power section HSC02. MLDen har vi kjøpt inn fra Borsch Rexroth, mens HSC02 hadde KA fra før av.

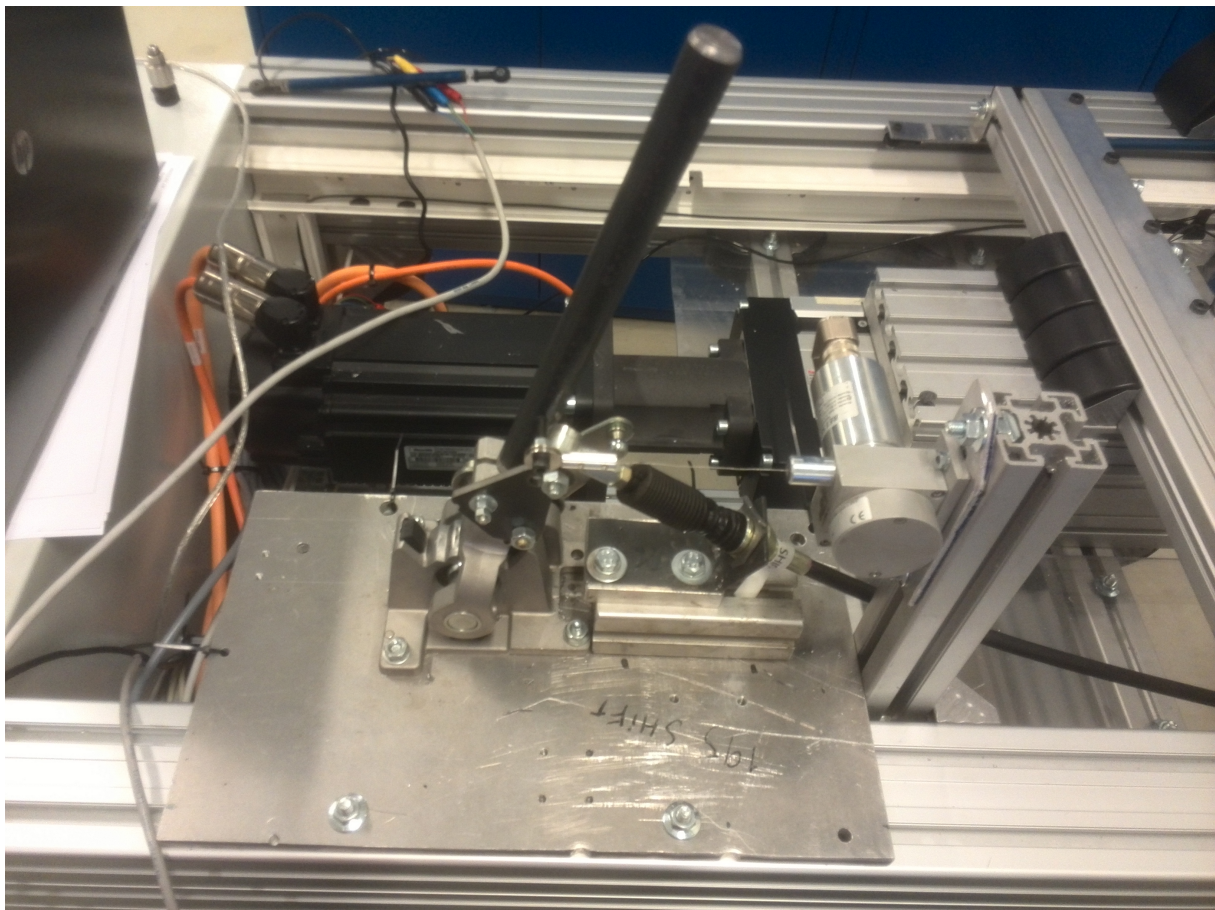


Figur 59 Rigg R-144

3.2.1 - Girspak

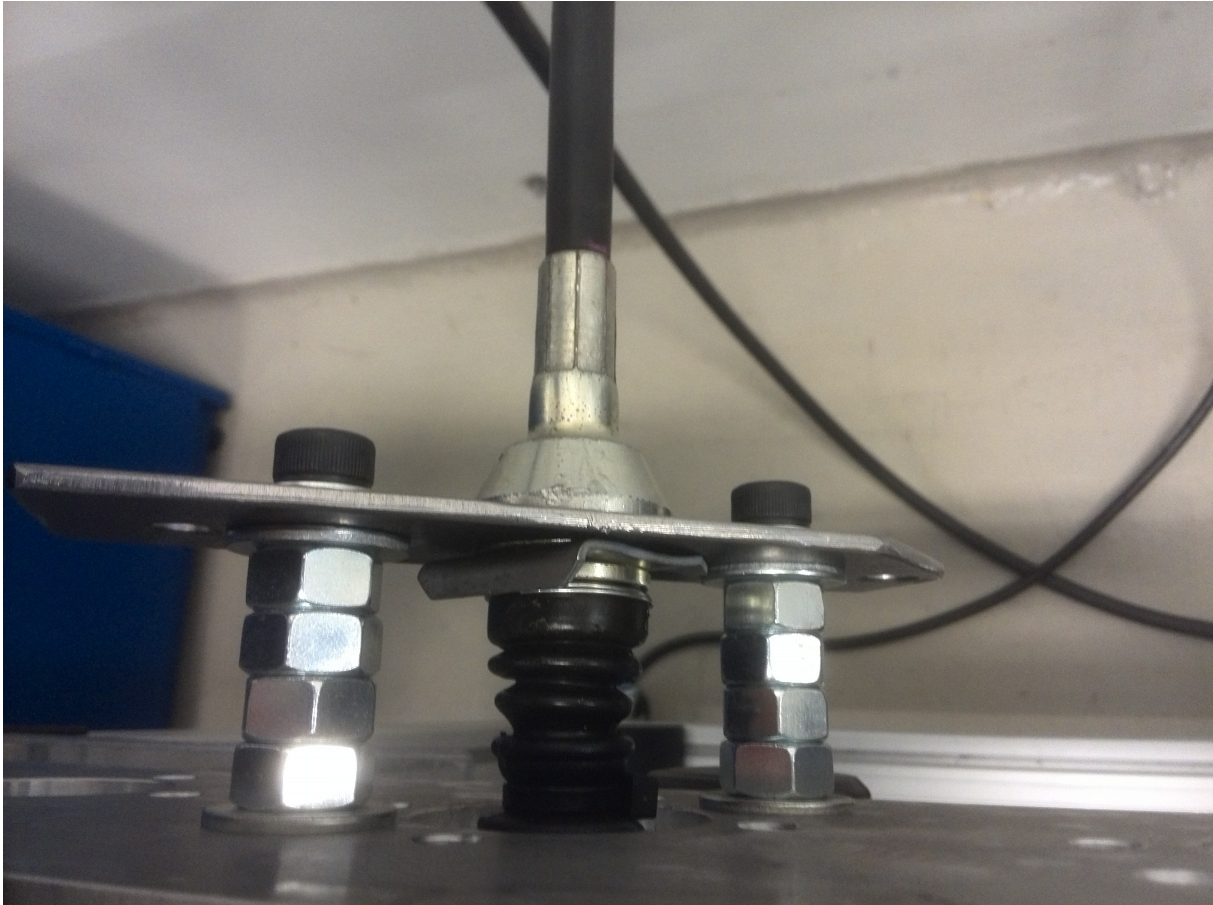
Vi tenkte først at det ville bli enklere å kjøre systemet opp mot luft-sylinderen vi koblet opp, men vi byttet etter hvert til en girspak som var festet til en metallplate og en kabel og vi kunne koble til den ene siden av lastcellen.

I og med at metallplata girspaken var festet i allerede hadde hull til skruer der vi ønsket å feste den, slapp vi borre nye. Den ble festet på siden av riggen som vist på figur 1. Trådpotmeteret ble skrudd fast i et metallstykke med profiler for feste av skruer. Enden av tråden var originalt festet i en skrue lenger ned på girspaken, men vi fant ut at vi ville feste den på selve girstanga isteden for å få mer nøyaktige målinger og fordi vi kunne benytte mer av den 500mm lange tråden på potmeteret. Dette betyr jo da at det er større forskjell på hver individuelle posisjon i forhold til volt.



Figur 60: Girspak ferdig skrudd fast til riggen

Kabelen ble festet til lastcellen som vist på figur 2. Det var mulig å bevege oss fra minste posisjon til største når vi dro i girspaken, men ikke når vi dyttet. Vi var i alle fall ikke villige til å dytte med så store krefter og risikere å ødelegge kabelen i forsøket. Vandringen på girspaken ble målt via med en annen posisjongiver til ca. 72,2mm.



Figur 61: Feste til kabelen

3.2.2 - Skap

Skapet på riggen vår inneholder hovedsakelig reléer, strømforsyninger, rekkeklemmer, forsterker til lastcelle og National Instruments compactRIO samt alle de frem modulene som hører til de digitale og analoge inputene. På den siden av skapet sitter det en nødstoppbryter i tilfelle krise, en reset-bryter og en grønn lampe slik at vi lett ser om reset-bryteren er trykket inn eller ikke.

På den andre siden sitter det åtte såkalte «15-pin D-Sub connectors», hvor hver av portene enten går til 5 V eller 24 V på modul 4 til cRIO. Alle portene har følgende funksjon;

AI 11 går til posisjonsgiveren som forteller oss om hvilken posisjon CKK er i.

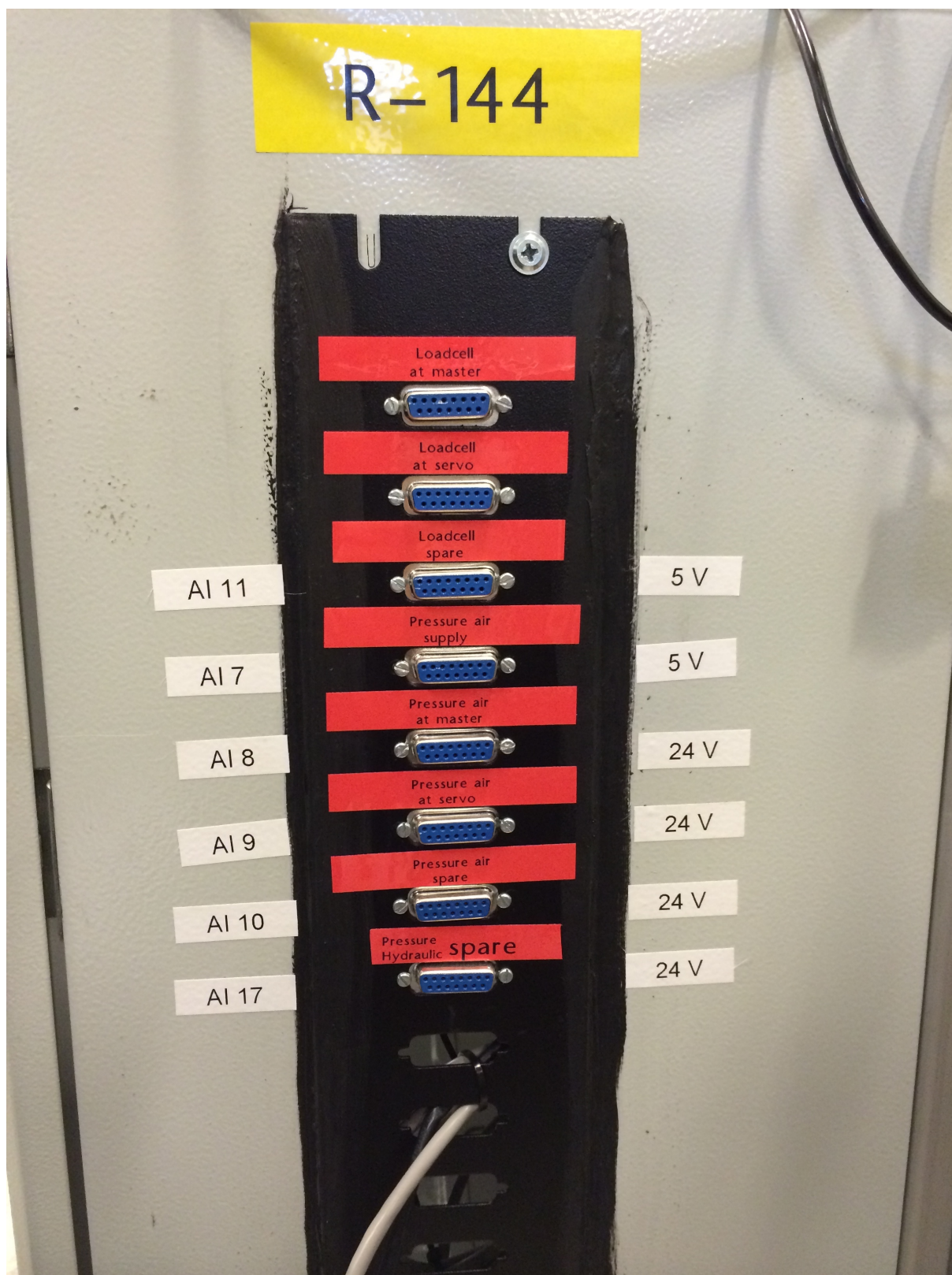
AI 7 går til den andre identiske posisjonsgiveren som skal sitte før aktuatoren.

AI 8 representerer trådpotmeteret (BEI SENSORS) som gir oss posisjonen til girstanga.

AI 9 skal gå til gearnob-lastcellen som festes på toppen av girstanga.

AI 10 gir oss informasjon fra lastcellen som ikke behøver forsterker, og som sitter rett etter girspaken, men før aktuatoren.

AI 17 er da den lastcellen som sitter mellom KA aktuatoren og CKK.



Figur 62: Tilkobling sensorer

3.3 – Videreutvikling

Simulatoren er ikke ferdig (se resultat i prosjektrapport).

Om vi hadde rukket å ferdigstille alle delmål og ønskelige krav, skulle det vært mulig å enkelt koble til systemet på hvilken som helst av oppdragsgivers gir og clutch-aktuatorer.

Uten å gjøre store mekaniske endringer på riggen skal man kunne endre hele simulatoren fra å være en girsimulator, til å bli en clutchsimulator. Dette skal kunne skje bare ved å bytte aktuator og bytte fra girspakoppsett til clutchoppsett. I LabVIEW må man bytte graf man leser fra for å få riktige posisjon/kraft verdier.

Per dags dato beregnes referansekraften som reguleres i PID regulatoren i LabVIEW. Dette går for tregt, så ved en videreutvikling ville kraften blitt beregnet i IndraWorks.

3.4 – Vedlegg

Alle ekstra informasjon som er beskrevet i teksten og all dokumentasjon til prosjektet, ligger vedlagt på CD. Dette gjelder blant annet brukermanualer, datablad og mer utdypende informasjon på enkelte områder.

De viktigste mappene er:

07 Manualer & Dokumentasjon på Utstyr

Her ligger dokumentasjon, datablader og annen info om de forskjellige komponentene i systemet.

08 Utstyr, Alternativer & Prinsippkisser

Grundig utdypning for valg av servodriver

09 IndraWorks SW test av Servomotor hos KA

Bilder og screenshots av første servomotortest med IndraWorks

17 Bilder, Figurer & Tegninger

Her ligger alt av bilder fra de forskjellige delen prosjektet

20 Regulering av Løsning

Utdypende MathLab-utregninger, simulering av regulering og støttedokumenter for reguleringen.

24 LabVIEW test

Koden til LabVIEW og screenshots (PS: koden kan ikke åpnes uten LabVIEW).

Brukermanual

GCS – Gir &

Clutchsimulator

Forfatter: Oskar



Innledning

I dette dokumentet beskrives akkurat hvordan systemet skal kunne styres av brukeren, feilkoder og feilmeldinger, avlesning av resultater og eventuelle endringer av utstyr.

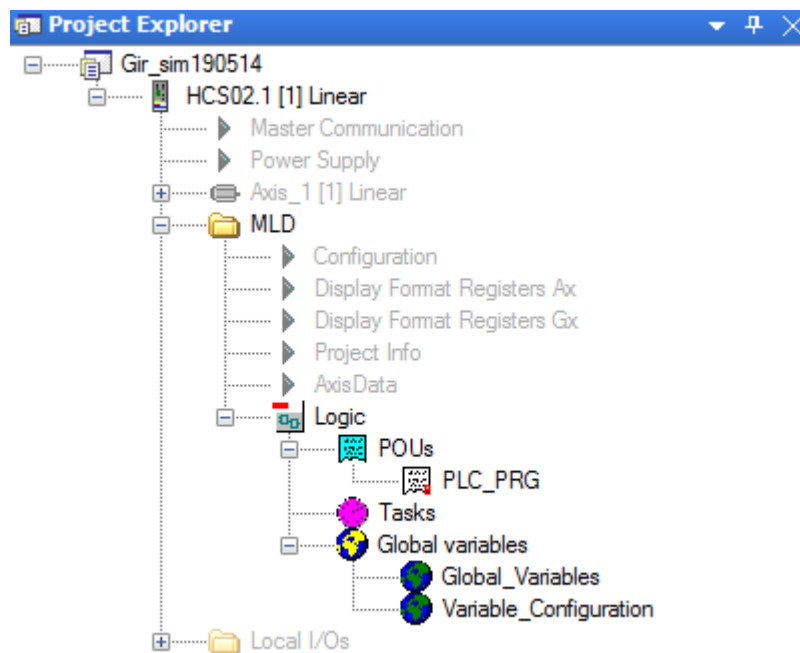
Alt av beregninger, prinsippsskisser, manualer og bilder ligger vedlagt.

Innholdsfortegnelse

1 – Hvordan styrer man systemet?	4
2 – Feilkoder	5
3 – Avlesning av resultater.....	6
4 – Endre utstyr	8

1 – Hvordan styrer man systemet?

Når man har gått inn i IndraWorks og gjort servomotoren klar for oppstart, da via «easystartup mode» og deretter trykt «enable» under «Commishoning» og ok på varselet om klemfare, kan man først starte med simuleringen. I dette treet kan man forøvrig også sette limit parametere som forteller hvor langt CKK kan maksimalt og minimalt gå i hver retning, og da kjøre mellom disse verdiene.



Figur 1: IndraWorks project tree

Under easy startup mode er det mulighet til å ta små eller store forflytninger manuelt. Dette gjøres ganske enkelt via «Jogging», der man stiller inn hvor mange rotasjoner per minutt servomotoren skal ha og deretter velger enten «Jog+» eller «Jog-», som man da holder inne.

I LabVIEW må brukeren trykke på «run MLD» som da skal sende et digitalt signal til IndraLogic som igjen starter PLS-sekvensen. Man må også trykke «Login» i IndraLogic for at for at prosjektet skal lastes inn, om man har laget det i ST, LD eller FBD. Denne funksjonen er ikke helt på plass enda, men vi regner med å kunne vise det frem på 3. presentasjon. Om det måtte være enten på film eller om det finnes 3-fase uttak til riggen så vi faktisk kunne bragt den med og vist frem.

2 – Feilkoder

I IndraWorks håndterer man feilmeldinger på den måten at man simpelthen trykker «clear error».

Feilmeldinger i IndraLogic ligner på dem man finner i et vanlig programmeringsspråk som for eksempel C++, der man får opp hvilken linje det gjelder, syntax error eller lignende. Det gjelder hovedsakelig i ST (StructuralText) vel og merke.

De aller fleste feilmeldingene vi traff på i LabVIEW handlet om manglende forbindelse mellom ledninger og informasjonsblokker. Dette førte jo da igjen til en feilmelding om at programmet kjører med en error.

3 – Avlesning av resultater

LabVIEW har vi grafer for posisjon og kraft som leses ut i fra et vindu slik som påfigur 1, her ser man visuelt hver og en av inputene som er koblet til National Instruments compactRIO. Signalene blir representert i volt og har en verdi mellom 0 og 5, hvor 0 da naturlig nok er minimumsposisjon og 5 er maksimumsposisjon. Det er også en knapp for start av logging og en grønn lampe som forteller at logging pågår.

AI 11 går til posisjonsgiveren som forteller oss om hvilken posisjon CKK (armen som beveger seg) er i.

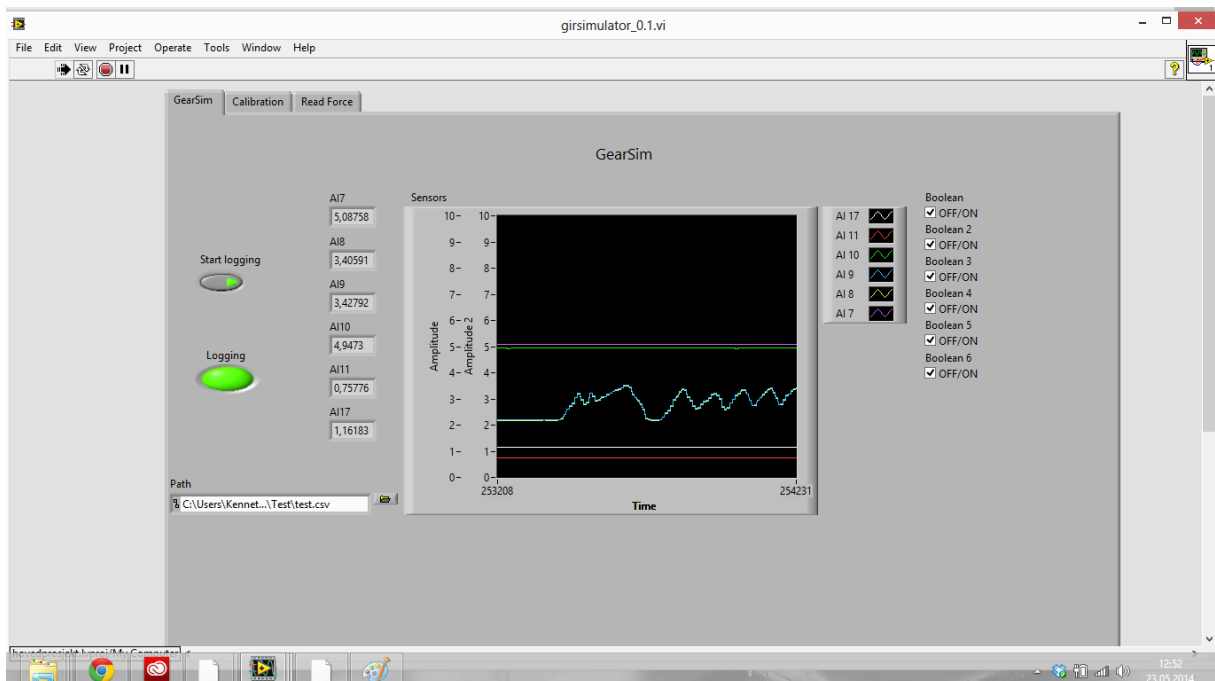
AI 7 går til den andre identiske posisjonsgiveren som skal sitte før aktuatoren.

AI 8 representerer trådpotmeteret (BEI SENSORS) som gir oss posisjonen til girstanga.

AI 9 skal gå til gearnob-lastcellen som festes på toppen av girstanga.

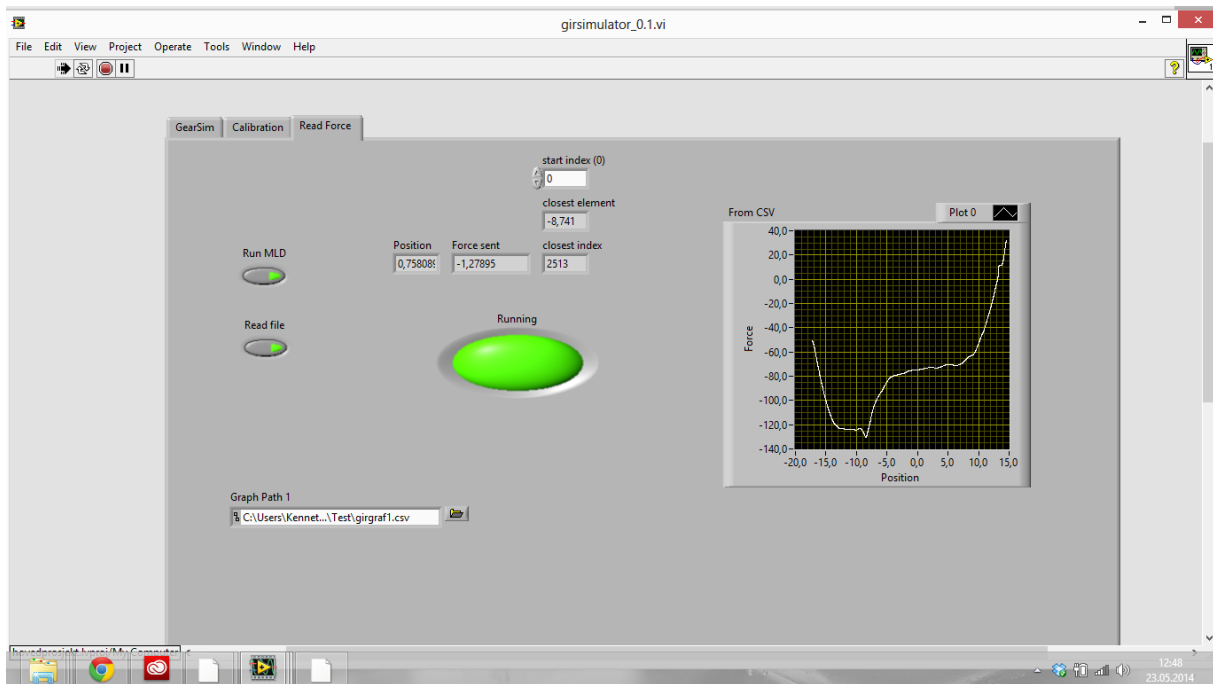
AI 10 gir oss informasjon fra lastcellen som ikke behøver forsterker, og som sitter rett etter girspaken, men før aktuatoren.

AI 17 er da den lastcellen som sitter mellom KA aktuatoren og CKK.



Figur 2: Graf fra LabVIEW over posisjon og kraft

På figur 3 ser vi verdier fra en CSV-fil (Excel) plottet på en graf med posisjon på x-aksen og kraft på y-aksen. Herifra kan man kjøre MLDen, lese filen og til alle tider enkelt se om programmet kjører med den grønne lampen. Hvis en av verdiene som blir målt er for eksempel i mellom to verdier i CSV-filen, vil programmet velge den verdien som ligger nærmest.



Figur 3: Kraft- og posisjons-graf

4 – Endre utstyr

Vi har ikke hatt nok tid til å få i land alle delmålene vi ville i løpet av prosjektet dessverre, et eksempel dette er clutchsimulatoren. Denne ville fungert nesten på akkurat samme måte, bortsett fra noen få programmeringstekniske og mekaniske endringer. På software siden var idéen at vi for det første trengte å bytte graf i LabVIEW fra Excel. Da hadde vi vært nødt til å måle nye verdier nesten på samme måte som når vi giret, bare med clutch i stedet. For det andre måtte vi endret små deler av programmet i LabVIEW og IndraWorks, som for eksempel endring av noen få rutiner eller variabler.

På den mekaniske siden ville vi vært nødt til å koble fra girspaken og tilhørende kabel som er koblet til lastcellen som sagt, deretter koble på clutch-pedal, kabel til lastcelle og tilhørende clutch-servo.

Prosjektrapport

GCS – Gir &

Clutchsimulator

Forfatter: Kenneth og Oskar



Innholdsfortegnelse

1 – Oppgaven	3
2 - Forventninger	4
3 – Utførelse av prosjektet	5
4 – Utfordringer	8
5 - Resultat.....	10
6 - Egenvurdering	11

1 – Oppgaven

Oppgaven er utlevert fra Kongsberg Automotive som Bacheloroppgave for en gruppe på HBV bestående av: Kenneth Bergerud, Krister Håkenrud, Martin Ingvaldsen og Oskar Gunbjørnsen.

Gruppas to hovedmål har først og fremst vært å finne ut om denne oppgaven faktisk lar seg løse med retningslinjene som ble gitt av oppdragsgiver. En annen gruppe har prøvd på samme oppgave tidligere, men var ikke i stand til å fullføre prosjektet på grunn av blant annet bruk av hydraulikk istedenfor en elektrisk servomotor og manglende kunnskaper innen reguleringsteknikk. Annet enn dette var det ingen som visste noe om oppgaven var løsbar og de har heller ikke hatt tid til å finne dette ut på egen hånd.

Det andre hovedmålet ble jo da å utarbeide en løsning som oppfylte krav om blant annet pris, responstid og kompatibilitet med LabVIEW som er et av programmene oppdragsgiver var godt kjent med. Som delmål har vi også fått i oppgave å lage en clutchsimulator som med enkle modifikasjoner på rigg R-144 og tabellene for kraftverdier skal kunne skiftes fra clutch til girskift og vice versa.

Primært ønsker KA seg disse simulatorene for å gjøre kundedemonstrasjoner og generelt fremvisninger av produktet enklere. Det er også et ønske om å sette simulatorene sammen med en lastebilsimulator slik at de får en mer realistisk følelse av Kongsberg Automotives produkter.

2 - Forventninger

Annet enn Kenneth som jobbet hos KA og var kjent med deres produkter, var det ingen andre som visste helt hvordan oppgaven kom til å bli. Martin og Oskar kom rett fra videregående skole med studieretning elektro før de startet på bachelor med samme studieretning, mens Krister jobbet med roboter i Norsk Titanium. Oppgaver som valg av servodriver og rett og slett hvordan man skulle benytte servomotoren til det vi ønsket, var helt ukjent.

Når vi opprinnelig fikk oppgaven lagt frem og forklart for oss, virket det litt overveldende med tanke på kompleksitet samt det tekniske språket som ble utvekslet mellom de som hadde kjennskap til det fra før. Allikevel var vi spente og hadde tro på at dette var noe vi kom til å klare.

Under en av de første forelesningene med Olaf Hallan Graven ble vi fortalt at vi kunne forvente å tenke på en helt annen måte ved prosjektets slutt sammenliknet med da vi først startet. Dette skulle vise seg å være helt sant. En av årsakene til dette er antageligvis hele prosessen med for eksempel timelister, veiledermøter og oppfølgingsdokumenter. Slike ting tok vi for gitt at kom til å bli trivielle oppgaver, men dette viste seg å være en stor utfordring ikke minst i begynnelsen når vi ikke hadde noen form for struktur eller maler slik mange andre grupper har hatt.

3 – Utførelse av prosjektet

Tatt i betraktning at vi har tre studenter fra elektro og én fra data, hvor Krister har en 60% stilling og Kenneth en helt annen ukeplan enn de andre på gruppa, så har prosjektarbeidet fungert ganske bra.

Den ordinære ukeplanen har de fleste ukene sett ut som følger;

Mandag starter vi dagen med å skrive et oppfølgingsdokument som skal reflektere hvor mye tid vi tror vi kommer til å bruke den uka, samt sammenlikne forrige ukes timeantall med den tiden vi trodde det ville ta. Deretter har vi et gruppemøte så fort Krister har ankommet (i og med at han pendler fra Hønefoss tar det naturligvis litt lenger tid) om hva hvilke person skal jobbe med og hva som er målet for den uka.

Tirsdag har elektro studentene hatt undervisning i multivariabel regulering fram til klokka to. Derfor har det naturligvis ikke blitt noen voldsomme mengder med gruppearbeid denne dagen, men heller individuelt arbeid med det som ble avtalt og diskutert på mandag.

Onsdag har Krister ordinært vært på jobb og Kenneth opptatt med sin undervisning i webscience. Dermed ble Oskar og Martin som regel sittende med dokumentasjon eller lignende.

Torsdag har vi hatt møte med veileder Øyvind Eek Jensen klokke tolv og møte med oppdragsgiver rundt klokka to. Ellers generelt sett en normal arbeidsdag.

Fredag jobber både Krister, Martin og Kenneth, derfor har denne dagen vært satt av til hjemmearbeid.

Før første presentasjon forsto vi ikke fullstendig hvor viktig en god prosjektmodell var. Vi hadde opp til dette punktet brukt en som het «evolusjonær prototyping», der arbeidsmetodene gikk i en slags sløyfe til man hadde et ferdig produkt. Først avdekket man krav, så skulle det designes, deretter bygges og til slutt testes.

Dette passet veldig dårlig med måten vi hadde jobbet på måneden før presentasjonen vår. Etter samtale med veileder fant vi ut at vi var nødt til å bytte til en modifisert versjon av «unified process», da denne fremhevet den iterative arbeidsmetoden vi kom til å benytte mye bedre.

Vi delte ganske tidlig inn ansvarsområder for å ha det på plass så fort som mulig. På denne måten visste alle på gruppa litt bedre hva de kom til å arbeide med. Slik har arbeidsfordelingen sett ut;

Kenneth Bergerud	Prosjektleder, økonomi, web/informasjon og LabVIEW.
Krister Håkenrud	Gantt-diagram, research av servodriver, beregninger og IndraWorks.
Martin Ingvaldsen	Generell dokumentasjon, elektrisk oppkobling og koblingskjemaer i Visio.
Oskar Gunbjørnsen	Spesifikasjon, test, administrative dokumenter og IndraWorks.

Selv om vi har delt inn ansvarsområdene på denne måten betyr det på ingen måte at den personen kun skal gjøre den jobben. Det er bare et område som den personen må se til at blir ferdigstilt på en representativ måte i forhold til det vi har gjort.

Kenneth sine oppgaver som prosjektleder har gått ut på å ha et aktivt samarbeid og samtale med oppdragsgiver, som for eksempel å avtale møter, si ifra om forberedning av tester slik at alt utstyr vi trengte var på plass og diverse andre administrative ting. Økonomi har grovt sett handlet kun om innkjøp av servodriver. På websiden vår har Kenneth lagt ut informasjon om gruppa, gruppededlemmers bakgrunn og generelt om oppgaven.

LabVIEW er koblet til NI cRIO og det er her tabellene med posisjoner og kraft verdier blir lastet opp for så å bli behandlet og sendt videre. Dette har Kenneth også ansvar for.

Krister har under første fase arbeidet og vært ansvarlig for Gantt-diagrammet. Det har fungert som en overordnet tidsplan i motsetning til prosjektmodellen som ikke har noen tidsakse, kun milepæler. Valg av servodriver var en stor del av andre fase i prosjektet og Krister gjorde en stor jobb på denne biten. De andre elektro studentene leste og gjorde litt research de også, men hovedarbeidet kom fra Krister.

I IndraWorks setter man parametere for akselerasjon, hastighet og kraft. Man kan også regulere og behandle verdier fra posisjons- og kraft-tabeller i IndraLogic. Denne delen av programmet benytter diverse programmeringsmetoder basert på PLS, da alt fra Ladder diagram til Structural Text. Krister og Oskar har ansvar for research og programmering av dette.

Martin har ansvar for generelt sett alt av ferdigstilling og finpussing av dokumenter. Vi tenkte han passet best til denne jobben siden han er nøye og vil gjøre ting på helt riktig måte av natur. Han har også ansvar for den elektriske klargjøringen og oppkoblingen av utstyret vi skal bruke. Oppkobling av lampe for reset-bryter, tilpassing av kabler, generell feilsøking og tegning av koblingsskjemaer for den delen av skapet som har blitt endret, er blant annet noe av det han har bidratt med.

Oskar har hatt ansvar for spesifisering av krav og tester i fase 1. I kravspesifikasjonen beskrives seks forskjellige kategorier av ting vi stiller krav til. Vi har: clutchsimulator, girsimulator, software, sensorer, testing/sikkerhet og servodriver. I testspesifikasjonen beskriver vi hvordan kravene skal testes.

Generelt sett har han skrevet rapporter for de testene som har blitt utført og visse andre administrative dokumenter som blant annet møtereferater og oppfølgingsdokumentene hver uke.

4 – Utfordringer

Som sagt så var det administrative noe av det vanskeligste å sette seg inn i under første fase av prosjektet. Etter 1. Presentasjonen, stoppet prosjektet litt opp fordi vi ikke helt visste hva vi skulle gjøre videre. Prosjektmodellen vår var også dårlig utformet, så den var ikke mye til hjelp. Vi valgte å endre prosjektmodell, og dette gjorde arbeidet med prosjektet lettere.

I andre fase eller research fasen som fant sted mellom første og andre presentasjon, var den største utfordringen valg av servodriver. I første omgang av valget sto det mellom tre alternativer; parallell interface som KA allerede var i besittelse av, Sercos 3 og Profibus. Men det skulle vise seg å bli vanskeligere enn som så. Parallell interface alternativet gikk bort i og med at det ikke hadde mulighet til å håndtere den dynamiske kraften vi var ute etter.

Det andre alternativet som røyk var Sercos 3 på grunn av at National Instruments ikke har Sercos moduler til sitt utstyr. Dette fikk vi beskjed om rett etter valget var tatt sammen med KA, men heldigvis før det ble sendt bestilling.

Profibus var en mulig løsning, men dette alternativet ble altfor dyrt og måtte derfor også forkastes. Det siste alternativet kom vi opp med i et møte med KA hvor poenget var å benytte Sercos 3 uten busskommunikasjon. Alternativ 4 var det vi til syvende og sist gikk for, men siden det var litt uklart om dette faktisk ville fungere, måtte vi lese rundt dette også som tok ekstra tid.

Bestillingen ble sendt uka før påske, som betydde at de siste ukene etter vi fikk varene kom til å bli nokså intensive, men dette var vi klar over.

Tredje og fjerde fase glei litt inn i hverandre på den måten at noen ble sittende med dokumentasjon med en gang de var ferdige med for eksempel det praktiske de skulle gjøre. Det var rett og slett ikke tid til å gjøre alt det praktiske ferdig og deretter starte på den gjenværende dokumentasjonen.

Den desidert største hindringen her var PLS-programmeringen. Annet enn litt grunnleggende ladder programmering i developer på videregående skole, hadde ingen av oss vært i nærheten av noe lignende før. I tillegg slet vi lenge med en feil i IndraWorks som ikke lot oss skru på driveren, i den forstand at vi hverken kunne benytte «Jogging» funksjonen eller sette servomotoren til å kjøre kontinuerlig mellom to endepunkter.

Vi prøvde med feilsøking via multimeter for å dobbeltsjekke at alt var koblet i henhold til Visio tegninger, til ingen nytte. Søkte opp nummeret på feilen i manualene til IndraWorks og en stund på Google, men fortsatt samme resultat. Vi prøvde til og med å koble inn den forrige kontroll enheten og kjøre den opp mot IndraWorks DS siden dette hadde fungert på PCen til vår eksterne veileder tidligere, men ingenting endret seg da heller. Det var ikke før vi installerte programmet på nytt, og koblet opp stort sett alle ledningene som gikk inn til MLDen på nytt at feilen forsvant. Til dags dato er vi fortsatt ikke helt klar over hva som forårsaket dette.

Kommunikasjonen med IndraWorks var heller ingen lett oppgave nettopp fordi vi hadde nærmest null erfaring med programmet. Med den forrige kontroll enheten kjørte vi en test ved hjelp av IndraWorks DS, men der var alt koblet opp og i orden fra starten av så dermed var det ikke representativt for utfordringen vi hadde foran oss på noen måte.

Første gang vi testet servoen, klarte vi fint å styre den frem og tilbake fra IndraWorks DS. Etter vi byttet MLD, fikk vi ikke noe kommunikasjon fra servoen. Vi prøvde å feilsøke i koblingene, men fant ikke noe galt, så vi valgte å ta ut alle kontaktene og koble de inn igjen. Deretter fikk vi opprettet kommunikasjon, og servoen kunne styres fra IndraWorks.

5 - Resultat

Vi har ikke kommet i mål med prosjektet i den grad vi ønsket. Vi er veldig nære ved å løse oppgaven, men det er programmeringen av PLS som har stoppet oss. I utgangspunktet skulle IndraWorks lese girgraf fra CSV fil for å finne referansekraft, måle aktuell kraft fra lastcelle, regulere dette og tilføre ny motkraft. Som en erstatning valgte vi å gjøre lese filen i LabVIEW og beregne referansekraften der og sende denne verdien videre til MLD.

Vi har kun grunnleggende erfaring med PLS, men dette begrenser seg til ladder diagrammer. For å få vårt system til å kjøre var vi nødt til å sette oss inn i et nytt språk som heter "structuraltext", noe som har vært en utfordring på grunn av den korte tiden mellom når vi fikk programmet og leveringstiden.

Det gjenstår også litt med LabVIEW programmet. Kalibreringen fungerer ikke helt som den skal men vi veit hva feilen er. Grunnet tiden har vi ikke vektlagt dette før innlevering av dokumenter, men vi tror feilen er fikset innen presentasjonen. Vi mangler også XY grafen for aktuell motkraft og posisjon for å sammenligne med den leste girgraf.

Visuelt sett vil det se ut som det gjenstår endel siden simulatoren ikke kjører, men det mangler bare litt på PLS biten før den fungerer. Vi er fornøyde med det vi har fått gjort så langt og vi setter alle kluter til for å få en fungerende simulator før presentasjonen.

I Gantt-diagrammet har vi ikke klart å fylle alle timene som er satt opp, dette er fordi vi kommer til å jobbe videre med prosjektet frem mot fremføringen, og henter inn de resterende timene da. Alle har ikke like mange timer, dette er fordi tre av oss i tillegg til prosjektet, har vært avhengig å jobbe ved siden av.

6 - Egenvurdering

Denne oppgaven har hatt et enormt læringsutbytte. Det er første gang de fleste av oss har jobbet med et prosjekt av denne dimensjonen, både når det gjelder teknisk utførelse og prosjektadministrasjon. Vi har fått et bra innblikk i hvordan et prosjekt styres og utføres på et profesjonelt nivå. Når det gjelder estimering av tidsbruk på ulike aktiviteter var det nærmest rein gjetting. Etter dette prosjektet vil det også være gjetting, men da mer kvalifisert. Det å jobbe etter en prosjektmodell var også en utfordring i starten, men vi fant til slutt ut at vi hadde valgt feil modell så vi byttet. Etter dette ble det mye lettere å følge prosjektmodellen og produktiviteten økte.

Likevel har de største problemene vært av teknisk natur. Vi har alle jobbet med ting vi ikke har noen tidligere erfaringer fra og det har vært mye nytt.

Når vi fikk oppgaven av oppdragsgiver var det ikke en garanti på om dette faktisk var mulig å gjennomføre innenfor rimelighetens grenser. Vi visste at oppgaven var krevende, men den gir et veldig realistisk bilde av en oppgave man gjerne jobber med som ingeniør. Vårt fokus var ikke å få en lett gjennomførbar oppgave, men heller tvert imot en utfordrende slik at vi kunne få mest mulig lærdom ut av den som mulig.

Som nevnt i tidligere kapittel er ikke oppgaven helt ferdig utført. På tross av dette er vi ganske tilfredse med resultatet. Det har vært mange delmål underveis og vi mangler bare siste gjennombruddet for å få dette til å kjøre som vi vil.

Oppdragsgiver har stilt opp hele veien og har støttet oss faglig ved behov for dette, og vi har hatt bra kommunikasjon med veilederene, både eksternt og internt. Vi har hatt et veldig bra samhold i gruppa hele veien, og vi føler at alle har dratt sin del av lasset gjennom hele prosjektet. Det er lagt mye hardt arbeid i denne oppgaven og vi ser tilbake på denne tiden med masse positive erfaringer og nye kunnskaper.