
Hovedrapport

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAGSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Hovedrapport
VERSJON / DATO	1.0 / 22. Mai 2016



KONGSBERG

**AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER**



i. Abstrakt

Denne hovedrapporten for ACTEE prosjektet inneholder en innledning til hele prosjektet, etterfulgt av en samling selvstående rapporter, alle rapportene er skrevet i forbindelse med planlegging design og utvikling av ACTEE, rapportene er listet opp i dokumentoversikten. Samlet beskriver de hvordan vi har gjennomført Hovedprosjektet, både i forhold til prosjektstyring og i forhold til teknisk utførelse.

ii. Dokumentoversikt

Hovedrapport	1
Intensjonsavtale	2
Kontrolldokument	3
Prosjektplan	4
Sprint retrospektiv 1	5
Sprint retrospektiv 2	6
Sprint retrospektiv 3	7
Sprint retrospektiv 4	8
Software designdokument	9
Software teknologidokument	10
Teknologidokument WSS	11
End Effector	12
Teknisk dokument elektronikk	13
Designdokument mekanikk	14
Teknologier for mekanikk	15
Teknologidokument for grep	16
Risikoanalyse	17
Testverifikasjon og risiko	18
User manual	19
Økonomi	20
Timelister	21
Fremtidsdokument	22
Konklusjon	23

iii. Dokumenthistorie

Tabell 1: Dokumenthistorie

Versjon 0.1	20.02.16	Mal av Thomas.
Versjon 1.0	22.05.16	Sammendrag, innledning og konklusjon til hele rapportsamlingen for utviklingen av ACTEE

1 Innledning

Denne rapporten beskriver utviklingen av et system for utførelse av tester på CROWS M153 våpenstasjoner ved KPS. Systemet skal automatisere testene ved å utvikle et gripeverktøy til en Universal Robots 5 robotarm, og utvikle programvare for styring av denne, til den hensikt og effektivisere arbeidet med testing og på denne måten spare tid og penger. Prosjektgruppen ACTEE består av 2 ingeniør studenter fra hver av de tre studieretningene data, elektro og maskin fra ingeniør utdannelsen tilbudt ved HSN. Denne samlingen med dokumenter vil beskrive hvordan prosjektgruppen har gått frem for å utvikle det automatiske testsystemet, både fra et teknisk og et prosjektstyrings perspektiv. Vi har derfor valgt å dele rapporten opp på følgende måte

Dokument 1 til 8 beskriver prosjektets grunnleggende innhold og prosjekts gjennomføring med retrospektiv av de gjennomførte sprintene.

Dokument 9 til 15 omhandler den tekniske biten av prosjektet med design og teknologidokumenter for de respektive komponentene i systemet.

Dokument 16 er en risikoanalyse gir en overordnet oversikt over risikoer under prosjektet og en gjennomgang av hvordan vi har behandlet disse risikoene.

Dokument 17, testverifikasjon og risikodokumentet og viser hvordan vi har oppfylt de forskjellige user storiene og hvilke tester vi har utført. Dette er satt i en sammenheng med de risiko som er aktuelle for hver user story.

Dokument 18, User manual er en brukermanual både for montering av gripeverktøyet, programmering av Arduino og bruk av det medfølgende testprogrammet.

Dokument 19, Økonomisk oversikt gir en kort innføring i budsjettet, og en oversikt over kostnader knytte til prosjektets gjennomføring.

Dokument 20, Timelister er de faktiske timelistene vi førte under prosjektet.

Dokument 21, Fremtidsdokument er en samling av forslag til hvordan systemet kan forbedres.

INTENSJONSAVTALE

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAKSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Intensjonsavtale
VERSJON / DATO	2.0 / 20. Mai 2016
FORFATTER	Alle har vært involvert i skrivingen.
SIDEANTALL	6



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



i. Innholdsfortegnelse

Innholdsfortegnelse	2
i. Dokumenthistorie	3
1 Innledning	3
2 Intensjonsavtale	4
2.1 Partene.....	4
2.2 Hensikt	4
2.3 Prioriteringer.....	4
2.4 Betingelser	4
2.5 Fremdriftsplan	4
2.6 Roller og ansvarsområder	5
2.6.1 ACTEE.....	5
2.6.2 KPS.....	5
2.6.3 HSN	5
2.7 Endringshåndtering	5
2.8 Økonomi.....	6

ii. Dokumenthistorie

Tabell 1: Dokumenthistorie

v 0.1 09.01.16	Disposisjon av Thomas.
v 0.2 10.01.16	Sammskriving av innhold, alle bidro.
v 0.3 11.01.16	Renskrevet av Bård
v 0.5 15.01.16	Renskrevet av Aina og Monica
v 0.9 16.01.16	Korrekturlest av alle
v 1.0 16.01.16	Format innholdsfortegnelse og dokumenthistorie av Thomas
v 1.1 08.02.16	Oppdatert mal Thomas.
v 2.0 20.05.16	Espen: Korrektur og ordnet tabeller, overskrifter.

1 Innledning

Prosjektets formål er å utvikle en prototype for et gripeverktøy til en UR5 robotarm og software for utførelse og verifikasjon av tester på våpenstasjon CROWS M153.

Forprosjektets formål er å synliggjøre gruppens forståelse for hva Kongsberg Protech Systems som oppdragsgiver ønsker og forventer. Dette er også en mulighet for KPS til å komme med tillegg til de krav som allerede er stilt, samt avdekke eventuelle feiltolkninger eller mangler som synliggjøres her.

Under forprosjektet har det også blitt opprettet nødvendig dokumentstruktur og kommunikasjonskanaler.

2 Intensjonsavtale

2.1 Partene

Partene i prosjektet. Leverandør, ACTEE (studentgruppe). Observatør, HSN (studiested). Oppdragsgiver, Kongsberg Protech Systems.

2.2 Hensikt

Kongsberg Protech Systems ønsker automatisering av sine tester ved ny programvare på våpenstasjonene. Hensikten med prosjektet er å utvikle et gripeverktøy som kan integreres med Universal Robot type UR5, og et system for testing og verifikasjon av disse testene. Oppdragsgiver ønsker at dette systemet skal kunne utføre deler av, eller hele testsekvenser på CROWS M153 og med dette være både tids - og ressursbesparende.

Under forprosjektet finner vi flere ferdige gripeverktøy tilgjengelige, som er kompatible med denne robotarmen. På bakgrunn av dette ønsker vi å nevne at dersom prosjektets hensikt skulle være å snarest mulig effektivisere dagens mer tidkrevende testprosess, kunne en god løsning være å velge et allerede ferdig produkt, for så å tilpasse produktet.

2.3 Prioriteringer

Vi vil på bakgrunn av oppdragsgivers prioriterte behov og ønsker, kategorisere kravene som fremkommer i henholdsvis A, B og C - krav.

2.4 Betingelser

Prosjektgruppen vil rapportere status på milepælplan annenhver uke til ekstern veileder, vil ellers ha løpende kommunikasjon etter behov.

2.5 Fremdriftsplan

En fullstendig fremdriftsplan vil foreligge i sin helhet på 1. presentasjon.

2.6 Roller og ansvarsområder

Tabell 2: Ansvarsområder

Thomas Wegener	Data (hardware)	Ledelse, fremdrift, dokumentasjon og digitalteknikk
Monica Nadia Haugen	Maskin	Stakeholders, dokumentasjon, mekanisk design
Aina Engen Nilsen	Maskin	Risk, rapporter, sekretær, mekanisk design
Carl Martin Mauseth	Elektro	Quality, test spesifikasjon
Espen S. Kraglund	Data (software)	Webdesign, blogg, økonomi og software
Bård Simen Hamborg Enget	Elektro	El - ansvarlig, tidsbudsjett, oppdatering kravspesifikasjon

2.6.1 ACTEE

Teamet har sammen et ansvar for å grundig dokumentere alle observasjoner, avgjørelser og oppgaver som utføres. Prosjektet skal følge en objektorientert arbeidsmodell, og dokumentasjon skal produseres i henhold til retningslinjer fra skolen og bestemt av gruppen.

2.6.2 KPS

KSP vil ha en eller flere personer tilgjengelige som kan veilede oss undersveis i prosessen ved behov.

2.6.3 HSN

HSN vil stille med intern veileder og intern sensor.

2.7 Endringshåndtering

I de fleste prosjekter vil det oppstå ønsker om endringer av prosjektets omfang og/eller tidsplan. Det er ikke alltid det er hensiktsmessig og gjøre endringer i siste liten. Prosessen beskrevet under vil i dette prosjektet bli anvendt for håndtering av slike endringsønsker. Prosessen vil legge grunnlaget for en riktig vurdering av kostnader, nytteverdi, prioriteringer og konsekvenser for fremdriftsplan for hvert enkelt endringsønske. Vi vil beregne endringens omfang og vurdere tidsbruk i forhold til implementering opp mot tilgjengelige ressurser, og deretter gjøre en vurdering om det er hensiktsmessig og godkjenne, utsette eller avvise dem. Endringsønskene med vurderingsprosess vil bli dokumentert. Og tilbakemelding vil alltid sammenfalle med

ukesavslutning, senest 2 uker etter at endringsforespørselen er mottatt. Hensikten med endringshåndteringen er å systematisere de innspill som tilkommer underveis i prosjektet, for så å kunne treffe hensiktsmessige beslutninger for hovedprosjektet. Endringer som eventuelt blir utsatt, vil komme med i prosjekt evalueringsprosessen, og kan komme som forslag til forbedringer ved eventuell videreutvikling.

2.8 Økonomi

Det vil her være naturlig at det forekommer enkelte utgifter relatert til prosjektet, og gruppen er forespeilet et budsjett fra KPS med en maksgrense på 20 000 NOK. Disse midlene er øremerket prosjektet, primært utgifter til prototype, deler, 3D - printing og plakat. ACTEE er inneforstått med at alle utgifter på forhånd skal godkjennes av oppdragsgiver. Prosjektgruppen har også sagt seg villig til å bidra med inntil 3000,- av egne midler. Et fullstendig budsjett vil settes opp.

Signatur oppdragsgiver

Ved å godkjenne denne avtalen, bekrefter jeg også å ha gjort meg kjent med innholdet i Kontrollokumentet og godkjenner dette, eventuelt med avtalte endringer. Som bekreftelse på eventuelle endringer vil revidert dokument sendes ut for aktivering.

KONTROLLOKUMENT

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAKSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Kontrolldokument
VERSJON / DATO	3.0 / 20. Mai 2016
FORFATTER	Thomas, Monica, Aina og Bård
SIDEANTALL	5



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



i. Innholdsfortegnelse

i. Innholdsfortegnelse	2
ii. Oversikt tabeller	2
iii. Dokumenthistorie.....	3
iv. Forkortelser	3
1 Innledning	4
2 Systemet.....	4
3 Systemfunksjoner	4
4 Bruksmiljø	5
5 Dokumentasjon	5

ii. Oversikt tabeller

Tabell 1: Dokumenthistorie.....	3
Tabell 2: Forkortelser	3

iii. Dokumenthistorie

Tabell 1: Dokumenthistorie

V 0.1 09.01.16	Disposisjon av Thomas
v 0.2 10.01.16	Sammskriving av innhold, alle bidro.
v 0.3 11.01.16	Renskrevet av Bård.
v 1.0 11.01.16	Format innholdsfortegnelse og dokumenthistorie av Thomas
v 1.9 16.01.16	Revisjon av dokumentet av Monica og Aina
v 1.9.5 16.01.16	Forkortelser lagt inn av Carl Martin
v 2.0 16.01.16	Format innholdsfortegnelse og dokumenthistorie av Thomas
v 2.2 20.05.16	Espen: Ordnet tabeller, header og ferdigstilt
v 3.0 22.05.16	Ferdigstillt

iv. Forkortelser

Tabell 2: Forkortelser

Forkortelser	Forklaring
HSN	Høgskolen i Sørøst-Norge
ACTEE	Automated CROWS Testing End Effector
EE	End Effector, gripeverktøy
CG	Control Grip, styringsjoystick for våpenstasjon
DCP	Display Control Panel
CROWS	Common Remotely Operated Weapon Station
UR5	Universal Robots 5 robotarm

1 Innledning

Dette dokumentet inneholder en samling av de punktene som er nevnt i innledende møter og samtaler. Dette er gjort for at vi skal være sikre på at vi dekker bruksbehovet.

Dokumentet er ment som et kontrolldokument der det forhåpentlig avdekkes eventuelle feiloppfatninger, og disse korrigeres. Vi vil etter gjennomgang bruke dette dokumentet, eller en revidert utgave, som et offisielt grunnlag for kravspesifikasjonen.

2 Systemet

Det skal utvikles et gripeverktøy som skal kunne monteres på en eksisterende UR5 robotarm. Verktøyet blir heretter omtalt som EE, End Effector.

Robotarm skal kunne kommunisere med tidligere utviklet software på testmaskinen, samt fungere som bindeledd mellom EE og testmaskinen.

Testmaskin skal styre systemet og loggføre alle hendelser, sensordata og resultater.

Sensorer plassert på WS skal rapportere tilbake til testmaskin ved endringer i monitorert data. Monitorert data inneholder registrering av avfyring, men kan utvides etter behov.

Systemet som en helhet skal ha fokus på muligheten for videreutvikling på et senere tidspunkt.

3 Systemfunksjoner

Hovedmålet til gripeverktøyet er at det skal kunne betjene CROWS M153 Hardware. Det skal kunne gripe CG og betjene både Palm og Trigger knappene på denne. Gripeverktøyet skal også kunne betjene DCP's vippebryter og knapper. Alle handlinger utført av systemet skal ta hensyn til utstyret det jobbes mot, og skal til en hver tid passe på at det ikke ødelegger eller skader utstyret.

Sekundært er det ønskelig at gripeverktøyet også skal kunne betjene de øvrige knappene på CG. Dette vil bli tatt hensyn til under utviklingen av verktøyet, slik at dette kan legges til senere i utviklingsprosessen uten å måtte gjøre en fullstendig ombygging.

Det er også et ønske om at gripeverktøyet skal kunne dra ut og putte tilbake pins fra break-out-box. Dette tas i så fall stilling til etterhvert da vi per dags dato ikke har nok informasjon om breakout boksen.

Det vil være mulig for systemet å lese av posisjonen til våpenstasjonen og robotarmen, samt lese av fyringssignalet på CROWS M153 via egenutviklede sensorer og signaler hentet fra

våpenstasjonen, robotarmen og gripeverktøyet. Dette for å kunne verifisere testene som skal kjøres.

Testmaskinen skal fungere som hjernen i systemet, og all funksjonalitet skal styres herfra. Kommandoer til UR5 robotarm og det utviklede gripeverktøyet vil bli sendt fra denne, og sensordata hentet fra eksterne sensorer vil bli samlet her for loggføring og rapportering.

Software på denne testmaskinen vil kunne kjøre fullstendige tester, den vil kunne logge handlingene til robotarmen, gripeverktøyet og sensorene koblet til systemet. Dette vil tilsammen muliggjøre en komplett rapport etter endt kjøring som skal være detaljert nok til å kunne konkludere om testen er bestått eller ikke.

Software skal være konstruert slik at det vil være mulig for KPS og legge til flere tester ved behov, men vil i første omgang fokusere på en test.

4 Bruksmiljø

Systemet skal brukes på KPS sin testlab, der den skal kunne avlaste en testoperatør.

5 Dokumentasjon

Vi skal produsere diagrammer og beskrivelser av kretser og programvare. 2D tegninger, 3D modell og BOM, vil sammen med en utfyllende brukerveiledning inngå i dokumentasjon for systemet.

Prosjektgruppen har ukentlige oppfølgingsmøter med intern veileder, samt på gruppenivå. Vi vil holde alle involverte parter ukentlig oppdatert via vår webside. Det vil også rapporteres annenhver uke til oppdragsgiver via ekstern veileder.

Vi har som mål å ende opp med en fungerende prototype, men med tanke på begrensninger i tid og ressurser, vil vi vektlegge de mest brukte funksjonene og kalle disse A-krav. Vi vil gjøre så godt vi kan for å komme i mål også med B-kravene som vil innbære en bredere funksjonalitet. I tillegg vil vi tilrettelegge for utvidelser som vil gjøre at systemet skal kunne oppfylle et sett med C-krav. C-kravene vil gi systemet full funksjonalitet og evnen til og utføre en komplett testsyklus og samtidig registrere alle ønskede utfall.

PROSJEKTPLAN

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAKSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Prosjektplan
VERSJON / DATO	2.0 / 20. Mai- 2016
FORFATTER	Aina Engen Nilsen
SIDEANTALL	24



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



i. Abstrakt

Prosjektplanen er det første dokumentet som vil gi ett innblikk i valg av prosjektmodell, Tidsestimering, planlegging og struktur av prosjektet. Etter å ha lest dette dokumentet vil du ha en forståelse av hvordan prosjektgruppa tenker å gjennomføre oppgaven, samt å sikre fremdrift gjennom prosessen.

ii. Innholdsfortegnelse

i. Abstrakt.....	2
iii. Vedleggs oversikt	4
iv. Oversikt tabeller	4
v. Oversikt figurer	4
vi. Dokumenthistorie.....	5
vii. Forkortelser	6
1 Innledning	7
2 Ansvarsområder.....	7
3 Bakgrunn	9
4 Interessenter	10
5 Mål for prosjektoppgaven	10
6 Mål for hovedprosjekt.....	11
7 Prosjektmodell	11
8 Rammer og avgrensninger	13
8.1 Økonomi	15
9 Gjennomføring	15
9.1 Egenutviklet oversiktspanel	16
9.2 Testgjennomføring.....	19
10 Risiko	20
10.1 Endringshåndtering og konsekvens	20
10.2 Grensesnitt	20
10.3 Kvalitetshåndtering	21
11. Prototyping.....	22
Referanser	23

iii. Vedleggs oversikt

Prosjektplan Vedlegg A 2.0: User Stories og Acceptance Criterias

Prosjektplan Vedlegg B 1.0: Risikoanalyse

iv. Oversikt tabeller

Tabell 1 Dokumenthistorie	5
Tabell 2 Forkortelser	6
Tabell 3 Actee	7
Tabell 4 Interessenter	10
Tabell 5 Sprint oversikt.....	14
Tabell 6 Milepæler	14
Tabell 7 Utgifter.....	15

v. Oversikt figurer

Figur 1 : Scrum prosess	12
Figur 2 : Tidslinje.....	16
Figur 3 : Egenutviklet oversiktspanel.....	17
Figur 4 : Utdrag aktiv sprint	18
Figur 5 : Utdrag US og AC	18
Figur 6 : Endringshåndtering	19
Figur 7 : Oversiktsbilde grensesnitt	21
Figur 8 : Illustrasjon prototype	22

vi. Dokumenthistorie

Tabell 1 Dokumenthistorie

Versjon 0.1	20.01.16	Mal av Thomas.
Versjon 0.2	21.01.16	Endringer Carl Martin, Espen, Aina Endringer Carl Martin og Espen
Versjon 0.3	28.01.16	Endringer Espen og Carl Martin Endringer Thomas, Carl Martin, Espen
Versjon 0.4	02.02.16 03.02.16	Endringer Aina; flyttet avsnitt, kommentert og tilpasset innhold. Endringer Carl Martin. Monica: korrektur, forslag, tillegg av tekst, endring/konsekvens, tillegg i innhold.
Versjon 0.5	04.02.16	Tillegg av Aina: (Se versjon 0.4.) Foreslått tekst modell Monica: tillegg tekst innledning, Interessenter. Korrektur Thomas: lesing og kommentarer, innspill. Definert mål, endret noen overskrifter.
Versjon 0.6	05.02.16	Aina: Oppdaterer gruppeinfo, disposisjon/ferdigstiller dok. (Ikke i mål) Espen: Korrektur, oppdatert interessenter, abstrakt Thomas: Lagt til Gjennomføring og Egenutviklet prosjektverktøy.
Versjon 0.7	06.02.16	Thomas: Gjennomlesing og kommentarer, rettet små skrivefeil. Bård Simen: Gjennomlesing og kommentarer, rettet små skrivefeil. Aina: Leser, kommenterer/ser på løsninger, fortsetter ferdigstilling/disposisjon
Versjon 0.8	07.02.16	Aina: gjennomlesing kommentarer, justeringer av formulering og avsnitt.
Versjon 0.9	08.02.16	Aina: Opprettet versjon 0.9. Setter inn nye forkortelser, flyttet økonomi innunder punktet avgrensninger. Nummerert avsnitt. Espen: Korrektur, små endringer. Monica: korrektur, risk, endring, kvalitet, punkt 9 Aina: FERDIGSTILLER.
Versjon 1.0	09.02.16	Thomas opprettet. 1.0 – Innholdsfortegnelse og formatering
Versjon 1.1	15.03.16	Aina setter inn referanser etter IEEE. Navnendring vedlegg. Bård Simen: Lagt til Vedlegg A i dokumentet.
Versjon 1.2	18.05.16	Oppdatert Vedlegg User Stories av Bård Simen
Versjon 2.0	20.05.16	Korrektur Carl Martin

vii. Forkortelser

Tabell 2 Forkortelser

Forkortelser	Forklaring
KPS	Kongsberg Protech Systems
HSN	Høgskolen i Sørøst-Norge
ACT [2]	Automated CROWS Testing (foregående bachelorgruppe)
ACTEE	Automated CROWS Testing End Effector
EE	End Effector, gripeverktøy
CG	Control Grip, styringsjoystick for våpenstasjon
DCP	Display Control Panel
CROWS	Common Remotely Operated Weapon Station
UR5 [1]	Universal Robots 5 robotarm
US	User Stories
AC	Acceptance Criterias
SM	Scrum Master
PO	Produkt Owner

1 Innledning

Dette prosjektet er siste del av vår bachelor grad ved Høgskolen i Sørøst-Norge, og er ment som en praktisk utfordring før arbeidslivet. Vi er en multidisiplinær gruppe som vil løse en krevende oppgave for Kongsberg Protech Systems. Vi skal utvikle et gripeverktøy kompatibelt med en UR 5 robotarm [1], en *End Effector*.

Prosjektet strekker seg fra januar til juni 2016, og er tiltenkt og være på 600 timer i gjennomsnitt per student.

Anskaffelse av oppgave og noen møter med oppdragsgiver ble foretatt i perioden Oktober - Desember 2015.



Denne prosjektplanen vil vise våre interesser hvordan vi tenker, og hvordan vi vil angripe oppgaven. Prosjektplanen vil bistå oss i planlegging av fremdrift og gjennomføring av prosjektet. I dette dokumentet forklarer vi om prosjektmodellen vår og viser en tidslinje med milepæler.

Dette prosjektet er en videreutvikling av ACT [2] sin bacheloroppgave fra 2014, medlemmene av denne gruppen jobber i dag på KPS og vil være gode ressurser for oss under utførelsen av prosjektet.

2 Ansvarsområder

For å sikre fremdrift på de forskjellige områdene av prosjektet, har vi valgt ut nøkkelpersoner med overordnet ansvar for fremdrift på sitt område. Utover dette vil vi jobbe sammen både med teknisk og strategi.

Tabell 3 Actee

	Thomas Wegener - Embedded Systems Scrum master Digitalteknikk - Kommunikasjon - Software
	Monica Nadia Haugen - Mekanikk/Produktutvikling Development team Systemdesign - Mekanikk

	<p>Aina Engen Nilsen - Mekanikk/Produktutvikling Development team Risiko - Dokumentasjon - Design</p>
	<p>Carl Martin Mauseth - Kybernetikk/Mekatronikk Development team Quality management - Visuell kommunikasjon - Elektro</p>
	<p>Espen S. Kraglund - Virtuelle systemer Development team Webansvarlig - Økonomi - Software</p>
	<p>Bård Simen H. Enget - Kybernetikk/Mekatronikk Product owner/ development team Sprint planning - User story backlog management</p>

3 Bakgrunn

Dette prosjektet er en videreutvikling av ACT sin bacheloroppgave fra 2014. Her ble det utviklet et Software system for utførelse og verifikasjon av testsekvenser på et CROWS M153 våpentårn. [2]

Kongsberg Protech Systems ønsker automatisering av sine tester ved ny programvare på våpenstasjonene. Det ønskes utviklet et gripeverktøy som kan integreres med Universal Robot type UR5 [1], samt et system for testing og verifikasjon av disse testene. Oppdragsgiver ønsker at dette systemet skal kunne utføre deler av, eller hele testsekvenser på CROWS M153 og med dette være både tids - og ressursbesparende.

Systemet som en helhet skal ha fokus på muligheten for videreutvikling på et senere tidspunkt.

4 Interessenter

Et prosjekt omgis av ulike interessenter. Primære interessenter er oppdragsgivere, investorer, brukere og deltakere i prosjektgruppa. Disse har direkte innvirkning på prosjektet i form av behov og ønsker om en løsning på en problemstilling. Sekundære interessenter spiller en mer indirekte rolle på selve funksjonaliteten, og kan påvirke i form av overordnede krav, som lover og regler fra myndigheter.

Tabell 4 Interessenter

Navn	Rolle	Mål
KPS	Oppdragsgiver	Effektivisering
Merethe Hjelsvold	Oppgaveformidler	Løse mål om effektivisering
Hege Skullestad	Ekstern sensor	Kvalitetsvurdering
August Kind Svendsen	Ekstern veileder SW	Veiledning/samarbeid
Alexander Kristoffersen	Ekstern veileder el.mek.	Veiledning/samarbeid
HSN	Høgskole	Kompetanselever
Karoline Moholth Mcclenaghan	Intern sensor	Kvalitetssikring
Sigmund Gudvangen	Intern veileder	Veiledning
Den Norske stat	Overordnet myndighet	Håndheve lov
Leverandører av deler	Vedlikehold	Næring/lønnsomhet
Testoperatører	Brukere	Brukervennlighet/avlastning

5 Mål for prosjektoppgaven

Vi skal effektivisere testingen av ny Software på CROWS M153 våpenstasjon. Oppgaven baserer seg på ACT - prosjektet, som vi viderefører ved å utvikle et verktøy som kan betjene styringen av våpenstasjonen i en testsituasjon, samt videreutvikle tilhørende Software for styring og logging av testene.

6 Mål for hovedprosjekt

Hovedprosjektet er avslutningen på vår ingeniørutdanning på Bachelornivå.

I rollen som ingeniør er det avgjørende å kunne anvende tilegnet teknisk kunnskap for å utarbeide nye løsninger eller optimalisere eksisterende løsninger. Som oftest dreier dette seg om sammensatte systemer bestående av bred teknologi innenfor flere spesialfelt.

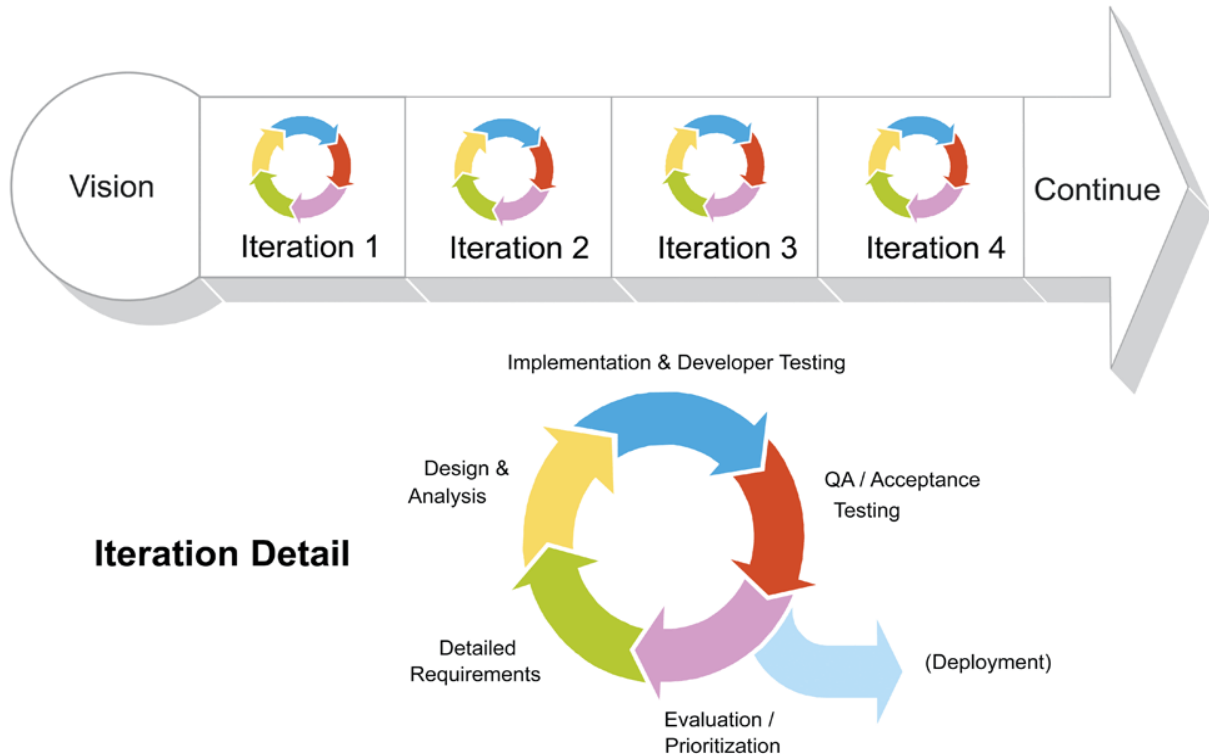
I arbeidslivet vil dette bety at flere ingeniører, ofte med ulike fagområder, jobber i prosjektgrupper for å løse ulike problemstillinger.

Vårt mål er å praktisere denne typen arbeidsform gjennom et samarbeid i en dynamisk og velfungerende gruppe. Vi ønsker å høste erfaringer ved å gjennomføre de ulike fasene i prosjektet strategisk med bruk av grundig dokumentasjon basert på prosjektmodell og ulike verktøy for prosesser. Ved å gjøre dette parallelt med utvikling av tekniske løsninger på tverrfaglig nivå, har vi mulighet til å modnes og bli tilstrekkelig kvalifisert til arbeidslivet. Vi vil øke vår kompetanse til å fungere både individuelt og i samarbeid med andre.

Prosjektgruppen er svært motivert for å løse problemstillingen fra oppdragsgiver i form av samarbeid gjennom lærerike og spennende arbeidsprosesser og presentasjoner.

7 Prosjektmodell

Vi har valgt Scrum som vår prosjektmodell. Dette er en agile modell som ofte brukes på Software prosjekter, men som også med fordel kan brukes på mer sammensatte systemer.



Figur 1: Scrum prosess

SCRUM's grunnprinsipp bygger på at komplekse prosesser håndteres best ved empirisk tilnærming (synlighet, inspeksjon og tilpasning). Prosjektmodellen baserer seg på et selvorganiserende, tverrfaglig team hvor ingen overordnet leder alene er hovedansvarlig. Som figur 1 illustrerer er prosessen inkrementell og iterativ, dette er hensiktsmessig for vårt prosjekt ved at vi hurtig kan ta frem nye prototyper slik at vi kan sysselsette alle fagretninger tidlig i prosjektet, samt fortløpende oppdage mangler/svakheter. Også fordi omfanget av oppgaven ikke samsvarer med tid til rådighet i prosjektet, vil denne modellen føre frem til et fullt fungerende produkt innenfor *enkelte* krav - i stedet for et nesten - fungerende produkt innenfor mange krav. Produktet vil være tilrettelagt for videreutvikling i stedet for å ha behov for omgjøring eller ferdigstilling ved leveranse.

Modellen er ressurseffektiv i forhold til dokumentasjon, da den er objekt orientert fremfor dokumentorientert.

De fleste SCRUM - prosjekter har produktet som fokus, etter innspill fra intern sensor har vi valgt og se på *prosjektet* som fokus for SCRUM - prosessen og produktet vil komme som en del av prosjektet. Dette er hensiktsmessig da det er mye dokumentasjon og andre elementer som kreves fra skolen.

I denne prosjektmodellen fremstilles krav i form av *User Stories* - interessenters krav og ønsker for produktet formulert som en ønskeliste;

“ Som oppdragsgiver vil jeg at produktet skal ha en levetid på minimum 15 år “

“ Som forbruker vil jeg at produktet skal tåle opp mot 80 grader celsius“

Hver US, når valgt til en sprint, får ett eller flere *Acceptance Criteria* som brukes til å forsikre at hver enkelt US er ferdig. Disse AC er testspesifikasjonen for SCRUM metoden, og er måten vi benytter for å forsikre at produktet blir testet. Eksempler på AC er:

“ slik at bedriften kan forsvare produksjonen økonomisk.”

“..... Slik at jeg kan ha produktet med meg i en badstue”.

For utfyllende tabell over US og tilhørende AC, **se Vedlegg A**.

Ved å bruke US som et verktøy, oppnår vi at interessenter klarer å visualisere systemets funksjonalitet og kapasitet. Dette gir bedre grunnlag for en tidlig validering.

8 Rammer og avgrensninger

Prosjektet har avgrensninger vi må ta hensyn til for å hindre at prosjektet flyter ut. Først og fremst vil vi måtte forholde oss til tidsbegrensning (**se punkt 7**), men også til ressursbegrensning i form av økonomi og kompetanse. Tydelige rammer settes også av oppdragsgiver i form av krav og ønsker til produkt (**se Vedlegg A**). Bachelorprosjektet har også føringer gitt av høyskolen som vi må forholde oss til.

Prosjektoppgaven er som nevnt tidsstyrt med endelig sluttdato, det vil være nødvendig med god oversikt og tidsestimering for å nå målet. Dette oppnås best ved at vi tidsestimerer oppgaver, og har møter underveis for å korrigere våre estimer.

For å ha kontroll på tidsbruk registrerer vi timene vi bruker på prosjektet kontinuerlig.

Tidsplanen bygger på prinsippene angitt av prosjektmodellen, prosjektfasene er sekvensielt strukturert (**se punkt 7**).

Tabell 5 Sprint oversikt

Sprint	Tidsrom	Innhold
0	04.01 - 29.01	Research, idemyldring, sprintplanlegging.
1	29.01 - 12.02 12.02 - 19.02	US, AC, tidsplan, 1. presentasjon og estimering. Sprint review og planlegging.
2	19.02 - 18.03 18.03 - 24.03	Defineres etter foregående sprint review. Sprint review og planlegging.
3	09.04 - 02.05 02.05 - 04.05	Defineres etter foregående sprint review. Sprint review og planlegging.
4	04.05 - 19.05 19.05 - 23.05	Defineres etter foregående sprint review. Sprint review og planlegging.
5	23.05 - 08.06	Defineres etter foregående sprint review. Prosjekt slutt.

For å sikre prosjektets fremdrift, vil vi ha en rekke milepæler underveis (tabell 6)

Tabell 6 Milepæler

Dato	Delmål
12.02.16	1. Presentasjon
18.03.16	2. Presentasjon
17.05.16	Innlevering dokumentasjon
23.05.16	Siste frist innlevering dokumentasjon
08.06.16	3. Presentasjon

8.1 Økonomi

Vi har utarbeidet et førsteutkast til budsjett. Vi har få konkrete tall og forholde oss til på nåværende tidspunkt, så vi har prøvd å estimere utgifter mens vi forholder oss til den oppgitte budsjettrammen på 20000 kroner vi har blitt forespeilet fra KPS. Vi har også et buffer på 3000 kr som prosjektgruppen er villig til å bidra med selv om det blir nødvendig. Budsjettet vil oppdateres ettersom gruppen kommer lengre i utviklingsfasen.

Tabell 7 Utgifter

Utgift	Beskrivelse	Budsjettert
Kontorutgifter	Papir, CD/DVD, Poster etc.	Kr 1000
Mek. Utvikling	Komponenter, 3D print,	Kr 10000
Elektro utvikling	Mikrokontroller, komponenter	Kr 5000
Webside	Betaling for domene	Kr 275
Presentasjonsutgifter	Kaffe og kaker	Kr 1000
Totale utgifter		Kr 17275

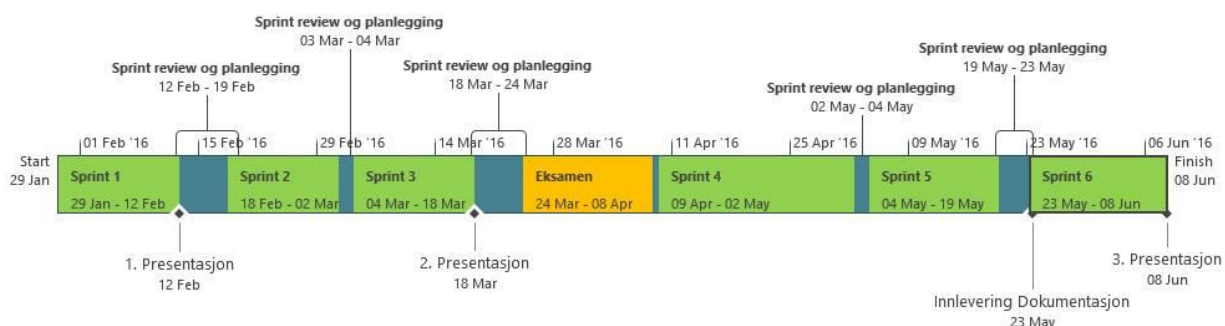
9 Gjennomføring

Planlegging og innledende møter startet allerede i september 2015. Vi fikk bekreftet oppgaven i november, og har i tiden frem til årsskiftet hatt flere møter med oppdragsgiver.

4. januar begynte vi planleggingen av oppgaven og det ble samlet informasjonen fra besøk, møter og kommunikasjon med oppdragsgiver. Vi skrev to dokumenter som en avslutning på vår forprosjektfase; et kontrolldokument og en intensjonsavtale. Videre fortsatte arbeidet med å definere krav fra både oppdragsgiver, skolen og studentgruppen.

Vi hadde i utgangspunktet tenkt å gå for en V-modell prosjektstruktur. Men det ble raskt klart for oss at vi måtte se oss om etter en mer agile modell. Etter mye research og en god innsats fra flere av gruppens medlemmer, endte vi opp med å gå for Scrum som prosjektmodell.

Ved å bruke Scrum skal vi ikke låse oss helt til en plan. På bakgrunn av at vi har et begrenset tidsrom for utførelse av oppgaven, har vi laget en stipulert oversikt over planleggingsfaser, sprinter og noen milepæler i prosjektet (figur 2).

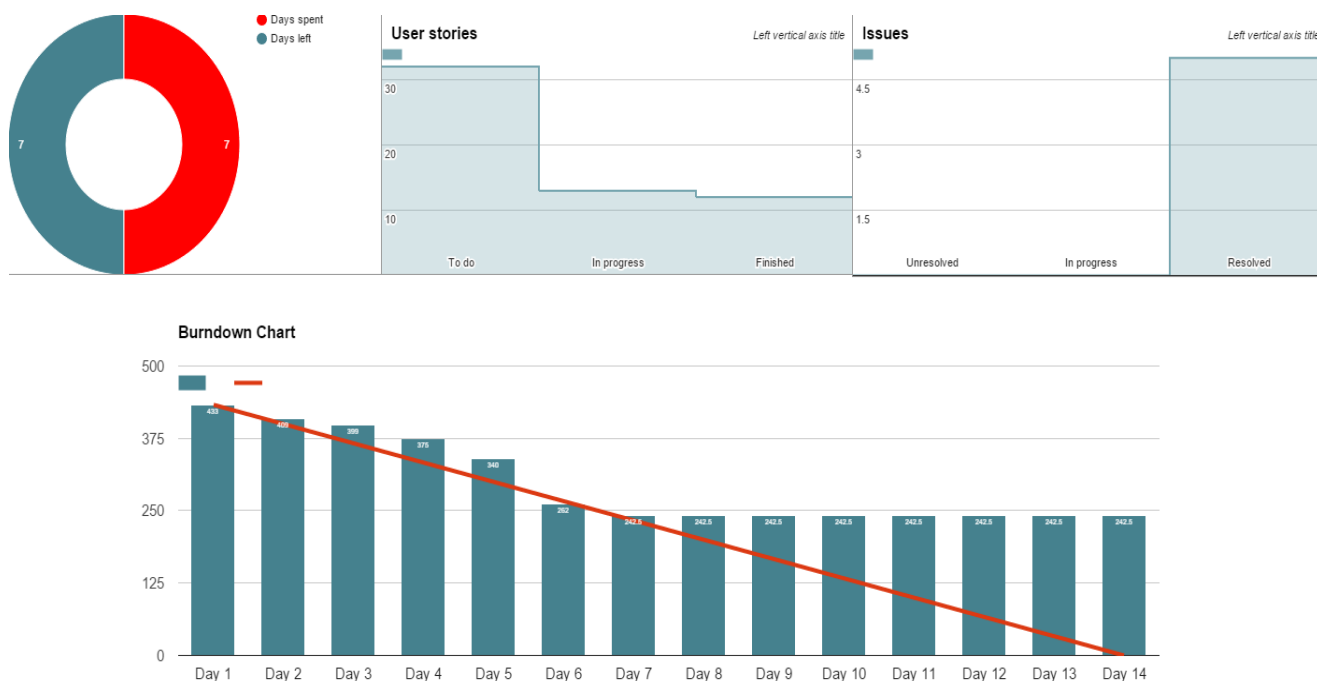


Figur 2: Tidslinje

Etter hver sprint vil hele teamet sammen med Product Owner og Scrum Master, avholde en sprintevaluering. Der vil vi gjennomgå endringer, utsettelse og nye oppdagelser. I tillegg vil vi evaluere kommunikasjon og effektivitet. Teamet bruker noe tid på research, og PO setter sammen en liste med oppgaver til neste sprint. Disse estimeres og vurderes av teamet, og gjennom forhandlinger kommer vi sammen fram til hva som skal med og hva som må utsettes. Vi har tatt i bruk visuelle diagrammer for å spore progresjon.

9.1 Egenutviklet oversiktspanel

Vi har laget et visuelt verktøy for å holde oversikt over progresjonen i hver sprint. For å kunne jobbe effektivt med de forskjellige oppgavene, var det ønskelig og ha et verktøy for å holde oversikt over utviklingen. Vi så på noen ferdig utviklede løsninger, men lisensene var kostbare og de billige løsningen var for dårlige. Derfor gikk vi for å sette av noen timer til å utvikle en egen regneark - basert oversikt (figur 3).



Figur 3: Egenutviklet oversiktspanel

Med denne oversikten kan vi raskt se hvor langt ut i sprinten vi er kommet. På det store søylediagrammet kan vi se progresjon i forhold til estimert gjenstående tid for summen av de resterende oppgavene tilknyttet sprinten. Den røde linjen illustrerer ideell utvikling.

US - diagrammet viser hvor mange av oppgavene som er som er påbegynt og slutført. Issues - diagrammet viser progresjon i forhold til ting som dukker opp underveis i prosjektet. Diagrammene henter verdiene til feltene fra et ark laget for å visualisere US som "kort" (figur 4).

9 [US-020210]	5	32:00	[Carl Martin, Espen, Aina, Thomas, Monica]	26:30
Som oppdragsgiver ønsker jeg at Prosjektplanen inneholder informasjon om hvordan prosjektet skal gjennomføres slik at vi får innblikk i hvordan gruppen har planlagt å jobbe.				
AC-020210-01 : Informasjonen om prosjektgjennomføringen er produsert.		AC-020210-02 : Føyd til informasjon, korrektur.		
AC-020210-02 : Informasjonen er lagt til i Prosjektplan dokumentet.				
10 [US-020220]	5	2:00		[Espen] 2:00
Som oppdragsgiver ønsker jeg at Prosjektplanen inneholder et økonomisk budsjett slik at vi får et overblikk over planlagte utgifter.				
		Flere av postene på budsjettet er estimeret.		
AC-020220-01 : Budsjettet er produsert.		AC-020220-01 : Budsjettet er produsert.		AC-020220-01 : Budsjettet er produsert.
AC-020220-02 : Budsjettet er lagt til i Prosjektplan dokumentet.		AC-020220-02 : Budsjettet er lagt til i Prosjektplan dokumentet.		AC-020220-02 : Budsjettet er lagt til i Prosjektplan dokumentet.

Figur 4: Utdrag aktiv sprint

Figur 4 inneholder 2 eksempler på hvordan vi fremstiller oppgavene i en aktiv sprint. Det øverste eksemplet er under arbeid, mens det nederste er ferdigstilt. I tillegg til å liste opp US, beskriver denne visualiseringen tilhørende AC for kvalitet og testing. Når en oppgave blir påbegynt blir feltene fylt ut med progresjon i tid, kommentarer og hvem som jobber med dem. Alle tidsestimat i US og prioriteringsverdier importeres fra sprint backloggen (figur 5).

EPIC	US-n	As a (hvem)	I want	So that	Tid	Priority	Acceptance
	US-010601	Som oppdragsgiver ønsker jeg at	Informasjon, kommunikasjon og arbeid relatert til oppgaven holdes konfidensielt	Da informasjon kan være av sensitiv natur	1:00	5	AC-010601-01 : Innhentet retningslinjer for publisering på web. AC-010601-02 : At vi får skriftlig godkjenning i forkant av presentasjon AC-010601-03 : Alle overholder taushetsplikten. AC-010601-04 : At vi får skriftlig godkjenning på rutiner for dokumentlagring
	US-010701	Som oppdragsgiver ønsker jeg at	Det skal vises prototyper	For tidlig og fortløpende V&V	1:00	5	AC-010701-01 : Prototype Godkjent av Product Owner. AC-010701-02 : Prototype er ferdig produsert. AC-010701-03 : Alle mål og dimensjoner er notert for videre bruk.
	US-051301	Som ingeniør ønsker jeg at	Den første prototypen skal ha en pekefinger.	For å tilrettelegge for at bevegelsesprogrameringen kan starte i sprint 2	16:00	4	AC-051301-01 : Første Prototype er utformet med en "pekefinger".
	US-051302	Som utvikler ønsker jeg at	Alle prototypene skal være like lange fra base til fingertupp	Slik at vi kan begynne programmering allerede i sprint 2	1:00	3	AC-051302-01 : Et estimat er gjort for alle nødvendige komponenter for fremtidig funksjonalitet AC-051302-02 : Prototypen er formet med plass estimatet i tankene.
	US-020101	Som oppdragsgiver ønsker jeg at	Systemet skal gjennomgående dokumenteres	For at vi skal kunne vise til hva vi har gjort, og at man kan fortsette arbeide der vi avslutter.	1:00	5	AC-020101-01 : Dokumentasjon for planlagte User Stories i Sprint1 er ferdigstilt, lagret og arkivert. Og CD og pe
	US-020102	Som intern veileder ønsker jeg at	Prosjektgruppa sender ukentlig skriftlig rapport	For å få oversikt over fremgang	6:00	5	AC-020102-01 : Skriftlig rapport er produsert. AC-020102-02 : Gruppa har godkjent rapporten. AC-020102-03 : Rapporten er sendt til intern veileder.
	US-020103	Som oppdragsgiver ønsker jeg at	Prosjektgruppa sender skriftlig status rapport hver 14 dag	For å vite hva som skjer	18:00	5	AC-020103-01 : Skriftlig rapport er produsert. AC-020103-02 : Gruppa har godkjent rapporten. AC-020103-03 : Rapporten er sendt til ekstem veileder.
	US-020201	Som oppdragsgiver ønsker jeg at	Det lages et prosjektplan dokument	For å få oversikt	1:00	5	AC-020201-01 : Prosjektplan er ferdig produsert. AC-020201-02 : Gruppa har godkjent prosjektplanen. AC-020201-03 : Bedriften har godkjent prosjektplanen. AC-020201-04 : Prosjektplan er sendt.
	US-020210	Som oppdragsgiver ønsker jeg at	Prosjektplanen inneholder informasjon om hvordan prosjektet skal gjennomføres	For å få oversikt	32:00	5	AC-020210-01 : Informasjonen om prosjektgjennomføringen er produsert. AC-020210-02 : Informasjonen er lagt til i Prosjektplan dokumentet.
	US-020220	Som oppdragsgiver ønsker jeg at	Prosjektplanen inneholder et økonomisk budsjett	For å få oversikt	2:00	5	AC-020220-01 : Budsjettet er produsert. AC-020220-02 : Budsjettet er lagt til i Prosjektplan dokumentet.
	US-020230	Som oppdragsgiver ønsker jeg at	Prosjektplanen inneholder en Overordnet Risiko vurdering	For å kunne se at gruppen har tatt høyde for risiko	16:00	5	AC-020230-01 : Risikovurdering er utarbeidet. AC-020230-02 : Risikovurdering er skrevet. AC-020230-03 : Risikovurdering er lagt til i Prosjektplan dokumentet.

Figur 5: Utdrag US og AC

Sprint backloggen er også en del av verktøyet vårt. Den består av en liste over alle US som er plukket ut av den store backloggen for å være en del av sprint 1. De er listet under hverandre, og før sprinten startes holdes en gjennomgang og en "planning poker" for å estimere tid det vil ta å utføre oppgaven. Tilslutt ser vi på tid til rådighet om noe må utsettes.

En siste del av sprint - oversikten omhandler endringshåndtering i forhold til US underveis i sprintene (figur 6).

[US-020230]		Resolved ▾	På møte
Denne oppgaven tar lengre tid en vi har stipulert, behov for å utarbeide en fulstendig vurdering av hele prosjektet, som så oppdateres kontunuerlig og inkluderes i hver status rapport.		Utvider antall timer avsatt til oppgaven. En oversikt over risk er en lønnsom investering i forhold til tidsbruk.	
[US-021702]		Resolved ▾	Thomas
Her bruker vi bare 16 av 32 timer. Vi har brukt denne posten til og kartlegge eksisterende kode fra ACT prosjektet, tegnet et enkelt USE case. men tiden har i hoedsak vært brukt til systemoversikt.		Her må det opparbeides litt mer C# kunnskaper før vi fortsetter	

Figur 6: Endringshåndtering

Hvis det oppstår et problem eller et endringsbehov i forhold til en av oppgavene i sprinten, opprettes en *Issue*. En Issue skal inneholde referanse til hvilken US det gjelder, en beskrivelse, og den skal merkes med *Unresolved* i dropdown - feltet. Den vil da dukke opp i diagrammet, og SM tar kontakt for å ta tak i situasjonen. Saken løses, og løsningen beskrives. Listen danner grunnlag for dokumentasjon, større saker kan ende med å generere endringsmelding.

9.2 Testgjennomføring

I testspesifikasjonen defineres selve verdien av å bruke US og AC. Disse metodene plukker kompleksiteten i prosjektet ned, og gir en nøye gjennomgang av de ulike kravene.

Tester, og gjennomføring av disse, styres av AC som blir valgt til hver US. Når US blir mer tekniske, vil AC også bli mer spesifikke for å sikre at alle hensyn er tatt.

En enkel US vil ikke kunne settes som ferdig før alle tester spesifisert i AC er ferdigstilt. Alle tester som er tekniske av natur skal dokumenteres fortløpende. Det vil opprettes et Testdokument, der testresultater for hver US loggføres.

US som er en del av en EPIC, en samling av US med tilhørighet til hverandre, vil ferdigstilles hver for seg. Tilslutt settes de sammen i en felles overordnet US. Denne US vil da ha ett nytt sett med AC som de skal testes mot, disse vil da være fokusert på EPICen sin funksjonalitet.

10 Risiko

Risiko beskriver potensialet for å miste noe av verdi som kan påvirke prosjektets mål, vi ser dermed viktigheten av en gjennomgående risikoanalyse. Med dette ønsker vi å gjøre prosjektet robust mot plutselige endringer, samt ha rask reaksjonstid på aktuelle og nødvendige tiltak. Gjennom prosjektet vil dette gi oss lærdom.

Formål er å avdekke flest mulig scenarier som kan oppstå i et prosjekt, og som vil ha en eller annen form for negativ innvirkning på resultatet. Å vite noe om risiko er viktig for å kunne fatte beslutninger, både for individer, i organisasjoner/bedrifter og på samfunnsnivå. Risikoanalyse er derfor et viktig fagområde. Når vi driver risikoanalyse, må vi ta utgangspunkt i en presis og operasjonaliserbar definisjon. En måte å gjøre dette på er å si at risiko er beregnet sannsynlighet for bestemte uønskede hendelser, og sårbarheten, eller konsekvensene, av slike hendelser. Den estimerte sannsynligheten er dermed et uttrykk for usikkerheten, både med hensyn til hvorvidt uønskede hendelser vil inntreffe, og med hensyn til hva konsekvensene i så fall vil bli [3].

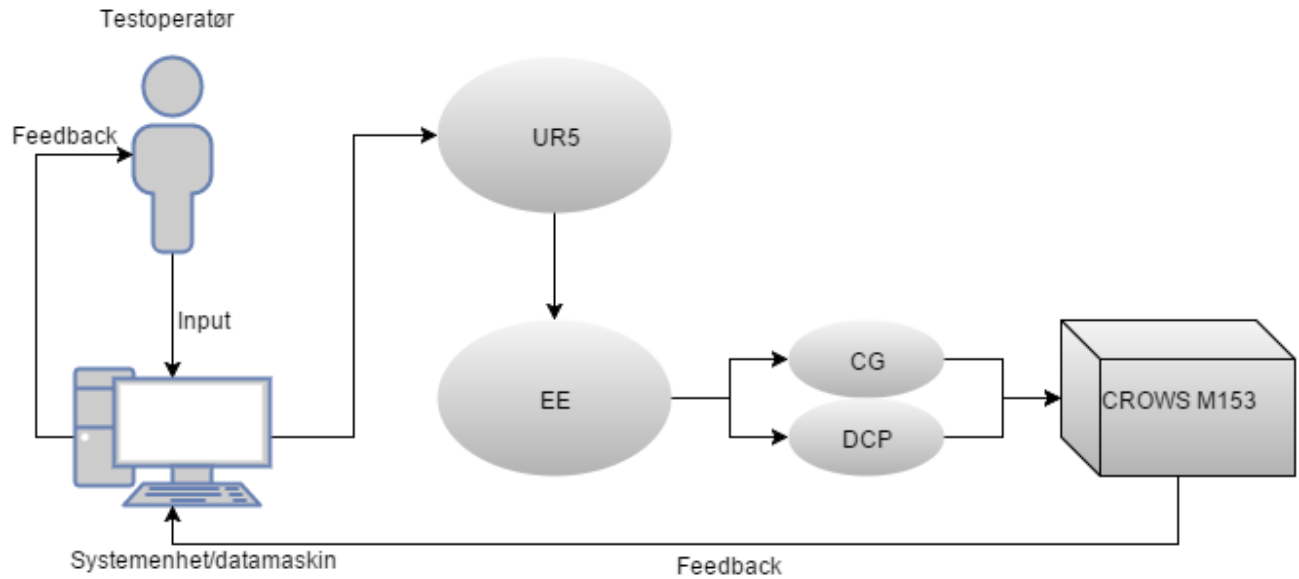
10.1 Endringshåndtering og konsekvens

Risiko reduseres ved ulike strategier. Veldefinerte risikoer er sårbare overfor endringer i løpet av utviklingen, på lik linje med krav. For eksempel kan systemets eksterne grensesnitt endres, eller krav om økt ytelse komme. På den måten kan variasjoner som kan endre krav, i dette eksempelet, bli evaluert med tanke på risiko og omfanget av endringen. En type endringshåndtering er å starte på nytt og re-prioritere.

Når endringer inntreffer, bør hele tiden kunne endre fokus inn mot det viktigste i fasen og dokumentere så detaljert som mulig hva slags løsning som er best, selv om vi ikke får gjennomført løsningen selv.

10.2 Grensesnitt

Vi anser grensesnitt som det mest usikre området i systemet, da problemer her både forventes å skje samtidig som det kan få store konsekvenser for systemet. Her er det mange rom for feil, at ting ikke fungerer som forventet. Vi vil derfor ha stort fokus på dette området, og vise gjennom kommende teknologi - dokumenter at vi har kontroll på de ulike grensesnitt (fig.7).



Figur 7: Oversiktsbilde grensesnitt

10.3 Kvalitetshåndtering

Vårt mål er å gjennomføre et prosjekt av god kvalitet, og dette oppnås ved å ha kvalitetsfokus som en rød tråd gjennom hele vårt prosjekt. For å sikre kvalitet i alle ledd vil gruppens kvalitetsansvarlig sørge for at gruppen tenker kvalitet i alle oppgaver.

Vi sikrer kvalitet ved:

- God og entydig kommunikasjon
- Prosjektoppfølgning og veiledning, internt og eksternt
- Riktig planlegging med strategiske verktøy
- Gjennomgående dokumentasjon, loggføring og sporing
- Kontroller og tester
- Iterasjoner og refleksjoner
- Risikovurderinger
- Fokus på sikkerhet
- Rett bruk av prosjektmodellen

11. Prototyping

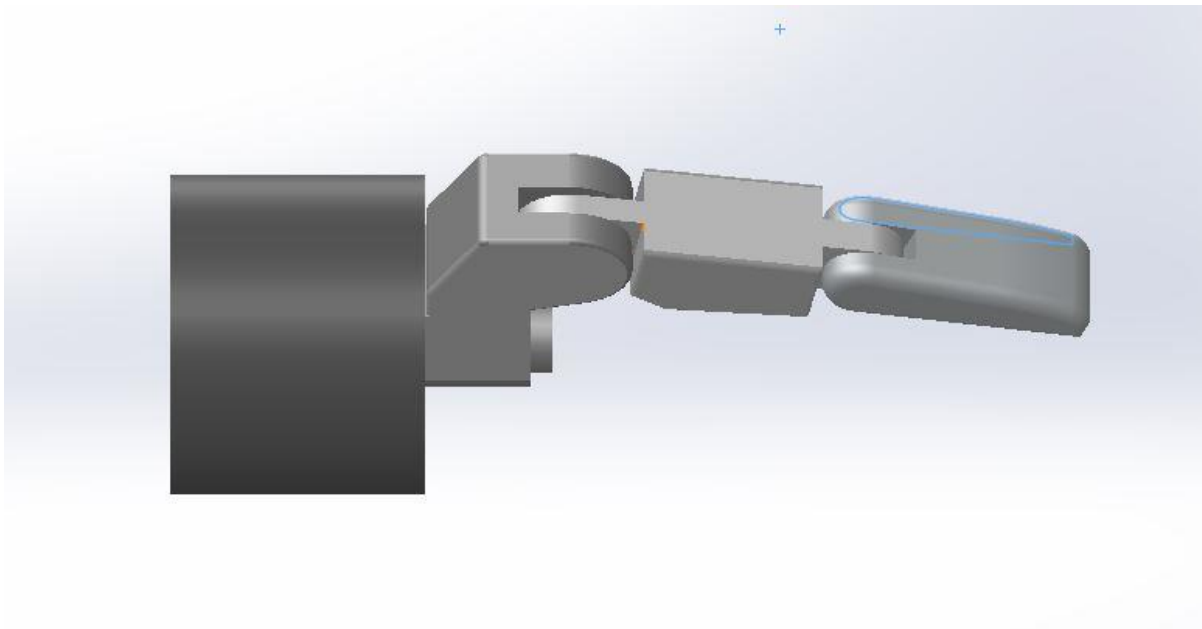
I henhold til vår prosjektmodell, har vi produsert en enkel illustrasjon av vårt fremtidige design av verktøyet.

Prototypens formål er å hjelpe oss å visualisere hvordan det endelige produktet skal bli, og å hjelpe oss til å se eventuelle feil eller forbedringer som må gjøres for å tilfredsstille kravene til gripeverktøyet.

Det er også hensiktsmessig å vise frem en fysisk modell på vår første presentasjon slik at våre interessenter lettere kan forstå hvordan vi har tenkt å løse oppgaven.

Vi har valgt å ta utgangspunkt i et eksisterende design som vi modifierer til å passe til vår oppgave. Dette er mest hensiktsmessig både med tanke på tid og resultat.

Ved å printe modellen i full skala, kan vi lettere avgjøre utforming og sammenstilling som behøves for de ulike komponentene



Figur 8: Illustrasjon prototype

Referanser

[1] Universal Robots AS. (2015). *UR5 - roboter*.

Tilgjengelig fra:

<http://www.universal-robots.com/no/produkter/ur5-robot/>

[2] A.K. Svendsen ; E.L. Roa ; H.B. Sørums ; S. Rudin. "*Automated CROWS Testing*" Bachelor Thesis, Fakultet for teknologi og maritime fag, Høgskolen i Buskerud og Vestfold, Kongsberg, 2014.

Tilgjengelig fra:

<http://hdl.handle.net/11250/216910>

[3] Publikasjon fra Geminisenteret, Risiko og sårbarhetsstudier ved NTNU og SINTEF.

"*Sikkerhet må skapes og gjenskapes hver dag. Det finnes ingen endelige løsninger*" [Online]

Tilgjengelig fra:

<http://docplayer.no/132672-Publikasjon-fra-geminisenteret-risiko-og-sarbarhetsstudier-ved-ntnu-og-sintef.html>

Prosjektplan Vedlegg A

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAKSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Prosjektplan Vedlegg A
VERSJON / DATO	2.1 / 18. Mai 2016
FORFATTER	Bård Simen Hamborg Enget
SIDEANTALL	45



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



i. Abstrakt

Dette dokumentet gir en fullstendig oversikt over alle User Stories og deres Acceptance Criterias som er en del av prosjektet så langt. User story backloggen er et «levende» dokument som vil vokse, og endre seg i løpet av prosjektperioden etter hvert som nye ønsker dukker opp, eller gamle krav endres eller utgår.

ii. Innholdsfortegnelse

i. Abstrakt.....	2
ii. Innholdsfortegnelse	3
iii. Dokumenthistorie.....	4
1 User stories.....	5
1.1 Ramme.....	6
1.2 Dokumentasjon.....	9
1.3 Sikkerhet	24
1.4 Funksjon.....	25
1.5 Design	36
1.6 Grensesnitt.....	39
1.7 Andre.....	41
1.8 Verifisering	44

iii. Dokumenthistorie

Versjon 0.1	03.02.16	Dokument opprettet og Layout laget av Bård Simen
Versjon 0.2	05.02.16 06.02.16	User Stories fra eget spreadsheet inn i layout av Bård Simen Skrevet Innledning av Bård Simen
Versjon 0.3	07.02.16	Lagt inn Layout forklaring av Bård Simen
Versjon 0.4	08.02.16	Ferdigstilt dokumentet, lagt klar til input/redigering av Bård Simen
Versjon 0.5	08.02.16	Redigert User Story layout forklaring, av Aina
Versjon 1.0	09.02.16	Formatering – Thomas
Versjon 2.0	15.02.16	Oppdatert User Stories av Bård Simen
Versjon 2.1	18.05.16	Oppdatert User Stories av Bård Simen

1 User stories

User stories er i dette dokumentet lagt inn i en tabell for best oversikt.

Øverste linje i hver tabell har en forklarende farge:

Grå: User Story tilhører foreløpig ikke en større gruppe.

Andre farger: gir User Stories tilhørighet i en større gruppe User Stories, også kalt en EPIC.

Tabellens felter (fra øverst til venstre) består av:

US-XXXXXX: Alle User Stories har fått en unik ID for sporbarhet.

Estimert tid: Dette er tid som er estimert til hver enkel User Story, De fleste her står tomme, da tid ikke blir estimert før User Story er en del av en sprint.

Prioritert: Alle User Stories blir tildelt en prioritet, dette gir en pekepinn på hvor viktig Story er vurdert i forhold til prosjektet. De kan bli rangert fra 1-5.

Sprint: Forteller oss hvilken sprint Story tilhører.

Andre linje er User Story, Skrevet i formatet Som Ønsker jeg at ..., slik at

Tredje linje er Acceptance Criterias for tilhørende User story. Når disse er oppfylt, og kun da, kan vi si at User Storien er oppfylt.

1.1 Ramme

US-010101	Estimert tid: 0	Prioritet: 4	Sprint: 5	
Som oppdragsgiver ønsker jeg at systemet skal kunne utføre testsekvenser, slik at bedriften kan spare tid og ressurser på testprosessen				
Acceptance Criteria/Kommentar: AC-010101-01 : EE skal kunne holde inn palm knapp, trykke trigger knapp, og bevege cg, samt kunne vippe bryter og trykke på knapper til DCP. Posisjon skal kartleggers og dokumenteres. AC-010101-02 : Dokumenter AC-010101-01.				
US-010201	Estimert tid: -	Prioritet: 4	Sprint:	
Som oppdragsgiver ønsker jeg at testsekvenser skal kunne defineres av en testoperatør, slik at han kan velge ut hvilke tester som skal kjøres.				
Acceptance Criteria/Kommentar:				
US-010301	Estimert tid: -	Prioritet: -	Sprint: -	
Som oppdragsgiver ønsker jeg at systemet skal kunne betjene CROWS M153 hardware, slik at testing av disse kan håndteres mest mulig automatisk.				
Acceptance Criteria/Kommentar: Overordnet Krav				
US-010401	Estimert tid: -	Prioritet: -	Sprint: -	
Som oppdragsgiver ønsker jeg at systemet skal designes med mulighet for fremtidig utvidelse, slik at vi kan gjøre flere endringer senere ved behov.				
Acceptance Criteria/Kommentar: Overordnet Krav				
US-010501	Estimert tid: -	Prioritet: -	Sprint: -	
Som oppdragsgiver ønsker jeg at systemet skal tilrettelegge for implementering av lavere prioriterte User Stories, slik at vi kan gjøre flere endringer senere ved behov.				
Acceptance Criteria/Kommentar: Overordnet Krav				
US-010601	Estimert tid: 2	Prioritet: 5	Sprint: 1	
Som oppdragsgiver ønsker jeg at informasjon, kommunikasjon og arbeid relatert til oppgaven holdes konfidensielt, slik at potensiell sensitiv informasjon ikke skal komme på avveie.				
Acceptance Criteria/Kommentar: AC-010601-01 : At vi får skriftlig godkjenning i forkant av publisering på web				

AC-010601-02 : At vi får skriftlig godkjenning i forkant av presentasjon

AC-010601-03 : Alle overholder taushetsplikten

AC-010601-04 : At vi får skriftlig godkjenning på rutiner for dokumentlagring

US-010701	Estimert tid: 67	Prioritet: 5	Sprint: ALLE
<p>Som oppdragsgiver ønsker jeg at det skal vises prototyper, slik at vi kan se hva gruppen har tenkt, og for tidlig og fortløpende V&V.</p>			
<p>Acceptance Criteria/Kommentar:</p> <p>AC-010701-01 : Prototype godkjent av Product Owner.</p> <p>AC-010701-02 : Prototype er ferdig illustrert i 3d tegning</p> <p>AC-010701-03 : Første del av EE design printes</p> <p>AC-010701-04 : Montering og integrering av servomotorer sammen med printet EE.</p> <p>AC-010701-05 : Dokumentering av AC-010701-01/02</p> <p>AC-010701-06 : Test av system sammen (EE og servomotorer)</p> <p>AC-010701-07 : Dokumentasjon av AC-010701-04.</p>			

1.2 Dokumentasjon

US-020101	Estimert tid: -	Prioritet: -	Sprint: -	
Som oppdragsgiver ønsker jeg at systemet skal gjennomgående dokumenteres, slik at vi kan se hva som har blitt gjort i prosjektet, og at vi kan fortsette arbeidet der gruppen avslutter.				
Acceptance Criteria/Kommentar: Overordnet Krav				

US-020102	Estimert tid: 24	Prioritet: 5	Sprint: ALLE	
Som intern veileder ønsker jeg at prosjektgruppa sender ukentlig skriftlig rapport, slik at jeg kan holde oversikt over fremgang.				
Acceptance Criteria/Kommentar: AC-020102-01 : Skriftlig rapport er produsert. AC-020102-02 : Gruppa har godkjent rapporten. AC-020102-03 : Rapporten er sendt til intern veileder. AC-020102-04 : Referat er skrevet og sendt intern veileder.				

US-020103	Estimert tid: 51	Prioritet: 5	Sprint: ALLE	
Som oppdragsgiver ønsker jeg at prosjektgruppa sender skriftlig status rapport hver 14.dag, slik at vi blir holdt oppdatert.				
Acceptance Criteria/Kommentar: AC-020103-01 : Skriftlig rapport er produsert. AC-020103-02 : Gruppa har godkjent rapporten. AC-020103-03 : Rapporten er sendt til ekstern veileder.				

US-020201	Estimert tid: -	Prioritet: -	Sprint: -	
Som oppdragsgiver ønsker jeg at det lages et prosjektplan dokument, slik at vi får oversikt over hva som er planlagt.				
Acceptance Criteria/Kommentar: Overordnet Krav				

US-020210	Estimert tid: 48	Prioritet: 5	Sprint: 1	
Som oppdragsgiver ønsker jeg at prosjektplanen inneholder informasjon om hvordan prosjektet skal gjennomføres, slik at vi får innblikk i hvordan gruppen har planlagt å jobbe.				
Acceptance Criteria/Kommentar: AC-020210-01 : Informasjonen om prosjektgjennomføringen er produsert. AC-020210-02 : Informasjonen er lagt til i Prosjektplan dokumentet.				

US-020220	Estimert tid: 8	Prioritet: 5	Sprint: 1	
-----------	-----------------	--------------	-----------	--

Som oppdragsgiver ønsker jeg at prosjektplanen inneholder et økonomisk budsjett, slik at vi får et overblikk over planlagte utgifter.

Acceptance Criteria/Kommentar:

AC-020220-01 : Budsjettet er produsert.

AC-020220-02 : Budsjettet er lagt til i Prosjektplan dokumentet.

US-020230	Estimert tid: 1	Prioritet: 5	Sprint: 1
-----------	-----------------	--------------	-----------

Som oppdragsgiver ønsker jeg at prosjektplanen inneholder en overordnet risiko vurdering, slik at vi kan se at gruppen har tatt høyde for risiko

Acceptance Criteria/Kommentar:

AC-020230-01 : Risikovurdering er utarbeidet.

AC-020230-02 : Risikovurdering er skrevet.

AC-020230-03 : Risikovurdering er lagt til i Prosjektplan dokumentet.

US-020231	Estimert tid: 4	Prioritet: 4	Sprint: 1
-----------	-----------------	--------------	-----------

Som risk ansvarlig ønsker jeg at risiko vurdering inneholder en Mekanikk risiko vurdering, slik at vi kan se at gruppen har tatt høyde for fagfeltets spesielle risk

Acceptance Criteria/Kommentar:

AC-020231-01 : Føringer er gitt til de fagansvarlige.

AC-020231-02 : Fagansvarlige har gitt tilbakemelding.

AC-020231-03 : Mekanisk Risikovurdering er lagt til i den overordnede risikovurderingen.

US-020232	Estimert tid: 4	Prioritet: 4	Sprint: 1
-----------	-----------------	--------------	-----------

Som risk ansvarlig ønsker jeg at risiko vurdering inneholder en software risiko vurdering, slik at vi kan se at gruppen har tatt høyde for fagfeltets spesielle risk

Acceptance Criteria/Kommentar:

AC-020232-01 : Føringer er gitt til de fagansvarlige.

AC-020232-02 : Fagansvarlige har gitt tilbakemelding.

AC-020232-03 : Software Risikovurdering er lagt til i den overordnede risikovurderingen.

US-020233	Estimert tid: 4	Prioritet: 4	Sprint: 1
-----------	-----------------	--------------	-----------

Som risk ansvarlig ønsker jeg at risiko vurdering inneholder en Elektro risiko vurdering, slik at vi kan se at gruppen har tatt høyde for fagfeltets spesielle risk

Acceptance Criteria/Kommentar:

AC-020233-01 : Føringer er gitt til de fagansvarlige.

AC-020233-02 : Fagansvarlige har gitt tilbakemelding.

AC-020233-03 : Elektrisk Risikovurdering er lagt til i den overordnede risikovurderingen.

US-020240	Estimert tid: 12	Prioritet: 5	Sprint: 1
-----------	------------------	--------------	-----------

Som oppdragsgiver ønsker jeg at prosjektplanen inneholder en oversiktlig fremvisning av User Stories , slik at vi lett kan se over User Stories og se hva gruppen har tenkt her.

Acceptance Criteria/Kommentar:

AC-020240-01 : layout for User Stories er utarbeidet.

AC-020240-02 : User stories er lagt inn i layout.

AC-020240-03 : User stories er lagt til i Prosjektplan dokumentet.

US-020241	Estimert tid: 2	Prioritet: 5	Sprint: 1
-----------	-----------------	--------------	-----------

Som oppdragsgiver ønsker jeg at User Stories har et prioriteringsfelt, slik at vi ser hvordan gruppen har prioritert User Stories.

Acceptance Criteria/Kommentar:

AC-020241-01 : Prioriterings felt er tilstede i User Story layouten.

US-020242	Estimert tid: 2	Prioritet: 5	Sprint: 1
-----------	-----------------	--------------	-----------

Som oppdragsgiver ønsker jeg at User Stories inneholder en kilde, slik at vi kan se hvor User Storien kommer fra.

Acceptance Criteria/Kommentar:

AC-020242-01 : Kilde felt er tilstede i User Story layouten.

US-020243	Estimert tid: 16	Prioritet: 5	Sprint: 1
-----------	------------------	--------------	-----------

Som oppdragsgiver ønsker jeg at User Stories som er planlagt i en sprint har acceptance criterias, slik at vi kan se hvordan gruppen bedømmer når en User story er ferdig

Acceptance Criteria/Kommentar:

AC-020243-01 : Alle User Stories planlagt i Sprint1 har Acceptance criteries.

AC-020243-02 : UserStories layout har felt for Acceptance criterias og disse er lagt til for Sprint1 UserStories.

US-020244	Estimert tid: 4	Prioritet: 5	Sprint: 1
-----------	-----------------	--------------	-----------

Som oppdragsgiver ønsker jeg at User Stories er merket med en unik identifikator, for å forenkle sporbarhet

Acceptance Criteria/Kommentar:

AC-020244-01 : Unik identifikasjon felt er tilstede i User Story layouten.

US-020250	Estimert tid: 16	Prioritet: 5	Sprint: 1
-----------	------------------	--------------	-----------

Som oppdragsgiver ønsker jeg at prosjektplanen inneholder et tidsestimat, slik at vi får et inntrykk av hva som skjer når.

Acceptance Criteria/Kommentar:

AC-020250-01 : Tidsestimat for Sprint 1 er utarbeidet.

AC-020250-02 : BurnDown chart er utarbeidet.

AC-020250-03 : Tidsestimat og BurnDown chart er tilrettelagt for å legges til i prosjektplan.
 AC-020250-04 : Informasjonen er lagt til i Prosjektplanen.

US-020251	Estimert tid: 4	Prioritet: 5	Sprint: 1
Som oppdragsgiver ønsker jeg at tidsestimatet har en tidsplan for fremtidige sprints, slik at vi kan se hva gruppen har estimert av fremtidige sprints.			
Acceptance Criteria/Kommentar: AC-020251-01 : En tidslinje med estimert planlegging blir utarbeidet. AC-020251-02 : Tidslinjen legges til i prosjektplanen.			

US-020252	Estimert tid: 4	Prioritet: 5	Sprint: 1
Som oppdragsgiver ønsker jeg at tidsestimatet har en oversikt over planlagt fravær og stillstand i prosjektet, slik at vi vet når det er planlagt at det skal være lite fremdrift på prosjektet.			
Acceptance Criteria/Kommentar: AC-020252-01 : Planlagt ferieperioder og eksamenslesing blir lagt til på tidslinjen.			

US-020301	Estimert tid: 36	Prioritet: 5	Sprint: 1
Som høyskole krever jeg at gruppen holder en 1. presentasjon, slik at vi kan bedømme fremgang i prosjektet.			
Acceptance Criteria/Kommentar: AC-020301-01 : Presentasjonens innhold er planlagt. AC-020301-02 : Presentasjonen fordeles. AC-020301-03 : Presentasjonen er sammenstilt. AC-020301-04 : Presentasjonen er gjennomgått.AC-020301-03 : Presentasjonen er sammenstilt. AC-020301-05 : Presentasjon overskrider ikke 20 minutter.AC-020301-03 : Presentasjonen er sammenstilt. AC-020301-06 : Presentasjon er ferdigstilt.AC-020301-03 : Presentasjonen er sammenstilt. AC-020301-07 : Tid og rom til presentasjon og møter er bestilt.AC-020301-03 : Presentasjonen er sammenstilt. AC-020301-08 : Veiledere, sensorer og publikum er invitert til presentasjon, og har godtatt.			

US-020302	Estimert tid: 16	Prioritet: 4	Sprint: 1
Som ingeniør ønsker jeg at 1.presentasjon inkluderer slides, slik at vi kan ha noe visuelt å vise til.			
Acceptance Criteria/Kommentar: AC-020302-01 : Slides er utarbeidet. AC-020302-02 : Slides er sammenstilt til en helhetlig presentasjon. AC-020302-03 : Presentasjon er godkjent av gruppen.			

US-020303	Estimert tid: 16	Prioritet: 4	Sprint: 1	
Som ingeniør ønsker jeg at alle medlemmene skriver et manus til 1.presentasjon, slik at et annet gruppemedlem kan steppe inn i tilfelle sykdom.				
Acceptance Citeria/Kommentar: AC-020303-01 : Alle gruppemedlemmer har utarbeidet er kortfattet manus. AC-020303-02 : Alle Gruppemedlemmers manus er lagt i 1.presentasjonsmappen.				
US-020304	Estimert tid: 36	Prioritet: 5	Sprint: 1	
Som ingeniør ønsker jeg at vi felles øver på hva vi skal si på 1.presentasjon, slik at vi felles kan sikre kvaliteten på det som sies.				
Acceptance Citeria/Kommentar: AC-020304-01 : Øvinger er gjennomgått. AC-020304-02 : Gruppen har godkjent resultatet.				
US-020305	Estimert tid: 16	Prioritet: 3	Sprint: 1	
Som høyskole ønsker jeg at første prototype vises frem på første presentasjon, slik at vi får synliggjort vår konsept ide				
Acceptance Citeria/Kommentar: AC-020305-01 : Prototype er vist frem på presentasjonen.				

US-020401	Estimert tid: 6	Prioritet: 5	Sprint: 3
Som høyskole krever jeg at gruppen holder en 2.presentasjon, slik at vi kan bedømme fremgang i prosjektet.			
Acceptance Citeria/Kommentar: AC-020401-01 : Presentasjon nr. 2 er holdt			
US-020402	Estimert tid: 12	Prioritet: 5	Sprint: 3
Som ingeniør ønsker jeg at 2.presentasjon inkluderer slides, slik at vi kan ha noe visuelt å vise til.			
Acceptance Citeria/Kommentar: AC-020402-01 : Slides er produsert AC-020402-02 : Slides er godkjent av gruppen			
US-020403	Estimert tid: 6	Prioritet: 5	Sprint: 3
Som ingeniør ønsker jeg at alle medlemmene skriver et manus til 2.presentasjon, slik at et annet gruppemedlem kan steppe inn i tilfelle sykdom.			
Acceptance Citeria/Kommentar: AC-020403-01 : Alle har lagt manus inn på google disk			
US-020404	Estimert tid: 36	Prioritet: 5	Sprint: 3
Som ingeniør ønsker jeg at vi felles øver på hva vi skal si på 2.presentasjon, slik at vi felles kan sikre kvaliteten på det som sies.			
Acceptance Citeria/Kommentar: AC-020404-01 : Øvinger er gjennomgått			
US-020501	Estimert tid: -	Prioritet: 5	Sprint: 6
Som høyskole krever jeg at gruppen holder en 3. presentasjon, slik at vi kan bedømme fremgang i prosjektet.			
Acceptance Citeria/Kommentar:			
US-020502	Estimert tid: -	Prioritet: 5	Sprint: 6
Som ingeniør ønsker jeg at 3.presentasjon inkluderer slides, slik at vi kan ha noe visuelt å vise til.			
Acceptance Citeria/Kommentar:			
US-020503	Estimert tid: -	Prioritet: 5	Sprint: 6

Som ingeniør ønsker jeg at alle medlemmene skriver et manus til 3.presentasjon, slik at et annet gruppemedlem kan steppe inn i tilfelle sykdom.

Acceptance Criteria/Kommentar:

US-020504

Estimert tid: -

Prioritet: 5

Sprint: 6

Som ingeniør ønsker jeg at vi felles øver på hva vi skal si på 3.presentasjon, slik at vi felles kan sikre kvaliteten på det som sies.

Acceptance Criteria/Kommentar:

US-020601

Estimert tid: 18

Prioritet: 5

Sprint: 5

Som høyskole ønsker jeg at gruppen lager en etteranalyse, slik at vi får en oversikt over gruppens tanker rundt prosjektet

Acceptance Criteria/Kommentar:

AC-020601-01 : Det utformes en etteranalyse der vi reflekteres over prosessen og eventuelle forbedringer utbedres.

US-020701

Estimert tid: 18

Prioritet: 5

Sprint: 5

Som høyskole ønsker jeg at hvert gruppemedlem lager et refleksjonsdokument, slik at vi får en oversikt over hva de enkelte medlemmene har gjort og hva de føler de har fått ut av prosessen.

Acceptance Criteria/Kommentar:

AC-020701-01 : Hvert enkelt gruppemedlem har laget en etteranalyse som legges ved dokumenteajonen.

US-020801

Estimert tid: 1

Prioritet: 5

Sprint: 3

Som høyskole ønsker jeg at gruppen produserer minimum et Teknologidokument for hvert fagfelt, slik at vi kan se at gruppen har gjort overveide valg i løpet av prosessen med tanke på produktets oppbygging og kapasitet.

Acceptance Criteria/Kommentar:

AC-020801-01 : Minst et teknologidokument er produsert

AC-020801-02 : Minst et teknologidokument er levert

US-020901

Estimert tid: 8

Prioritet: 5

Sprint: 5

Som høyskole ønsker jeg at gruppen lager en poster, slik at det finnes en visuell, kortfattet "reklame" for gruppen og oppgaven deres.

Acceptance Criteria/Kommentar:

AC-020901-01 : Design er utformet

AC-020901-02 : Design godkjent av PO.

AC-020901-03 : Innhold kvalitetsikkret og godkjent av hele gruppa.

US-021001	Estimert tid: 14	Prioritet: 5	Sprint: 3	
Som høyskole ønsker jeg at det lages et testverifikasjons dokument, slik at testresultater leveres samlet i dokumentasjonen.				
Acceptance Criteria/Kommentar: AC-021001-01 : Testverifikasjonsdokument er produsert AC-021001-02 : Testverifikasjonsdokument er levert				

US-021002	Estimert tid: 32	Prioritet: 5	Sprint: 4	
Som høyskole ønsker jeg at testverifikasjon dokumentet vedlikeholdes, Slik at det er mest mulig "up to date"				
Acceptance Criteria/Kommentar: AC-021002-01 : Komponenter og grensesnitt er styrkeberegnet/analysert der dette er naturlig. AC-021002-02 : Komponenter og grensesnitt er fysisk testet. AC-021002-03 : Testresultater er dokumentert og samlet for innskrivning i testverifikasjonsdokument.				

US-021003	Estimert tid: 64	Prioritet:	Sprint: 5	
Som høyskole ønsker jeg at testverifikasjons dokumentet ferdigstilles, slik at det er klart til å legges inn i den endelige dokumentasjonen				
Acceptance Criteria/Kommentar: AC-021003-03 : Testresultater er dokumentert, disponert og korrekturlest i testverifikasjonsdokumentet.				

US-021501	Estimert tid: -	Prioritet: -	Sprint: -	
Som oppdragsgiver ønsker jeg at dokumentasjonen skal inneholde mekaniske 2D-tegninger, slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.				
Acceptance Criteria/Kommentar: Overordnet Krav				

US-02150101	Estimert tid: -	Prioritet: -	Sprint: -	
Som bachelorgruppe ønsker vi at dokumentasjon skal inneholde mekanisk 2D tegninger for prototype sprint 2, slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.				
Acceptance Criteria/Kommentar: Slettet, ikke aktuelt				

US-021502	Estimert tid: -	Prioritet: -	Sprint: -	
Som oppdragsgiver ønsker jeg at dokumentasjonen skal inneholde mekaniske 3D-tegninger, slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.				

Acceptance Criteria/Kommentar:
Overordnet Krav

US-02150201	Estimert tid: 1	Prioritet: 4	Sprint: 2
Som bachelorgruppe ønsker vi at dokumentasjon skal inneholde 3D filer for prototype fra sprint 2, slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.			
Acceptance Criteria/Kommentar: AC-02150101-01 : 2D tegninger er lagt ved i teknologidokumentet			

US-021503	Estimert tid: -	Prioritet: -	Sprint: -
Som oppdragsgiver ønsker jeg at dokumentasjonen skal inneholde BillOfMaterials for all egenutviklet mekanikk, slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.			
Acceptance Criteria/Kommentar: Overordnet Krav			

US-021505	Estimert tid: 32	Prioritet: 5	Sprint: 2
Som bachelorgruppe ønsker vi at det skal utarbeides et grunnlag for et mekanisk teknologidokument, slik at det tilrettelegges for dyptgående forståelse av oppbygging og hensikt.			
Acceptance Criteria/Kommentar: AC-021505-01 : Ramme for dokumentet er produsert med oppsett og nødvendig punkter for innhold AC-021505-02 : Informasjon funnet i Sprint2 er lagt inn i dokumentet der det er relevant			

US-021506	Estimert tid: 30	Prioritet: 5	Sprint: 3
Som oppdragsgiver ønsker jeg at det skal utarbeides et end effector teknologidokument, Slik at det gis en dyptgående innsikt/forståelse av oppbygging og hensikt			
Acceptance Criteria/Kommentar: AC-021506-01 : EE teknologidokument er produsert AC-021506-02 : EE teknologidokument er levert AC-021506-03 : EE teknologidokument har med både elektro og mekanisk materiale			

US-021507	Estimert tid: 72	Prioritet: 5	Sprint: 4
Som oppdragsgiver ønsker jeg at end effector teknologidokument holdes ved like, Slik at det er mest mulig "up to date"			
Acceptance Criteria/Kommentar: AC-021507-01 : EE teknologidokument generere innhold AC-021507-02 : Teknologidokument for Design			

AC-021507-03 : Teknologidokument for Mekanikk
 AC-021507-04 : Teknologidokument for Elektronikk
 Note: Denne US-en vil bli nøyere dokumentert i sprint 5. Gjennom sprint 4 vil det bare genereres innhold til dokumentasjon.

US-021508	Estimert tid: 80	Prioritet: 5	Sprint: 5
Som oppdragsgiver ønsker jeg at end effector teknologidokument ferdigstilles, Slik at det er klart til å være med i innlevering 3			
Acceptance Citeria/Kommentar: AC-021508-01 : EE teknologidokument ferdigstilles AC-021508-02 : Teknologidokument for Design ferdigstilles AC-021508-03 : Teknologidokument for Mekanikk ferdigstilles AC-021508-04 : Teknologidokument for Elektronikk ferdigstilles			

US-021601	Estimert tid: -	Prioritet: -	Sprint: -
Som oppdragsgiver ønsker jeg at dokumentasjonen skal inneholde tegninger for alle egenutviklede elektriske kretser, slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.			
Acceptance Citeria/Kommentar: Overordnet Krav			

US-02160101	Estimert tid: 32	Prioritet: 4	Sprint: 2 til 4
Som bachelorgruppe ønsker vi at WS-sensors oppbygging dokumenteres med kretstegninger, slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.			
Acceptance Citeria/Kommentar: AC-021604-01 : Det er produsert korrekte skjemaer over kretsene. AC-021604-02 : Skemaer er lagret på hensiktsmessig sted.			

US-021602	Estimert tid: -	Prioritet: -	Sprint: -
Som oppdragsgiver ønsker jeg at dokumentasjonen skal inneholde BillOfMaterials for alle egenutviklede elektriske kretser, slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.			
Acceptance Citeria/Kommentar: Overordnet Krav			

US-02160201	Estimert tid: 8	Prioritet: 3	Sprint: 2 til 5
Som bachelorgruppe ønsker vi at BOM lages for WS-sensoren , slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.			

Acceptance Criteria/Kommentar:

AC-021605-01 : BOM er produsert og lagt i teknologidokument

US-021603	Estimert tid: -	Prioritet: -	Sprint: -
Som ingeniør ønsker jeg at dokumentasjonen skal inneholde data ark for alle 3.parts komponenter, slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.			
Acceptance Criteria/Kommentar:			
Overordnet Krav			

US-02160301	Estimert tid: 10	Prioritet: 2	Sprint: 2 til 5
Som bachelorgruppe ønsker vi at dokumentasjonen skal inneholde data ark for alle 3.parts komponenter brukt på WS-sensor, slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.			
Acceptance Criteria/Kommentar:			
AC-021606-01 : Datark er samlet så langt det lar seg gjøres å lagt i egen mappe			

US-021701	Estimert tid: 16	Prioritet: 5	Sprint: 3
Som oppdragsgiver ønsker jeg at dokumentasjonen skal inneholde UML diagrammer for all egenutviklet software, slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.			
Acceptance Criteria/Kommentar:			
AC-021703-01 : Klassediagrammer er utarbeidet			
AC-021703-02 : Klassediagrammer blir vist frem og forklart i teknologidokumentet			

US-021702	Estimert tid: 48	Prioritet: 4	Sprint: 2
Som ingeniør ønsker jeg at dokumentasjonen skal inneholde USE CASE for alle brukeroparasjoner, slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.			
Acceptance Criteria/Kommentar: AC-021702-01 : Teknologidokumentet inneholder use case for alle brukeroparasjoner. AC-021702-02 : Use caser er godkjent av begge fra software			

US-021703	Estimert tid: -	Prioritet: -	Sprint: -
Som ingeniør ønsker jeg at dokumentasjonen skal inneholde klassediagrammer for operatørprogramvaren, slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.			
Acceptance Criteria/Kommentar: Valgt bort 07.04.16			

US-021704	Estimert tid: -	Prioritet: -	Sprint: -
Som oppdragsgiver ønsker jeg at kildekode er strukturert og lettleselig, slik at eventuelle andre utviklere kan forstå og legge til funksjonalitet senere			
Acceptance Criteria/Kommentar: Overordnet Krav			

US-021705	Estimert tid: 20	Prioritet: 4	Sprint: 3
Som oppdragsgiver ønsker jeg at det lages et software teknologidokument, slik at oppdragsgiver har dette til bruk i videreføring av prosjektet			
Acceptance Criteria/Kommentar: AC-021702-01 : Teknologidokumentet inneholder use case for alle brukeroparasjoner. AC-021702-02 : Use caser er godkjent av begge fra software			

US-021706	Estimert tid: 4	Prioritet: 4	Sprint: 2
Som bachelorgruppe ønsker vi at det lages en mal for software designdokument med disposisjoner, slik at vi kan begynne på teknologidokumentet i sprint 2			
Acceptance Criteria/Kommentar: AC-021706-01 : Teknologidokument er opprettet. AC-021706-02 : Teknologidokument er satt opp med disposisjoner AC-021706-03 : Det er skrevet abstrakt og innledning.			

US-021707	Estimert tid: 84	Prioritet: 5	Sprint: 5
Som oppdragsgiver ønsker jeg at All egenutviklet software er dokumentert grundig, slik at vi kan forstå hva dere har gjort			

Acceptance Criteria/Kommentar:

AC-021707-01 : Software designdokument er oppdatert

AC-021707-02 : Software teknologidokument er utarbeidet.

US-021801	Estimert tid: 1	Prioritet: 5	Sprint: 5
Som oppdragsgiver ønsker jeg at dokumentasjonen skal inkludere en brukermanual, slik at dagens eller eventuelt nye operatører lett skal kunne forstå bruken av produktet.			
Acceptance Criteria/Kommentar:			
AC-021801-01 : US-021803, US-021804, US-021805, US-021806, US-021807 og US-021808 er gjort			

US-021802	Estimert tid: -	Prioritet: -	Sprint: -
Som oppdragsgiver ønsker jeg at brukermanual skal beskrive software, slik at den som leser brukermanualen kan bruke software.			
Acceptance Criteria/Kommentar:			
Slettet, dobbelt føring			

US-021803	Estimert tid: 16	Prioritet: 4	Sprint: 5
Som oppdragsgiver ønsker jeg at brukermanual skal beskrive hvordan man legger inn nye sekvenser, slik at testoperatøren kan gjøre dette, uten at det nødvendigvis må være noen til å lære han opp.			
Acceptance Criteria/Kommentar:			
AC-021803-01 : Prosesen er beskrevet og lagt til i dokumentet på hensiktsmessig sted			

US-021804	Estimert tid: 16	Prioritet: 4	Sprint: 5
Som testoperatør ønsker jeg at brukermanual skal være på engelsk, slik at denne ene brukermanualen kan dekke et bredere spekter av testoperatører.			
Acceptance Criteria/Kommentar:			
AC-021804-01: Brukermanual må være skrevet på engelsk			

US-021805	Estimert tid: 16	Prioritet: 4	Sprint: 5
Som testoperatør ønsker jeg at brukermanualen skal beskrive bruk av software, slik at man kan være mindre avhengig av opplæring for å kunne bruke testsystemet.			
Acceptance Criteria/Kommentar:			
AC-021805-01 : Bruke av Software er lagt til			
AC-021805-02 : Omprogramering er beskrevet			
AC-021805-02 : Oppdatering er beskrevet			

US-021806	Estimert tid: 8	Prioritet: 4	Sprint: 5
-----------	-----------------	--------------	-----------

Som testoperatør ønsker jeg at brukermanualen skal beskrive alle forutsetninger som kreves for å kjøre test, slik at disse er lett tilgjengelig for testoperatør, så man ikke støter på problemer underveis i testen.

Acceptance Citeria/Kommentar:

AC-021806-01: Produsert Monterings manual

US-021807	Estimert tid: 4	Prioritet: 4	Sprint: 5	
Som HMS - ansvarlig ønsker jeg at brukermanual skal inneholde sikkerhetsrutiner, slik at de skal være lett tilgjengelig for testoperatøren, og øke sjansen for at han/hun har satt seg inn i disse.				
Acceptance Criteria/Kommentar: AC-021807-01 : Sikkerhetsrutiner er lagt til i brukermanualen på hensiktsmessige steder.				
US-021808	Estimert tid: 4	Prioritet: 4	Sprint: 5	
Som HMS - ansvarlig ønsker jeg at sikkerhetsrutinene i brukermanualen skal være tydelig merket, slik at de er vanskelig og "overse" for testoperatøren				
Acceptance Criteria/Kommentar: AC-021808-01 : Sikkerhetsrutiner er tydelig merket				

1.3 Sikkerhet

US-030101	Estimert tid: 6	Prioritet: 5	Sprint: 4	
Som oppdragsgiver ønsker jeg at samtlige gruppemedlemmer setter seg inn i konsernets sikkerhetsrutiner, slik at alle gruppemedlemmene har dette klart for seg, og ikke bryter disse.				
Acceptance Criteria/Kommentar: AC-030101-01 : Alle gruppene medlemmer har vært igjennom sikkerhets brief. AC-030101-02 : Alle gruppens medlemmer har fylt ut papirene som kreves for å få tillgang til parken/bygningen.				
US-030201	Estimert tid: -	Prioritet: -	Sprint: -	
Som oppdragsgiver ønsker jeg at alle har kontinuerlig fokus på sikkerhet gjennom prosjektet, slik at konsernets verdier ivaretas gjennom hele prosjektet.				
Acceptance Criteria/Kommentar: Overordnet Krav				
US-030301	Estimert tid: -	Prioritet: -	Sprint: -	
Som oppdragsgiver ønsker jeg at ingen slipper inn på huset uten adgangskort, slik at gruppens medlemmer ikke slipper uvedkommende inn ved et uhell.				
Acceptance Criteria/Kommentar: Overordnet Krav				
US-030401	Estimert tid:--	Prioritet:--	Sprint:--	
Som HMS – ansvarlig ønsker jeg at UR5 skal ha en avgrenset sikkerhetssone på 1 meter, slik at jeg kan være sikker på at personell eller annet utstyr ikke skades, eller hindrer UR5 i sitt arbeid.				
Acceptance Criteria/Kommentar: Valgt bort 07.04.16				

1.4 Funksjon

US-040101	Estimert tid: 28	Prioritet: 5	Sprint: 2
Som oppdragsgiver ønsker jeg at software skal kunne kommunisere med robotarm via TCP, slik at signaler kan sendes og mottas mellom disse to.			
Acceptance Criteria/Kommentar: AC-040101-01 : Software kan sette opp tilkobling med robotarm AC-040101-02 : Det er programert en listener loop på robotarm AC-040101-03 : Det kjører en testsekvens når tilkobling skjer for å vise at tilkobling skjer.			
US-040201	Estimert tid: -	Prioritet: -	Sprint: -
Som oppdragsgiver ønsker jeg at software skal kunne kommunisere med WS, slik at signaler kan sendes og mottas mellom disse to.			
Acceptance Criteria/Kommentar: Slettet, se issues			
US-040301	Estimert tid: 36	Prioritet: 5	Sprint: 4 til 5
Som oppdragsgiver ønsker jeg at software skal kunne kjøre bevegelsessekvenser på UR 5 robotarm, slik at software kan kontrollere arm.			
Acceptance Criteria/Kommentar: AC-040301-01 : Software kan sende innlastet xml fil med instruksjoner til ur 5 robotarm og de blir forstått og fulgt. AC-040301-02 : Det er lagt til funksjonalitet i programmet slik at brukeren kan lage egne testsekvenser utifra de mulige funksjonene på DCP og CG AC-040301-03 : Det er opprettet en funksjon for hver hver bevegelsessekvens som er mulig på UR 5			
US-040401	Estimert tid: 20	Prioritet: 5	Sprint: 4 til 5
Som oppdragsgiver ønsker jeg at UR5 må kunne kommunisere med arduino som styrer hånd, slik at software kan kontrollere hånd.			
Acceptance Criteria/Kommentar: AC-040601-01 : Det er satt opp ett statemachine diagram er utarbeidet og overlevert elektro			
US-040501	Estimert tid: -	Prioritet: -	Sprint: -
Som oppdragsgiver ønsker jeg at software skal kunne styre gripeverktøyet, slik at software kan kontrollere hånd.			
Acceptance Criteria/Kommentar: Slettet, UR5 skal styre EE			

US-040601	Estimert tid: 10	Prioritet: 5	Sprint: 4
Som oppdragsgiver ønsker jeg at software skal kunne ta imot fyringssignal, slik at software kan verifisere fyring.			
Acceptance Criteria/Kommentar: AC-040601-01 : Software tar imot en variabel som bestemmer om fyring skjer eller ikke AC-040601-02 : Software kan vise om fyring skjer i GUI			

US-040701	Estimert tid: 16	Prioritet: 5	Sprint: 4
Som oppdragsgiver ønsker jeg at software skal kunne lese testinstruksjoner fra fil, slik at software kan utføre testsekvenser			
Acceptance Criteria/Kommentar: AC-040701-01 : Software leser inn en XML fil og lagrer den i et string array			

US-042001	Estimert tid: 12	Prioritet: 5	Sprint: 4
Som oppdragsgiver ønsker jeg at det opprettes et loggsystem som skal registrere hendelser, slik at det kan lese ut av loggen hva systemet har gjort.			
Acceptance Criteria/Kommentar: AC-042001-01 : Loggsystem er opprettet og testet			

US-042002	Estimert tid: 4	Prioritet: 5	Sprint: 4
Som oppdragsgiver ønsker jeg at loggsystemet skal notere instruksjon i loggen, slik at det kan leses ut av loggen hvilke instruksjoner software gir til systemet.			
Acceptance Criteria/Kommentar: AC-042002-01 : Det er satt opp en funksjon som skriver til fil			

US-042003	Estimert tid: 8	Prioritet: 5	Sprint: 4 til 5
Som oppdragsgiver ønsker jeg at loggsystemet skal verifisere bevegelse ved UR5 arm, slik at det kan leses ut av loggen at det systemet ble gitt instruksjoner om, faktisk ble gjort			
Acceptance Criteria/Kommentar: AC-042003-01 : Hendelser som gjøres på UR5 noteres i loggen AC-042003-02 : Bekreftelse på at de er gjennomført notres i loggen			

US-042004	Estimert tid: 8	Prioritet: 5	Sprint: 4 til 5
Som oppdragsgiver ønsker jeg at loggsystemet skal verifisere bevegelse ved EE, slik at det kan leses ut av loggen at det systemet ble gitt instruksjoner om, faktisk ble gjort			
Acceptance Criteria/Kommentar: AC-042004-01 : UR5 rapporterer for EE at handlinger er igangsatt.			

US-042005	Estimert tid: -	Prioritet: -	Sprint: -
Som oppdragsgiver ønsker jeg at loggsystemet skal verifisere hendelse ved WS, slik at det kan leses ut av loggen at det systemet ble gitt instruksjoner om, faktisk ble gjort			
Acceptance Criteria/Kommentar: Valgt bort 07.04.16			

US-042006	Estimert tid: -	Prioritet: -	Sprint: -
Som oppdragsgiver ønsker jeg at loggsystemet skal sammenligne forventet resultat med faktisk resultat, slik at systemet automatisk kan kontrollere om testen er godkjent			
Acceptance Criteria/Kommentar: Slettet, se issues			

US-042007	Estimert tid: 4	Prioritet: 2	Sprint: 4
Som oppdragsgiver ønsker jeg at loggsystemet skal markere uventet utfall i loggen, slik at tester med feil er lett avdekke.			
Acceptance Criteria/Kommentar: AC-042007-01 : Når testen vi instruerte roboten til å gjøre er ferdig vil det komme en beskjed tilbake til test programmet som skrives til loggen.			

US-043001	Estimert tid: 10	Prioritet: 5	Sprint: 1, 2 til 4
Som oppdragsgiver ønsker jeg at EE skal kunne festes på UR5, slik at de to kan brukes sammen.			
Acceptance Criteria/Kommentar: AC-043001-01 : Interface er designet AC-043001-02 : Interface er godkjent av PO AC-043001-01 : Interface er printet AC-043001-03 : Interface er fysisk testet koblet på UR5			

US-043002	Estimert tid: 12	Prioritet: 5	Sprint: 2 til 4
Som ingeniør ønsker jeg at det fysiske grensesnitt UR5/EE skal bestå av to deler før kobling mot EE, slik at EE kan monteres på/av på en hensiktsmessig måte			
Acceptance Criteria/Kommentar: AC-043002-01 : Koblingen er printet AC-043002-02 : Koblingen er fysisk testet			

US-043003	Estimert tid: 2	Prioritet: 5	Sprint: 2
Som ingeniør ønsker jeg at at koblingen til UR5 tilrettelegges for festing med 4 M6 skruer rundt origo, slik at den skal passe korrekt og være festet stødig uten å skade gjengene på UR5			

Acceptance Criteria/Kommentar:

AC-043003-01 : Design av interface er tilpasset monteringshull på UR5

US-043004	Estimert tid: 2	Prioritet: 5	Sprint: 2
Som ingeniør ønsker jeg at det skal tas en faglig vurdering på valg av løsningene for US-043001, US-043002, US-043003 og at dette begrunnes i teknologidokumentet for EE, slik at vi sikrer at skrueforbindelser og grensesnitt ikke overbelastes			
Acceptance Criteria/Kommentar:			
AC-043004-01 : Det begrunnes kort i tekst som føres inn i teknologidokumenetet.			

US-043101	Estimert tid: 32	Prioritet: 5	Sprint: 4 til 5
Som oppdragsgiver ønsker jeg at EE skal kunne motta instruksjoner fra software via UR5, slik at den kan gjøre de bevegelserne testsoftware instruerer den om			
Acceptance Criteria/Kommentar:			
AC-043101-01 : Statemaskinen er kodet			
AC-043101-02 : Arduino tar imot instrukser fra UR5 (test)			
AC-043101-03 : Dokumentasjon av AC-043101-01/02.			

US-043201	Estimert tid: -	Prioritet: -	Sprint: -
Som oppdragsgiver ønsker jeg at EE skal kunne sende informasjon til software, slik at EE kan verifisere når en instruksjon er gjennomført, og dette kan lagres i loggen.			
Acceptance Criteria/Kommentar:			
Valgt bort 07.04.16			

US-044001	Estimert tid: -	Prioritet: -	Sprint: -
Som oppdragsgiver ønsker jeg at EE skal kunne betjene CG, slik at tester som inkluderer betjening av CG kan automatiseres.			
Acceptance Criteria/Kommentar:			
Overordnet Krav			

US-044002	Estimert tid: 40	Prioritet: 5	Sprint: 2 til 5
Som oppdragsgiver ønsker jeg at EE skal kunne holde inne palmknapp, slik at tester som inkluderer betjening av palmknapp kan automatiseres.			
Acceptance Criteria/Kommentar:			
AC-044002-01 : Beregninger er gjort slik at det skal er nok styrke i utforming av EE			
AC-044002-02 : Servomotorer er programert til å utføre sekvensene for å aktivere pamknapp.			
AC-044002-03 : Dokumentasjon av AC-044002-01/02.			

US-04400201	Estimert tid: 24	Prioritet: 5	Sprint: 2 til 3
-------------	------------------	--------------	-----------------

Som ingeniør ønsker jeg at EE tilpasset til å holde inne palmknapp modelleres detaljert i Solidworks, slik at det klargjøres for neste fase, der elektronikk skal lage styring.

Acceptance Criteria/Kommentar:

AC-04400201-01 : 3D modell detaljert med klart definert funksjon for betjening av Palm

AC-04400201-02 : Modellen er designet med nødvendige volum for elektronikk og komponenter

US-044004

Estimert tid: 8

Prioritet: 5

Sprint: 2

Som ingeniør ønsker jeg at EE skal ha en anordning i håndflaten til å passe CG , slik at man oppnår stødighet i grepet

Acceptance Criteria/Kommentar:

AC-044004-01 : EE utstyres med "formbar" anordning for best friksjon i grep

AC-044004-02 : EE 's anordning i håndflaten gir stødig grep om CG ved fysisk test

US-044005

Estimert tid: 32

Prioritet: 5

Sprint: 4

Som oppdragsgiver ønsker jeg at EE skal kunne betjene triggerknapp, slik at tester som inkluderer betjening av triggerknapp kan automatiseres.

Acceptance Criteria/Kommentar:

AC-044005-01 : Beregninger er gjort slik at det skal er nok styrke i utforming av EE

AC-044005-02 : Servomotorer er programert til å utføre sekvensene for å aktivere triggerknapp.

AC-044005-03 : Dokumentasjon av AC-044005-01/02.

US-044006	Estimert tid: -	Prioritet: -	Sprint: -
Som oppdragsgiver ønsker jeg at EE skal kunne betjene øvrige knapper på CG, slik at tester som inkluderer betjening av øvrige knapper på CG kan automatiseres.			
Acceptance Criteria/Kommentar: Valgt bort 07.04.16			

US-044101	Estimert tid: -	Prioritet: -	Sprint: -
Som oppdragsgiver ønsker jeg at EE skal kunne betjene DCP, slik at tester som inkluderer betjening av DCP kan automatiseres.			
Acceptance Criteria/Kommentar: Overordnet Krav			

US-044102	Estimert tid: 16	Prioritet: 5	Sprint: 4
Som oppdragsgiver ønsker jeg at EE skal kunne betjene de forskjellige vippebryterne på DCP, slik at tester som inkluderer betjening av vippebrytere på DCP kan automatiseres.			
Acceptance Criteria/Kommentar: AC-044102-01 : Beregninger er gjort slik at det skal er nok styrke i utforming av EE AC-044102-02 : Programmering av EE skal være ferdig slik at Software kan programmere UR5 til å gjøre sekvens. AC-044102-03 : Software programmerer sekvens for bevegelse av UR5 slik at EE utfører test for betjening av vippebrytere til DCP AC-044102-04 : Dokumentasjon av AC044102-01/02/03			

US-044103	Estimert tid: 16	Prioritet: 5	Sprint: 4
Som oppdragsgiver ønsker jeg at EE skal kunne trykke på alle knappene på DCP, slik at tester som inkluderer betjening av alle knapper på DCP kan automatiseres.			
Acceptance Criteria/Kommentar: AC-044103-01 : Beregninger er gjort slik at det skal er nok styrke i utforming av EE AC-044103-02 : Programmering av EE skal være ferdig slik at Software kan programmere UR5 til å gjøre sekvens. AC-044103-03 : Software programmerer sekvens for bevegelse av UR5 slik at EE utfører test for betjening av Knapper til DCP AC-044103-04 : Dokumentasjon av AC044103-01/02/03			

US-044201	Estimert tid: -	Prioritet: -	Sprint: -
Som oppdragsgiver ønsker jeg at EE skal kunne betjene WSCP, slik at tester som inkluderer betjening av WSCP kan automatiseres.			
Acceptance Criteria/Kommentar: Valgt bort 07.04.16			

US-044201	Estimert tid: -	Prioritet: -	Sprint: -	
Som oppdragsgiver ønsker jeg at EE skal betjene break-out box, slik at tester som inkluderer betjening av break-out box kan automatiseres.				
Acceptance Criteria/Kommentar: Valgt bort 07.04.16				

US-045001	Estimert tid: -	Prioritet: -	Sprint: -	
Som oppdragsgiver ønsker jeg at UR5 sin posisjon skal styres av software på testmaskin, slik at testsoftware kan fungere som hjernen i systemet.				
Acceptance Criteria/Kommentar: Slettet, flere som dekker det samme				

US-045101	Estimert tid: 16	Prioritet: 5	Sprint: 4	
Som oppdragsgiver ønsker jeg at UR5 skal kommunisere tilbake til software på testmaskin, slik at testsoftware kan lagre bekreftelse på at instruksjoner er utført i loggen.				
Acceptance Criteria/Kommentar: AC-045101-01 : Det er mulig for software og ta imot informasjon fra UR5 AC-045101-02 : Informasjon fra UR5 lagres i software. AC-045101-03 : Det er tilrettelagt at dette kan sendes videre til logg				

US-045201	Estimert tid: 20	Prioritet: 5	Sprint: 4	
Som oppdragsgiver ønsker jeg at UR5 skal kunne sette opp en socket listener loop, slik at UR5 kan lytte etter tilkobling fra testsoftware.				
Acceptance Criteria/Kommentar: AC-045201-01 : Socket listener loop med bevegelsessekvenser som inkluderer gripeverktøy er opprettet				

US-046001	Estimert tid: -	Prioritet: -	Sprint: -	
Som oppdragsgiver ønsker jeg at det monteres sensor(er) på WS, slik at hendelser på WS kan monitoreres.				
Acceptance Criteria/Kommentar: Overordnet Krav				

US-046101	Estimert tid: 3	Prioritet: 5	Sprint: 2 til 4	
Som oppdragsgiver ønsker jeg at sensor skal registrere alle avfyringer, slik at alle avfyringer som skjer registreres				
Acceptance Criteria/Kommentar: AC-046101-01 : US-046102 er gjort.				

AC-046101-02 : US-046103 er gjort.
 AC-046101-03 : US-046104 er gjort.

US-046102	Estimert tid: 8	Prioritet: 5	Sprint: 2 til 4
Som ingeniør ønsker jeg at tilkoblingsmulighetene på WS for sensor kartlegges, slik at sensor kan designes til å passe overens med denne utgangen.			
Acceptance Citeria/Kommentar:			
AC-046102-01 : Møte gjennomføres/informasjon skaffes			
AC-046102-02 : Informasjonen er lagret på hensiktsmessig sted uten sporbarhet tilbake til WS			

US-046103	Estimert tid: 38	Prioritet: 5	Sprint: 2 til 4
Som ingeniør ønsker jeg at det designes en krets som tolker avfyrings signalet, og oversetter dette til en 5V puls, slik at signalet kan mottas av arduinoen.			
Acceptance Citeria/Kommentar:			
AC-046103-01 : Ideen til krets er notert ned			
AC-046103-02 : Konseptet er simulert i f.eks OrCad for å teste om det er mulig			
AC-046103-03 : Konseptet er testet i praksis			

US-046104	Estimert tid: 4	Prioritet: 5	Sprint: 2
Som ingeniør ønsker jeg at arduino kodes til å sende beskjed til skjerm ved høyt signal på innporten, slik at vi kan verifisere at kretsen fungerer og fyringer kan registreres.			
Acceptance Citeria/Kommentar:			
AC-046104-01 : Kode skrevet slik at ved merket avfyring vil det umiddelbalrt bli sendt beskjed til testmaskin om dette.			

US-046201	Estimert tid: 2	Prioritet: 5	Sprint: 4
Som oppdragsgiver ønsker jeg at sensor skal kommunisere med software på testmaskin, slik at alle hendelser som registreres kan rapporteres tilbake til testsoftware.			
Acceptance Citeria/Kommentar:			
AC-046201-01 : Sensor kommunikasjon er satt opp og denne er testet			

US-046202	Estimert tid: 4	Prioritet: 5	Sprint: 2
Som ingeniør ønsker jeg at det klargjøres for at Arduino skal kommunisere med testmaskin, slik at dette er klart for når software lager loggsystemet.			
Acceptance Citeria/Kommentar:			
AC-046202-01 : Kode er laget på arduino for håndtering av kommunikasjon til maskin.			
AC-046202-02 : Kode er testet så langt det lar seg gjøre før loggsystemet er på plass.			

US-046301	Estimert tid: 2	Prioritet: 4	Sprint: 4
Som oppdragsgiver ønsker jeg at kommunikasjon mellom sensor og software på testmaskin fungerer trådløst, slik at vi slipper kabler mellom stasjonær testmaskin og bevegelig WS.			
Acceptance Criteria/Kommentar: AC-046301-01 : Sensor kommunikasjon er satt opp og denne er testet			

US-046302	Estimert tid: 16	Prioritet: 4	Sprint: 2
Som ingeniør ønsker jeg at det klargjøres for at Arduino skal kunne kommunisere trådløst, slik at dette er klart for når software lager loggsystemet			
Acceptance Criteria/Kommentar: AC-046302-01 : Kode er laget på arduino for håndtering av trådløs kommunikasjon. AC-046302-02 : Kode er testet så langt det lar seg gjøre før loggsystemet er på plass.			

US-046401	Estimert tid: 20	Prioritet: 5	Sprint: 5
Som ingeniør ønsker jeg at det designes en krets og skrives kode som supplerer WS software med en Våpen ID, slik at den faktisk gjør fyringer.			
Acceptance Criteria/Kommentar: AC-046401-01 : Info er fremskaffet for å få det til AC-046401-02 : Ideen til krets er notert ned, og kodet AC-046401-03 : Konseptet er testet i praksis			

US-047001	Estimert tid: 24	Prioritet: 5	Sprint: 5
Som ioppdragsgiver ønsker jeg at WSS sine funksjoner dokumenteres, slik at WSS sine funksjoner kommer klart frem			
Acceptance Criteria/Kommentar: AC-047001-01 : Fyringskrets og kode er dokumentert AC-047001-02 : Våpen ID krets er designet og kode skrevet AC-047001-03 : Konklusjon om WSS er skrevet			

US-047001	Estimert tid:–	Prioritet:–	Sprint:–
Som oppdragsgiver ønsker jeg at WS status skal monitorers av software, slik at software kan merke eventuelle problemer som oppstår.			
Acceptance Criteria/Kommentar: Valgt bort 07.04.16			

US-047002	Estimert tid:–	Prioritet:–	Sprint:–
Som oppdragsgiver ønsker jeg at software skal stanse umiddelbart ved fatal feil detektert av våre sensorer på WS, slik at sjansen for at system skal ta skade/ødelegges er minimal.			
Acceptance Criteria/Kommentar: Valgt bort 07.04.16			

US-047003	Estimert tid:–	Prioritet:–	Sprint:–
Som oppdragsgiver ønsker jeg at software skal gi feilmelding ved automatisk stopp, slik at testoperatør lettere skal kunne finne feilkilde			
Acceptance Criteria/Kommentar: Valgt bort 07.04.16			

US-047004	Estimert tid:–	Prioritet:–	Sprint:–
Som testoperatør ønsker jeg at software skal kunne resettes etter stopp, slik at den kan settes klar til ny test etter en feil.			

Acceptance Criterias/Kommentar:
Valgt bort 07.04.16

US-047101	Estimert tid:--	Prioritet:--	Sprint:--	
Som testoperatør ønsker jeg at systemet skal ha en manuell nødstopp knapp, slik at testoperatør enkelt skal kunne stoppe systemet om han ser noe er feil.				
Acceptance Criterias/Kommentar: Valgt bort 07.04.16				

US-047102	Estimert tid:--	Prioritet:--	Sprint:--	
Som testoperatør ønsker jeg at manuell nødstopp knapp skal være tydelig merket, slik at testoperatør skal slippe å lete etter denne ved behov.				
Acceptance Criterias/Kommentar: Valgt bort 07.04.16				

US-047103	Estimert tid:--	Prioritet:--	Sprint:--	
Som testoperatør ønsker jeg at manuell nødstopp knapp skal være lett tilgjengelig, slik at testoperatør lett skal kunne trykke på den ved behov.				
Acceptance Criterias/Kommentar: Valgt bort 07.04.16				

1.5 Design

US-050101	Estimert tid: -	Prioritet: -	Sprint: -
Som UR5 robot ønsker jeg at EE skal ikke overstige en egenvekt på 5kg, slik at UR5 ikke skal få problemer med å løfte den			
Acceptance Criterias/Kommentar: Overordnet Krav			

US-050102	Estimert tid: -	Prioritet: -	Sprint: -
Som End-Effector ønsker jeg at jeg lages av materiale som er både lett og sterkt, slik at EE totalvekt ikke overstiger 5 kg.			
Acceptance Criterias/Kommentar: Valgt bort 07.04.16, da det blir en prototype i plast			

US-050103	Estimert tid: 16	Prioritet: 5	Sprint: 2
Som End-Effector krever jeg at jeg har komponenter som er sterke nok til å fullføre bevegelser, slik at også tester som krever disse bevegelsene kan gjøres.			
Acceptance Criterias/Kommentar: AC-050103-01 : Komponenter som tilfører kraft, skal resultere i en styrke på minst 25N i angrepspunktet AC-050103-02 : Det skal vises til utregninger som legges i teknologidokument			

US-050104	Estimert tid: -	Prioritet: -	Sprint: -
Som End-Effector krever jeg at jeg tåler gjentatte bevegelser, slik at jeg kan gjenta tester mange ganger, og sikre at jeg har et langt livsløp.			
Acceptance Criterias/Kommentar: Slettet, Da det ikke ble aktuelt å lage i annet materiale.			

US-050105	Estimert tid: -	Prioritet: -	Sprint: -
Som End-Effector krever jeg at jeg har slitesterke overflateegenskaper, slik at jeg tåler å utsettes for mye friksjon, men fremdeles ha lang levetid			
Acceptance Criterias/Kommentar: Slettet, klønete formulert			

US-050106	Estimert tid: -	Prioritet: -	Sprint: -
Som End-Effector krever jeg at jeg har nok styrke i materialene jeg er laget av, slik at jeg ikke ødelegges av noen av handlingene jeg skal gjennomføre.			
Acceptance Criterias/Kommentar: Slettet, Da det ikke ble aktuelt å lage i annet materiale.			

US-050301	Estimert tid: –	Prioritet: –	Sprint: –
Som End-Effector ønsker jeg at jeg tilpasses CG former, slik at det er sikkert at jeg har et godt og stabilt grep.			
Acceptance Citeria/Kommentar: Slettet, da vi har andre stories som dekker dette			

US-050401	Estimert tid: 2	Prioritet: 3	Sprint: 2
Som ingeniør ønsker jeg at EE har en luke for å komme til komponenter, slik at EE er lett å gjøre vedlikehold og tilpasninger på.			
Acceptance Citeria/Kommentar: AC-050401-01 : Luken er designet AC-050401-02 : Luken er prøvd i praksis og gir enkel tilgang til komponenter			

US-051001	Estimert tid: 2	Prioritet: 5	Sprint: 2
Som UR5 robot krever jeg at komponenter i EE ikke bruker mer enn 600mA, slik at jeg er sikker på at jeg kan forsyne nok strøm til EE.			
Acceptance Citeria/Kommentar: AC-051001-01 : Beregninger er gjort ved valg av komponenter så dette er på plass.			

US-051101	Estimert tid: 1	Prioritet: 5	Sprint: 2
Som UR5 robot krever jeg at elektronikken i EE skal kunne kjøre på 24V eller 12V, slik at jeg ikke dreper elektronikken, da dette er det jeg kan levere.			
Acceptance Citeria/Kommentar: AC-051101-01 : Valg av kraftoverføringer er foretatt mhp "ramme" på spenning lik 12 el 24 volt.			

US-051201	Estimert tid: 4	Prioritet: 5	Sprint: 4
Som End-Effector krever jeg at jeg har en egen kontrollør for å kunne styre komponentene jeg er bygget av, slik at jeg vet hva jeg skal gjøre, selv om UR5 har en begrenset mengde output signaler.			
Acceptance Citeria/Kommentar: AC-051201-01 : Anskaffelse av Aruino ATmega 2560 AC-051201-02 : Test av Arduino ATmega 2560			

US-051202	Estimert tid: 8	Prioritet: 5	Sprint: 2 til 3
Som ingeniør ønsker jeg at kravene til kontrollør kartlegges og det gjøres et valg av kontrollør, slik at den kan bestilles så fort som mulig			

Acceptance Criteria/Kommentar:

AC-051202-01 : Kontrollkort skal velges etter behov for applikasjoner til EE og WS.

AC-051202-02 : Kontrollkort skal bestilles.

US-051301	Estimert tid:--	Prioritet:--	Sprint:--	
Som ingeniør ønsker jeg at den første prototypen skal ha en "pekefinger", slik at bevegelsesprogrameringen kan starte i sprint 2				
Acceptance Criteria/Kommentar: Slettet, da den ikke lenger er aktuell.				

US-051302	Estimert tid:--	Prioritet:--	Sprint:--	
Som utvikler ønsker jeg at alle prototypene skal være like lange fra base til fingertupp, slik at vi kan ferdigstille en del programering allerede i sprint 2				
Acceptance Criteria/Kommentar: Slettet, da den ikke lenger er aktuell.				

US-051401	Estimert tid: 32	Prioritet: 5	Sprint: 2	
Som PO ønsker jeg at det kartlegges hvilke komponenter vi planlegger å bruke for å styre EE, slik at disse kan bestilles så fort som mulig, og designet kan lages med hensyn på komponentene som skal inni				
Acceptance Criteria/Kommentar: AC-051401-01 :Valg av komponenter er gjort basert på research AC-051401-02 : Valg av komponent er tatt AC-051401-03 : Motor er bestilt				

US-051501	Estimert tid: 4	Prioritet: 5	Sprint: 3	
Som Elektro ansvarlig ønsker jeg at det kartlegges en løsning for transformering av spenningen ut fra UR5, slik at komponentene og kontrolleren får korrekt spenning				
Acceptance Criteria/Kommentar: AC-051501-01 : Det skal velges et system som kan regulere spenningen fra 24V til 6V for å kunne levere spenning til Servoer/komponenter og Arduino. AC-051501-02 : Det skal vises en krets for teoretisk verifisering. AC-051501-03 : Komponent(er) researched for bestilling i eventuell sprint 4.				

1.6 Grensesnitt

US-060101	Estimert tid: –	Prioritet: –	Sprint: –
Som oppdragsgiver ønsker jeg at testmaskinen skal kunne gi instruksjoner til UR5, slik at vi har en sentralisert styring av systemet.			
Acceptance Citeria/Kommentar: Slettet, flere som dekker det samme			

US-060201	Estimert tid: 16	Prioritet: 5	Sprint: 4
Som oppdragsgiver ønsker jeg at testmaskinen skal kunne motta informasjon fra UR5, slik at informasjon som er viktig for loggen kommer frem.			
Acceptance Citeria/Kommentar: AC-060201-01 : Software kan motta nødvendig data fra UR 5 AC-060201-02 : Nødvendig data kan lagres i programmet for bruk i logg			

US-060301	Estimert tid: 20	Prioritet: 5	Sprint: 4 til 5
Som oppdragsgiver ønsker jeg at UR5 skal kunne gi instruksjoner til EE, slik at vi har en sentralisert styring av systemet.			
Acceptance Citeria/Kommentar: AC-060301-01 : UR 5 sender signaler til EE via arduino AC-060301-02 : EE beveger seg slik den skal som følge av signaler fra UR 5			

US-060401	Estimert tid: –	Prioritet: –	Sprint: –
Som ingeniør ønsker jeg at det tilrettelegges for å lagre informasjon fra EE, slik at informasjon som er viktig for loggen kommer frem.			
Acceptance Citeria/Kommentar: Slettet, se issues			

US-060501	Estimert tid: 8	Prioritet: 5	Sprint: 4 til 5
Som oppdragsgiver ønsker jeg at testmaskinen skal kunne gi instruksjoner til sensorer, slik at vi har en sentralisert styring av systemet.			
Acceptance Citeria/Kommentar: AC-060501-01 : Det er opprettet en funksjon som kan sende informasjon til sensor AC-060501-02 : Funksjonen er testet og det er dokumentert i testverifikasjons dokumentet			

US-060601	Estimert tid: –	Prioritet: –	Sprint: –
Som oppdragsgiver ønsker jeg at testmaskinen skal kunne motta informasjon fra sensorer, slik at informasjon som er viktig for loggen kommer frem.			

Acceptance Citeria/Kommentar:
Slettet, dobbeltføring

US-060701	Estimert tid:--	Prioritet:--	Sprint:--	
Som oppdragsgiver ønsker jeg at testmaskinen skal kunne gi instruksjoner til WS, slik at testsekvenser som krever dette kan gjennomføres.				
Acceptance Citeria/Kommentar: Slettet, se issues				

US-060801	Estimert tid:--	Prioritet:--	Sprint:--	
Som oppdragsgiver ønsker jeg at testmaskinen skal kunne motta informasjon fra WS, slik at testsekvenser som krever dette kan gjennomføres.				
Acceptance Citeria/Kommentar: Slettet, se issues				

1.7 Andre

US-070101	Estimert tid: 28.	Prioritet: 5	Sprint: ALLE	
Som prosjektgruppe ønsker vi god kommunikasjon med oppdragsgiver, slik at vi kan sikre et godt samarbeide.				
Acceptance Criteria/Kommentar: AC-070101-01 : Oppdragsgiver har fått statusrapport til rett tid. AC-070101-02 : Alle Gruppemedlemmer har møtt til avtalt tid til evt møter.				
US-070201	Estimert tid: 16	Prioritet: 5	Sprint: 1	
Som prosjektgruppe ønsker vi møter med oppdragsgiver, slik at vi sikrer et godt grunnlag til User Stories.				
Acceptance Criteria/Kommentar: AC-070201-01 : Møter er gjennomført				
US-070301	Estimert tid: 16	Prioritet: 5	Sprint: 1	
Som Scrum master ønsker jeg at det lages et prosjekthåndteringsverktøy, slik at vi kan se hvor vi er i sprinten til enhver tid				
Acceptance Criteria/Kommentar: AC-070301-01 : Verktøyet er utarbeidet AC-070301-02 : Verktøyets bruk er gjennomgått med gruppen AC-070301-03 : Verktøyet er godkjent av gruppen				
US-070401	Estimert tid: 14.	Prioritet: 4	Sprint: ALLE	
Som Scrum master ønsker jeg at gruppa har et budsjett som oppdateres fortløpende, slik at vi har god økonomisk kontroll				
Acceptance Criteria/Kommentar: AC-070401-01 : Det er gått en ny runde med alle fagretninger der de har hatt mulighet til å komme med input. AC-070401-02 : Informasjonen er bearbeidet og lagt inn i budsjettkdokumentet. AC-070401-03 : Informasjonen er oppdatert på alle relevante steder..				
US-070501	Estimert tid: 12	Prioritet: 4	Sprint: ALLE	
Som Scrum master ønsker jeg at gruppa har et timebudsjett som oppdateres fortløpende, slik at vi har oversikt over hvem som har brukt timer, og på hva				
Acceptance Criteria/Kommentar: AC-070501-01 : Alle har oppdatert timelistene sine.				
US-070601	Estimert tid: 36	Prioritet: 4	Sprint: 1	

Som høyskole ønsker jeg at gruppa lager en nettside, slik at vi kan følge med der, og få et inntrykk av hva oppgaven går ut på

Acceptance Criteria/Kommentar:

AC-070601-01 : Innholdet er godkjent av arbeidsgiver

AC-070601-02 : Nettsiden er publisert

US-070602

Estimert tid: 3

Prioritet: 3

Sprint: 1

Som prosjektgruppe ønsker vi at presentasjonene lages og legges på Websidene, slik at presentasjonene alltid er tilgjengelig på nett

Acceptance Criteria/Kommentar:

AC-070602-01 : Presentasjonen er implementert i nettsiden

US-070603

Estimert tid: 8

Prioritet: 3

Sprint: 2

Som høyskole ønsker jeg at nettsidene inkluderer en blog, slik at vi enklere kan følge med på hva som skjer med prosjektet

Acceptance Criteria/Kommentar:

AC-070603-01 : Blog er implementert på hjemmesiden

AC-070603-02 : En første post er lagt ut

US-070604

Estimert tid: 1

Prioritet: 3

Sprint: 2

Som høyskole ønsker jeg at nettsidene inkluderer en presentasjon av gruppens medlemmer, slik at det er lett å se både for oss, men også for andre, hvem som er en del av gruppa

Acceptance Criteria/Kommentar:

AC-070604-01 : Presentasjon av gruppe-medlemmer er lagt ut

AC-070604-02 : Alle har fått muligheten til å oppdatere hva som skal stå der.

US-070605

Estimert tid: 1

Prioritet: 3

Sprint: 3

Som bachelorgruppe ønsker jeg at presentasjon 2 er tilgjengelig på nettsiden, slik at vi lett kan informere våre interessenter

Acceptance Criteria/Kommentar:

AC-070605-01 : 2 presentasjon er lagt tilgjengelig på nettsiden

US-070606

Estimert tid: -

Prioritet: 3

Sprint: 6

Som bachelorgruppe ønsker jeg at presentasjon 3 er tilgjengelig på nettsiden, slik at vi lett kan informere våre interessenter

Acceptance Criteria/Kommentar:

US-070607

Estimert tid: 3

Prioritet: 3

Sprint: ALLE

Som bachelorgruppe ønsker jeg at blog oppdateres regelmessig, slik at vi lett kan informere våre interessenter

Acceptance Criteria/Kommentar:

AC-070607-01 : Oppdateringer er lagt ut på blogg.

US-070701

Estimert tid: 4

Prioritet: 5

Sprint: 2

Som oppdragsgiver ønsker jeg at at det lages en burndown chart for ferdigstilte User Stories, slik at vi kan tracke progresjonen på ferdigstilling av User Stories

Acceptance Criteria/Kommentar:

AC-070701-01 : Dokumentet er utarbeidet.

AC-070701-02 : Burndown chart er tilpasset og lagt til i Statusrapport

US-070702

Estimert tid: 22

Prioritet: 5

Sprint: ALLE

Som intern sensor krever jeg at project burndown skal fremstilles på en måte, så oppdragsgiver kan få muligheten til å se hvilke User Stories vi ikke rekker. og være med på å prioritere userstories, slik at vi har en felles oppfatning av målet, og at de kan hjelpe oss å prioritere de som er viktigst.

Acceptance Criteria/Kommentar:

AC-070702-01 : Godkjendt av gruppen

AC-070702-02 : Godkjendt av oppgragsgiver

US-070801

Estimert tid: 33

Prioritet: 5

Sprint: ALLE

Som Product Owner ønsker jeg at User Stories dokumentet holds ved like, slik at dokumentet alltid vil være oppdatert

Acceptance Criteria/Kommentar:

AC-070801-01 : Dokumentet er oppdater med de siste endringene før Sprints slutt

US-070901

Estimert tid: 71

Prioritet: 5

Sprint: ALLE

Som oppdragsgiver ønsker jeg at at riskdokumentet oppdateres fortløpende og brukes underveis i prosjektet, slik at RISK hele tiden er i fokus for prosjektgruppen

Acceptance Criteria/Kommentar:

AC-070901-01 : Risk ansvarlig har gått runden og samlet inn ny info om risk fra alle fagansvarlige

AC-070901-02 : Riskdokument er oppdatert

AC-070901-03 : Riskdomumentet er gjennomgått og godkjent i plenum

US-071001

Estimert tid:—

Prioritet:—

Sprint:—

Som oppdragsgiver ønsker jeg at alle standarder skal være NS eller ISO, slik at alt følger konsernets standard

Acceptance Criteria/Kommentar:
Valgt bort 07.04.16

1.8 Verifisering

US-080101	Estimert tid:--	Prioritet:--	Sprint:--	
Som End-Effector krever jeg at at alle mine funksjoner er testet med simuleringer., slik at vi er teoretisk sikker på at jeg holder mål.				
Acceptance Criteria/Kommentar: Valgt bort 07.04.16				

US-080102	Estimert tid:--	Prioritet:--	Sprint:--	
Som ingeniør ønsker jeg at vi simulerer CG og EE-Palmtrigger sammen, slik at vi kan verifisere at de vil fungere i teorien, før de produseres.				
Acceptance Criteria/Kommentar: Valgt bort 07.04.16				

US-081001	Estimert tid:--	Prioritet:--	Sprint:--	
Som ingeniør ønsker jeg at det tas fram en 3D modell av CG, slik at de kan simuleres sammen				
Acceptance Criteria/Kommentar: Valgt bort 07.04.16				

Prosjektplan Vedlegg B

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAUGSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Prosjektplan Appendix B
VERSJON / DATO	2.0 / 15. Mars 2016
FORFATTER	Aina Engen Nilsen
SIDEANTALL	25



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



i. Abstrakt

Dette dokumentet er en oppdatert versjon av Prosjektplan Appendix B. Etterhvert som prosjektet har kommet over i en mer teknisk fase, ser vi viktigheten av å kunne bruke risikoanalysen aktivt. For å best få til dette, har vi valgt å tynne ut analysen. Vi vil fokusere mer på mitigerende enn reaktive tiltak, og implementere et håndteringsverktøy for de risikoscenarier som vil kunne ha størst innvirkning på vårt prosjekt.

ii. Innholdsfortegnelse

i. Abstrakt.....	2
ii. Innholdsfortegnelse	3
iii. Oversikt tabeller.....	4
iv. Dokumenthistorie.....	5
1 Innledning	6
2 Kategorisering suksessfaktorer	6
3 Overordnet risikoanalyse	7
3.1.0 Kategorisering	8
3.2.0 Kategorisering	12
4 Spesifikk risikoanalyse	16
4.1.0 Kategorisering	17
4.2.0 Kategorisering	21
4.3.0 Kategorisering	23
5. Konklusjon	25

iii. Oversikt tabeller

Tabell 1: Dokumenthistorie.....	5
Tabell 2: risikooversikt.....	6
Tabell 3: Sannsynlighet og sårbarhet.....	6
Tabell 4: personorienterte suksessfaktorer.....	7
Tabell 5: PER - 01.....	8
Tabell 6: PER - 02.....	8
Tabell 7: PER - 04.....	9
Tabell 8: PER - 05.....	10
Tabell 9: PER - 06.....	10
Tabell 10: scenario og konsekvens	11
Tabell 11: PRR - 03	12
Tabell 12: PRR - 04	12
Tabell 13: PRR - 07	13
Tabell 14: PRR - 09	13
Tabell 15: PRR - 10	15
Tabell 16: PRR - 11	15
Tabell 17: Oversikt risiko elektro	16
Tabell 18: ELR - 01	17
Tabell 19: ELR - 02	17
Tabell 20: ELR - 03	18
Tabell 21: ELR - 04	18
Tabell 22: ELR - 05	19
Tabell 23: ELR - 07	19
Tabell 24: ELR - 08	20
Tabell 25: Oversikt risiko data	20
Tabell 26: DAR - 01	21
Tabell 27: DAR - 02	21
Tabell 28: DAR - 05	22
Tabell 29: Oversikt risiko maskin.....	22
Tabell 30: MAR - 12	23
Tabell 31: MAR - 15	23
Tabell 32: MAR - 18	24

iv. Dokumenthistorie

Tabell 8: Dokumenthistorie

Versjon 0.1	02.02.16	Opprettet av Aina E. Nilsen
Versjon 0.2	08.02.16	Endringer av Aina E. Nilsen: satt inn utkast risk for maskin, satt inn topp og bunntekst iht dokumentstil. Disposisjon. Endret dokumentnavn. Satt inn fire risiko maskin,
Versjon 1.0	08.02.16	Aina: FERDIGSTILT: opprettet versjon 1.0
Versjon 1.1	26.02.16	Thomas/Aina: gjennomgang og oppdatering risiko.
Versjon 1.2	09.03.16	Oppdatering og tilføyning teknisk risiko av Aina
Versjon 1.3	10.03.16	Tillegg og oppdatering risiko av Aina
Versjon 2.0	15.03.16	Ferdigstilling oppdatert risikodokument av Aina

1 Innledning

Risiko beskriver potensialet for å miste noe av verdi som kan påvirke prosjektets mål, vi ser dermed viktigheten av en gjennomgående risikoanalyse.

Dette dokumentet gir oversikt over risiko i prosjektet, og kategorisering av disse. Risikoanalysen vil bygges på etterhvert som prosjektet skrider frem.

2 Kategorisering suksessfaktorer

Sannsynlighet for at en hendelse skal inntreffe rangeres fra 1 - 12, hvor 12 er nesten sikkert. Sårbarhet sier noe om hvor stor konsekvens av oppstått hendelse vi får, og rangeres også fra 1 - 12 hvor 12 er stor sårbarhet (tabell 1.1).

Tabell 9: risikooversikt

Sannsynlighet (1-12)		Sårbarhet (1-12)	
1-3	Sjelden	1-3	Ubetydelig
4-6	Usannsynlig	4-6	Mindre
7-9	Moderat	7-9	Moderat
10-12	Sannsynlig	10-12	Stor

For å best illustrere risiko, samles data i en egen risikomatrise (tabell 1.2). Produktet av sannsynlighet og sårbarhet gir oss en tallfestet risiko for aktuell hendelse, og deretter en fargekode for enkel oversikt.

Tabell 10: Sannsynlighet og sårbarhet

Sannsynlighet	Sårbarhet			
	Ubetydelig	Mindre	Moderat	Stor
Sannsynlig: Vil antagelig	10-36	40-72	70-108	100-144

oppstå de fleste ganger				
Moderat: Kan oppstå de fleste ganger	7-27	28-45	49-81	70-108
Usannsynlig: Kan oppstå en eller annen gang	4-18	16-36	28-45	40-72
Sjelden: Kan oppstå under sjeldne forhold	1-9	4-18	7-27	10-36

Grønn sone: lav risiko - håndteres ved rutineprosedyrer.

Gul sone: Moderat risiko - Ansvarsområder ved håndtering må spesifiseres.

Rød sone: Høy risiko - Bør vurderes i samråd med seniorrådgiver (erfaring og kompetanse).

Mørk rød sone: Ekstra høy risiko - Øyeblikkelig handling/tiltak kreves.

3 Overordnet risikoanalyse

Tabell 11: personorienterte suksessfaktorer

Risiko ID	Scenario	Konsekvens
PER - 01	Sykdom/ulykke	Gruppemedlem må trekke seg underveis.
PER - 02	Dårlig Kommunikasjon	Misforståelser internt og eksternt. Vanskeligheter med å forklare / ordlegge seg.
PER - 04	Interne konflikter	Uenighet i gruppen skaper splid.
PER - 05	Manglende prosjektstyring	Nøkkelperson(er) for prosjektstyring uteblir
PER - 06	Overbelastning/ arbeidsmengde	Skjevfordeling arbeidsmengde og individuell grenser for stress

3.1.0 Kategorisering

Tabell 12: PER - 01

PER - 01		Sykdom/ulykke		Gruppemedlem må trekke seg underveis	
Sannsynlighet		7	42		
Sårbarhet		6			
Konsekvens		Gruppen må fordele arbeid seg i mellom, og kompensere for eventuelle faglige mangler.			
Tiltak		a) Redusere oppgavens omfang tilsvarende ressurstap b) Være meget bevisst arbeidsfordeling, og ta medmenneskelige hensyn c) Sørge for en god og åpen dialog rundt temaet hele veien d) Sørge for å ha tilgang til hverandres arbeid i form av felles plattform			

Tabell 13: PER - 02

PER - 02	Dårlig kommunikasjon		Misforståelser internt og eksternt. Vanskeligheter med å forklare/ordlegge seg.
Sannsynlighet	6	60	
Sårbarhet	10		
Konsekvens	Fører til alt fra frustrasjon til feiltolkninger, og kan lede til store feil og mangler.		
Tiltak	a) Bli enige på forhånd hvordan kommunikasjon skal foregå, og		

	følge dette. b) Skriftlige avtaler og godkjenninger av dokumenter c) Økt fokus på kommunikasjon i form av medarbeidersamtaler d) Samtaleteknikk (bevisst forhold til avsender - mottaker, tydelighet) e) Gruppedynamikk og lav terskel for diskusjoner
--	--

Tabell 14: PER - 04

PER - 04		Interne konflikter		Uenighet som fører til splid i gruppen	
Sannsynlighet		3	24		
Sårbarhet		8			
Konsekvens		Forsinkelser i prosjekt, frafall av medlemmer			
Tiltak		a) Ha føringer på hvordan vi tar opp et problem. b) Ved store uenigheter som ikke lar seg løse internt, bør det hentes inn. en utenforstående (eks.veileder) for bistand, eventuelt beslutning. c) Gruppen skal ha en intern arbeidskontrakt. d) Fokus på kommunikasjon i form av medarbeidersamtaler e) Gruppedynamikk og lav terskel for diskusjoner			

Tabell 15: PER - 05

PER - 05		Manglende prosjektstyring		Nøkkelperson(er) for prosjektstyring uteblir
Sannsynlighet		6	42	
Sårbarhet		7		
Konsekvens		Forsinkelser i prosjekt, ikke fullstendig produkt/feil produkt		
Tiltak		a) Velge en prosjektmodell som gir god oversikt for alle, samt har fokus på kollektiv fremdrift. b) Ha to personer med hovedansvar for fremdrift og oversikt.		

Tabell 16: PER - 06

PER - 06		Overbelastning/ arbeidsmengde		Skjevfordeling arbeidsmengde og individuell grenser for stress	
Sannsynlighet		7	49		
Sårbarhet		7			
Konsekvens		Unødvendig frustrasjon som kan føre til fravær eller frafall i gruppen. Ressurstap og forsinkelser			
Tiltak		a) Være bevisst egne begrensninger og tidlige tegn på overarbeidelse b) Være bevisst på og interessert i medstudenters ve og vel c) Forsøke fordele arbeid så jevnt det lar seg gjøre, og ta hensyn der det er nødvendig d) Være flinke til å si i fra så tidlig som mulig			

Tabell 17: scenario og konsekvens

Risiko ID	Scenario	Konsekvens
PRR - 03	Tekniske problemer	Tap av filer, programvare, tekniske feil sub - disipliner, etc.
PRR - 04	Forsinkelser i arbeid og ferdigstilling	Klarer ikke overholde tidsplanen
PRR - 07	Sene endringer	Kravendringer fra oppdragsgiver sent i prosjektforløpet
PRR - 09	Avhengigheter mellom disipliner	Vi klarer ikke avdekke vesentlige avhengigheter mellom disiplinene tidlig nok, og blir avventende i forhold til hverandre
PRR - 10	Ikke tilstrekkelig teknisk kommunikasjon mellom disipliner	For lite tydelig kommunikasjon og felles planlegging mellom disipliner som har direkteavhengighet i form av grensesnitt
PRR - 11	Mangel på tid fører til lite gjennomtenkte løsninger	fare for at ambisjonsnivået kombinert med tidsbegrensning leder til forhastede og lite gjennomtenkte slutninger

3.2.0 Kategorisering

Tabell 18: PRR - 03

PRR - 03		Tekniske problemer		Tap av filer, programvare, tekniske feil i sub - disipliner, etc.	
Sannsynlighet		6	76		
Sårbarhet		12			
Konsekvens		Merarbeid, forskyvning i tidsplanen			
Tiltak		a) Ha gode rutiner for backup, lagring og historikk			

Tabell 19: PRR - 04

PRR - 04		Forsinkelser i arbeid og ferdigstilling		Klarer ikke overholde tidsplanen	
Sannsynlighet		8	64		
Sårbarhet		8			
Konsekvens		Forsinkelser i prosjektet som kan føre til ufullstendig resultat			
Tiltak		a) Ha en realistisk tidsplan b) Hele tiden revurdere tidsbruk og tidsplan c) Bryte ned oppgaver i mindre deler d) Revurdere vinklingen på problemstillinger og hvordan vi løser			

Tabell 20: PRR - 07

PRR - 07	Sene endringer		Kravendringer fra oppdragsgiver sent i prosjektforløpet
Sannsynlighet	10	20	
Sårbarhet	2		
Konsekvens	Merarbeid, merutgifter og tidsutfordringer. Kan gi ufullstendig produkt		
Tiltak	a) Ha skriftlige føringer for dette tidlig i prosess b) Ha plan for smidige endringer/ endringshåndtering c) Bruke en tilrettelagt /egnet prosjektmodell		

Tabell 21: PRR - 09

PRR - 09	Avhengighet mellom disipliner		Vi klarer ikke avdekke vesentlige avhengigheter mellom disiplinene tidlig nok, og blir avventende i forhold til hverandre
Sannsynlighet	10	80	
Sårbarhet	8		
Konsekvens	Frustrasjon, forsinkelse i prosjekt, merkostnad		
Tiltak	a) Avklare på forhånd flest mulig overordnede (forventede) avhengigheter b) Være tydelige i vår kommunikasjon om hva som må foreligge i hvilken rekkefølge under egen disiplin c) Vise interesse for og tilegne seg nødvendig kunnskap om de		

	andre disipliners områder i prosjektet
--	--

Tabell 22: PRR - 10

PRR - 10	Manglene forklarende teknisk kommunikasjon mellom disipliner	For lite tydelig kommunikasjon og felles planlegging mellom disipliner rundt tekniske avhengigheter (interfaces) av hverandre	
Sannsynlighet	8	48	
Sårbarhet	6		
Konsekvens	Feil og mangler som følge av vesentlige misforståelser, forsinkelser og frustrasjon		
Tiltak	a) Ha føringer for god og tydelig kommunikasjon b) Legge til rette for planlegging sammen c) Være bevisst sitt tekniske språk og forklaringer opp mot medstudenter d) Skriftlige avtaler på løsninger		

Tabell 23: PRR - 11

PRR - 11	Mangel på tid fører til lite gjennomtenkte løsninger	fare for at ambisjonsnivået kombinert med tidsbegrensning leder til forhastede og lite gjennomtenkte beslutninger	
Sannsynlighet	7	49	
Sårbarhet	7		
Konsekvens	Vesentlige feil og mangler, ufullstendig produkt. Forsinkelser og merarbeid/kostnader		
Tiltak	a) Forsøke å ha en realistisk forestilling om innhold i oppgaven, samt tidsbruk b) Ha føringer for hvordan vi best velger løsninger c) Holde korte møter ved alle veivalg, tydelig kommunikasjonen		

	mellom fagretningene for å forsikre oss om at vi sitter med samme oppfatning.
--	---

4 Spesifikk risikoanalyse

Tabell 24: Oversikt risiko elektro

Risiko ID	Scenario Elektro
ELR - 01	Feil på bestilt komponent
ELR - 02	Bestilt komponent ødelagt eller forsvinner i frakt
ELR - 03	Overspenning
ELR - 04	For lite strøm fra UR5
ELR - 05	Elektromagnetisk forstyrrelse
ELR - 07	Feilkobling
ELR - 08	Komponenter som ryker gjennom test

4.1.0 Kategorisering

Tabell 25: ELR - 01

ELR - 01		Feil på bestilt komponent		
Sannsynlighet		4	40	
Sårbarhet		10		
Konsekvens		Merarbeid, merkostnad, forsinkelser		
Tiltak		a) Kontrollere komponent ved mottakelse b) Være bevisst på valgt av leverandør		

Tabell 26: ELR - 02

ELR - 02		Bestilling forsvinner eller ødelegges under frakt		
Sannsynlighet		4	40	
Sårbarhet		10		
Konsekvens		Tidsplan forskyves, merkostnader		
Tiltak		a) Sporing av pakke b) Kontrollere pakke ved mottakelse c) Være bevisst på valg av leverandør		

Tabell 27: ELR - 03

ELR - 03		Overspenning		
Sannsynlighet		3	30	
Sårbarhet		10		
Konsekvens		Ødeleggelser, tidsplan forskyves, merkostnader		
Tiltak		a) Kontroll/"overspenningsvern" b) Forsikre at riktig mengde spenning/strøm føres til komponenter		

Tabell 28: ELR - 04

ELR - 04		for lite strøm fra UR5		
Sannsynlighet		4	32	
Sårbarhet		8		
Konsekvens		Systemet virker ikke		
Tiltak		a) Ekstra strømkilde/en booster b) Bruk av servomotorer kontra DC		

Tabell 29: ELR - 05

ELR - 05		Elektromagnetisk forstyrrelse		
Sannsynlighet		8	64	
Sårbarhet		8		
Konsekvens		Signalforstyrrelser, systemet virker ikke optimalt		
Tiltak		a) Isolering/filtrering av støy		

Tabell 30: ELR - 07

ELR - 07	Feilkobling		
Sannsynlighet	3	30	
Sårbarhet	10		
Konsekvens			
Tiltak	a) Kontrollering før testing		

Tabell 31: ELR - 08

ELR - 08		Komponenter som ryker i test		
Sannsynlighet		5	30	
Sårbarhet		6		
Konsekvens		merkostnad, tidsforskyvninger		
Tiltak		a) bestille backup på forhånd b) Sørge for sikkert testmiljø/område c) Kontroll på inngangsverdier		

Tabell 32: Oversikt risiko data

Risk ID	Scenario
DAR - 01	Tidkrevende feil i kode.
DAR - 02	Menden kompetanse som må tilegnes ila. prosjektet er for stor.
DAR - 05	Ikke klarer tidsfrist.

4.2.0 Kategorisering

Tabell 33: DAR - 01

DAR - 01	Tidkrevende feil i kode		
Sannsynlighet	10	80	
Sårbarhet	8		
Konsekvens	Forsinkelser, merarbeid		
Tiltak	a) Planlegge for feilretting, debugging		

Tabell 34: DAR - 02

DAR - 02	Mengden kompetanse som må tilegnes ila. prosjektet er for stor.		
Sannsynlighet	6	60	
Sårbarhet	10		
Konsekvens	Forskyvning tidsplan, dårlig motivasjon og dertil fremdrift. Redusert kvalitet		
Tiltak	a) Skaffe en oversikt over nødvendig kompetanse og forberede kunnskapene.		

Tabell 35: DAR - 05

DAR - 05	Klarer ikke tidsfrist.		
Sannsynlighet	10	50	
Sårbarhet	5		
Konsekvens	Forsinkelse i prosjektet, merarbeid og merkostnad		
Tiltak	a) Sette frister for ferdigstilling med margin til kritisk deadline. b) Prioriterer realistisk slik at de viktigste funksjonene blir fullført.		

Tabell 36: Oversikt risiko maskin

Risk ID	Scenario
MAR - 12	For stor belastning
MAR - 15	Problemer med gjennomføring av simulering
MAR - 18	Avhengigheter av andres arbeid for egen fremdrift

4.3.0 Kategorisering

Tabell 37: MAR - 12

MAR - 12		For stor belastning på prototype og utstyr		
Sannsynlighet		7	42	
Sårbarhet		6		
Konsekvens		Brudd / tretthetsbrudd, merkostnad, ikke presis.		
Tiltak		a) Endre vinkler på utstyr. b) Kontrollere innfestninger. c) Kontrollere inngangsverdier d) Testing e) Dimensjonering		

Tabell 38: MAR - 15

MAR - 15	Problemer med gjennomføring av simulering		Vanskeligheter med gjennomføring av simulering slik at vi ikke får verifikasjonen vi trenger
Sannsynlighet	8	80	
Sårbarhet	10		
Konsekvens	Usikkerhet rundt mål og funksjonalitet, kan føre til unødvendig tids - og resursbruk		
Tiltak	a) Forenkle modell så godt det lar seg gjøre før simulering b) Be om veiledning c) Begrense kompleksitet og omfang av simulering		

Tabell 39: MAR - 18

MAR - 18	Avhengighet av andres arbeid for egen fremdrift		At filer ikke blir delt tidsnok, arbeid ikke utføres som avtalt eller etter planen.
Sannsynlighet	10	60	
Sårbarhet	6		
Konsekvens	Forsinkelser i fremdrift og frustrasjon. Merarbeid og skjev arbeidsfordeling		
Tiltak	a) Være tydelige i kommunikasjon om hva som er nødvendig for videre fremdrift b) Sette tidsfrist på spesifikke oppgaver og holde disse c) Forsøke å fordele arbeidsmengde best mulig, både med tanke på kapasitet og kompetanse/ forutsetning for tilegning av kompetanse		

5. Konklusjon

Det er ikke nødvendigvis et mål å unngå all risiko. Målet er å planlegge tiltak for å *redusere* risiko, samt håndtere eventuelle scenarioer som kan oppstå på best mulig måte. Mitigerende tiltak vil derfor ha fokus.

Vi skal ligge på et risikonivå som er akseptabelt for gjennomføring av prosjektet, som foregående analyse definerer.

Vi vil fortløpende evaluere og oppdatere risikoanalysen, da vår prosjektmodell er tuftet på hyppig iterasjon.

Vi konkluderer med at vår risikoanalyse omfatter tilfredsstillende mengde scenarier, og mener at vi kan forsvare et akseptabelt nivå for gjennomføring av prosjekt.

Sprint retrospektiv 1

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAGSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Sprint retrospektiv 1
VERSJON / DATO	2.0 / 22. Mai 2016
FORFATTER	Thomas Wegener
SIDEANTALL	16



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



i. Abstrakt

Dette dokumentet beskriver hvordan vi har gjennomført sprint 1. Det inneholder en analyse av samarbeid og kommunikasjon, og en fullstendig gjennomgang av oppgaver knyttet til sprint 1.

ii. Innholdsfortegnelse

i. Abstrakt.....	2
ii. Innholdsfortegnelse	2
iii. Oversikt over figurer og tabeller	2
iv. Dokumenthistorie.....	3
1. Innledning	4
2. Samarbeid	4
3. Kommunikasjon	4
4. Administrasjon	4
5. Dokument - strategi.....	5
6. User Story gjennomgang.....	6
7. Konklusjon	15

iii. Oversikt over figurer og tabeller

Figur 1: Burndown chart.....	4
Tabell 1 Dokumenthistorie	3

Vi har bevisst valgt å ikke liste alle tabellene i User Story gjennomgang kapittelet, der hver User Story vi har jobbet med i denne sprinten er representert i en egen tabell.

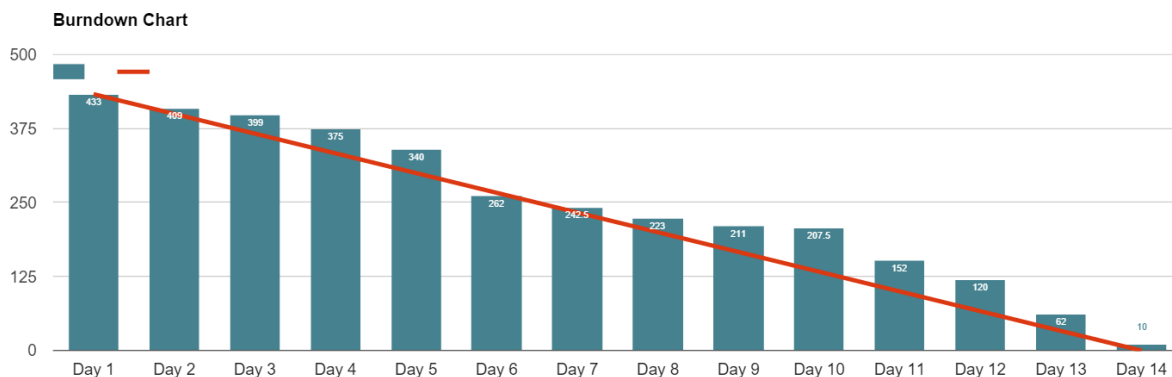
iv. Dokumenthistorie

Tabell 1 Dokumenthistorie

Versjon 0.1	20.02.16	Mal av Thomas.
Referat	16.02.16	Referat fra møtet av Aina
Versjon 0.2	24.02.16	Tekst av Thomas, publisering for korrektur.
Versjon 1.0	26.02.16	Korrektur av Thomas og Monica
Versjon 1.1	15.03.16	Thomas: Revisjon og endring av dokumentnavn.
Versjon 1.2	18.05.16	Thomas: Lagt til Abstrakt, innledning og konklusjon
Versjon 2.0	22.05.16	Thomas: Ferdigstilt

1. Innledning

Vi har gjennomført en produktiv første sprint. Dette dokumentet beskriver en evaluering av arbeidet og er en gjennomgang av user stories.



Figur 1: Burndown chart

2. Samarbeid

Vi er enige om at gruppen fungerer bra, og er fornøyd så langt. Vi tar opp ting med en gang, og er flinke til å luften meninger. Vi er også enige om å fortsette i samme spor videre, og konstruktiv kritikk/tilbakemeldinger oss imellom er viktig.

3. Kommunikasjon

Vi har forstått at nøkkelen til suksess er nettopp kommunikasjon. Vi kommer til å innføre kommunikasjon/teambuilding som et fast punkt på agenda, og Scrum Master vil fungere som 'kommunikasjons - noden' i gruppa. Vi skal også være bevisst på samtaleteknikk og mottaker for det vi vil kommunisere. Har blitt enige om at vi maks kan stå på to oppgaver i sprint - log samtidig, i tillegg ha bedre kommentarer og hyppigere oppdatering her. Dette for å minimere risiko for kommunikasjons feil.

4. Administrasjon

Thomas etterlyser tilbakemeldinger på dette, gruppen har ingenting å utsette. Vi er enige om at alle beslutninger tas i plenum. Vi er også oppmerksomme på at fra nå er ikke endringer i US/AC like fleksible siden vi har definert oppgave. Disse må nå administreres på en litt annen måte, rådfør med Bård Simen før eventuelle endringer foretas.

5. Dokument - strategi

Vi fikk innspill fra intern veileder vedr. våre dokumenter (ref. internt møtereferat 16, 11.02.16), og vil endre dokumentene fortløpende. Viktig med korrekt referanseføring, vi ser nærmere på dette. Monica og Aina ser på revisjonene.

Når det gjelder de store dokumentsammenstillingene, lærte vi i sprint 1 hvordan vi ikke ønsker å gjøre det videre. Ny strategi: en person har oversikt/fugleperspektiv og styrer oppsett/disposisjon, samt sørger for god sammenheng. Selve innholdet genereres sammen. Dokumentet går gjentatte runder ut fra "fuglen" til gruppa med føringer for innhold eller ønske om innspill, korrektur eller liknende. Ny strategi tenker vi vil være mest tids- og ressurseffektiv, og ikke minst i tråd med selve arbeidsmodellen vår.

6. User Story gjennomgang

Etter gjennomført sprint sitter vi nå igjen med 3 kategorier av de User Storiene vi har hatt med.

- De som er utført har grå bakgrunn.
- Der det er behov for gjentakelse senere er merket med grønn bakgrunn.
- De som ikke ble gjennomført er merket med rød.

Vi oppretter issues der vi oppdager at vi må avvike fra den opprinnelige planen. Det kan være med hensyn på tidsbruk, problemer som oppstår, eller at noe ikke blir gjennomført. Issues blir ført inn i et regneark med korresponderende nummer. Vi har listet dem sammen med tilhørende user story i tabeller med gul farge.

[US-010601]	5	1:00
Som oppdragsgiver ønsker jeg at informasjon, kommunikasjon og arbeid relatert til oppgaven holdes konfidensielt slik at potensiell sensitiv informasjon ikke skal komme på avveie		
AC-010601-01: Innhentet retningslinjer for publisering på web		
AC-010601-02: At vi får skriftlig godkjenning i forkant av presentasjon		
AC-010601-03: Alle overholder taushetsplikten		
AC-010601-04: At vi får skriftlig godkjenning på rutiner for dokumentlagring		

ISSUE
Spørsmål til product owner. Holder det med muntlig godkjenning av sikkerhetsrutinene eller skal vi printe et skriv for signering på tirsdag? – OK
Avtalt med oppdragsgiver at vi ikke trenger skriftlig. Så lenge vi følger avtalt retningslinjer

Etter møte hos KPS er acceptance criteria modifisert, oppgaven er utført så lenge retningslinjene blir fulgt. Gradert informasjon skal kun behandles på KPS maskiner eller i papirform innenfor Kongsberggruppens område. Det gruppen produserer av tegninger og tekniske løsninger er vi forsiktig med hvordan vi deler. Vi får lov å låne bilder publisert på nett til presentasjoner og websider. (Ikke bilde av nattmodus). Vi kan presentere prosjektet og dets formål. Vi får lov å sette Kongsberg logoen på nettsiden presentasjonen og i dokumentene.

[US-010701]	5	1:00
Som oppdragsgiver ønsker jeg at det skal vises prototyper slik at vi kan se hva gruppen har tenkt, og for tidlig og fortløpende V&V		
AC-010701-01: Prototype godkjent av product owner. AC-010701-02: Prototype er ferdig produsert. AC-010701-03: Alle mål og dimensjoner er notert for videre bruk.		

Prototypen ble sendt til 3d printing. Dette er en oppgave vi tar tilbake i prosjektbackloggen da vi skal ferdigstille flere prototyper.

[US-051301]	4	16:00
Som ingeniør ønsker jeg at den første prototypen skal ha en pekefinger. Slik at bevegelses-programmeringen kan starte i sprint 2		
AC-051301-01: Første Prototype er utformet med en "pekefinger".		

[US-051302]	3	1:00
Som utvikler ønsker jeg at alle prototypene skal være like lange fra base til fingertupp slik at vi kan ferdigstille en del programmering allerede i sprint 2		
AC-051302-01: Et estimat er gjort for alle nødvendige komponenter for fremtidig funksjonalitet AC-051302-02: Prototypen er formet med plass estimatet i tankene.		

Denne storyen valgte vi å utsette samtidig som vi besluttet å utsette kravet om integrasjon med ur5. Det ville lette arbeidet og samtidig vil vi unngå unødvendige begrensninger i designfasen. Vi vil komme tilbake til det når vi har et mer fungerende design

[US-020101]	5	1:00
Som oppdragsgiver ønsker jeg at systemet skal gjennomgående dokumenteres slik at vi kan se hva som har blitt gjort i prosjektet, og at vi kan fortsette arbeide der gruppen avslutter.		
AC-020101-01: Dokumentasjon for planlagte user stories i sprint1 er ferdigstilt, lagret og arkivert. Og CD og perm er lever.		

[US-020102]	5	6:00
Som intern veileder ønsker jeg at prosjektgruppa sender ukentlig skriftlig rapport slik at jeg kan holde oversikt over fremgang.		
AC-020102-01: Skriftlig rapport er produsert. AC-020102-02: Gruppa har godkjent rapporten. AC-020102-03: Rapporten er sendt til intern veileder.		

Vi sender hver uke en rapport til vår internveileder der vi forteller hvem som har jobbet med hva og hva planen er videre. Dette vil være en gjentakende oppgave igjennom hele prosjektet.

[US-020103]	5	15:00
Som oppdragsgiver ønsker jeg at prosjektgruppa sender skriftlig status rapport hver 14.dag slik at vi blir holdt oppdatert.		
AC-020103-01: Skriftlig rapport er produsert.		
AC-020103-02: Gruppa har godkjent rapporten.		
AC-020103-03: Rapporten er sendt til ekstern veileder.		

Denne tar vi med i hver sprint.

[US-020201]	5	1:00
Som oppdragsgiver ønsker jeg at det lages et prosjektplan dokument slik at vi får oversikt over hva som er planlagt.		
AC-020201-01: Prosjektplan er ferdig produsert.		
AC-020201-02: Gruppa har godkjent prosjektplanen.		
AC-020201-03: Bedriften har godkjent prosjektplanen.		
AC-020201-04: Prosjektplan er sendt.		

[US-020210]	5	81:30
Som oppdragsgiver ønsker jeg at prosjektplanen inneholder informasjon om hvordan prosjektet skal gjennomføres slik at vi får innblikk i hvordan gruppen har planlagt å jobbe.		
AC-020210-01: Informasjonen om prosjektgjennomføringen er produsert.		
AC-020210-02: Informasjonen er lagt til i Prosjektplan dokumentet.		

ISSUE		
Sprengt tidsbudsjettet.		
Denne posten har fått resterende tid fra to andre poster totalt 29 timer. Øker sprintens totale tidsforbruk med 20,5 timer		

Her har vi prioritert dokumentkvalitet på det som skal leveres inn i forbindelse med første presentasjon. Vi har oppdaget et behov for noen rutinemessige endringer i forbindelse med skriving av rapporter og dokumenter, som gjør at vi lettere kan estimere tidsbehovet for ferdigstilling av dokumentasjon senere i prosjektet.

AC-020210-02: Føyd til informasjon, korrektur.

AC-020210-02: Informasjon lagt til. Prosjektplan fullført.

[US-020220]	5	2:00
-------------	---	------

Som oppdragsgiver ønsker jeg at prosjektplanen inneholder et økonomisk budsjett slik at vi får et overblikk over planlagte utgifter.

AC-020220-01: Budsjettet er produsert.

AC-020220-02: Budsjettet er lagt til i Prosjektplan dokumentet.

Flere av postene på budsjettet er estimert.

[US-020230]	5	16:00
Som oppdragsgiver ønsker jeg at prosjektplanen inneholder en overordnet risiko vurdering slik at vi kan se at gruppen har tatt høyde for risiko		
AC-020230-01: Risikovurdering er utarbeidet.		
AC-020230-02: Risikovurdering er skrevet.		
AC-020230-03: Risikovurdering er lagt til i Prosjektplan dokumentet.		

ISSUE

Denne oppgaven tar lengre tid enn vi har stipulert, behov for å utarbeide en fullstendig vurdering av hele prosjektet. Denne oppdateres så kontinuerlig og inkluderes i hver status rapport.

Utvider antall timer avsatt til oppgaven. En oversikt over risk er en lønnsom investering i forhold til tidsbruk.

[US-020231]	4	4:00
Som risk ansvarlig ønsker jeg at risiko vurdering inneholder en mekanikk risiko vurdering slik at vi kan se at gruppen har tatt høyde for risiko fagfeltets spesielle risk		
AC-020231-01: Føringer er gitt til de fagansvarlige.		
AC-020231-02: Fagansvarlige har gitt tilbakemelding.		
AC-020231-03: Mekanisk Risikovurdering er lagt til i den overordnede risikovurderingen.		

[US-020232]	4	2:30
Som risk ansvarlig ønsker jeg at risiko vurdering inneholder en software risiko vurdering slik at vi kan se at gruppen har tatt høyde for risiko fagfeltets spesielle risk		
AC-020232-01: Føringer er gitt til de fagansvarlige.		
AC-020232-02: Fagansvarlige har gitt tilbakemelding.		
AC-020232-03: Software Risikovurdering er lagt til i den overordnede risikovurderingen.		

[US-020233]	4	1:00
Som risk ansvarlig ønsker jeg at risiko vurdering inneholder en elektro risiko vurdering slik at vi kan se at gruppen har tatt høyde for risiko fagfeltets spesielle risk		
AC-020233-01: Føringer er gitt til de fagansvarlige.		
AC-020233-02: Fagansvarlige har gitt tilbakemelding.		
AC-020233-03: Elektrisk Risikovurdering er lagt til i den overordnede risikovurderingen.		

[US-020240]	5	12:00
Som oppdragsgiver ønsker jeg at prosjektplanen inneholder en oversiktlig fremvisning av user stories slik at vi lett kan se over user stories og se hva gruppen har tenkt her.		
AC-020240-01: Layout for user stories er utarbeidet.		
AC-020240-02: User stories er lagt inn i layout.		
AC-020240-03: User stories er lagt til i Prosjektplan dokumentet.		

AC-020240-01: Layout utarbeidet og lagret i eget dokument.
AC-020240-02: Lagt inn i layout.
AC-020240-03: Endret til å være et vedlegg, laget, men mangler Word malen.
AC-020240-03: Ferdig lagt til i mal, sendt til Final godkjenning

[US-020241]	5	1:00
Som oppdragsgiver ønsker jeg at user stories har et prioriteringsfelt slik at vi ser hvordan gruppen har prioritert user stories.		
AC-020241-01: Prioriterings felt er tilstede i user story layouten.		

[US-020242]	5	1:00
Som oppdragsgiver ønsker jeg at user stories inneholder en kilde slik at vi kan se hvor user storien kommer fra.		
AC-020242-01: Kilde felt er tilstede i user story layouten.		

[US-020243]	5	8:00
Som oppdragsgiver ønsker jeg at user stories som er planlagt i en sprint har acceptance criterias slik at vi kan se hvordan gruppen bedømme når en user story er ferdig		
AC-020243-01: Alle user stories planlagt i sprint1 har acceptance criteries.		
AC-020243-02: User stories layout har felt for acceptance criterias og disse er lagt til for Sprint1 User Stories.		

AC-020243-01: Acceptance criteria er reinskrevet for alle US i Sprint 1, nå lagt til.
AC-020243-02: Det er laget felt for AC i layout.

[US-020244]	5	1:00
Som oppdragsgiver ønsker jeg at user stories er merket med en unik identifikasjon slik at det er lett sporbarhet tilbake til user stories.		
AC-020244-01: Unik identifikasjon felt er tilstede i user story layouten.		

Her bruker vi US-XX XX XX nummeret som følger en user story, De får korresponderende AC-XX XX XX.

[US-020250]	5	16:00
Som oppdragsgiver ønsker jeg at prosjektplanen inneholder et tidsestimat slik at vi får et inntrykk av hva som skjer når.		
AC-020250-01: Tidsestimat for sprint 1 er utarbeidet.		
AC-020250-02: Burndown chart er utarbeidet.		
AC-020250-03: Tidsestimat og burndown chart er tilrettelagt for å legges til i prosjektplan.		
AC-020250-04: Informasjonen er lagt til i Prosjektplanen.		

[US-020251]	5	2:00
Som oppdragsgiver ønsker jeg at tidsestimatet har en tidsplan for fremtidige sprinter slik at vi kan se hva gruppen har estimert av fremtidige sprinter.		
AC-020251-01: En tidslinje med estimert planlegging blir utarbeidet.		
AC-020251-02: Tidslinjen legges til i prosjektplanen.		

Tidslinjen blir oppdatert i sprint review, og er derfor ikke en user story vi trenger å ha med i flere sprinter.

[US-020252]	5	1:00
Som oppdragsgiver ønsker jeg at tidsestimatet har en oversikt over planlagt fravær og stillstand i prosjektet slik at vi vet når det er planlagt at det skal være lite fremdrift på prosjektet.		
AC-020252-01: Planlagt ferieperioder og eksamenslesing blir lagt til på tidslinjen.		

[US-020301]	5	36:00
Som høyskole krever jeg at gruppen holder en 1.Presentasjon slik at vi kan bedømme fremgang i prosjektet.		
AC-020301-01: Presentasjonens innhold er planlagt.		
AC-020301-02: Presentasjonen fordeles.		
AC-020301-03: Presentasjonen er sammenstilt.		
AC-020301-04: Presentasjonen er gjennomgått.		
AC-020301-05: Presentasjon overskrider ikke 20 minutter.		
AC-020301-06: Presentasjon er ferdigstilt.		
AC-020301-07: Tid og rom til presentasjon og møter er bestilt.		
AC-020301-08: Veiledere, sensorer og publikum er invitert til presentasjon, og har godtatt.		

[US-020302]	4	16:00
Som ingeniør ønsker jeg at 1.Presentasjon inkluderer slides slik at vi kan ha noe visuelt å vise til.		
AC-020302-01: Slides er utarbeidet.		
AC-020302-02: Slides er sammenstilt til en helhetlig presentasjon.		
AC-020302-03: Presentasjon er godkjent av gruppen.		

Alle gruppens medlemmer bidro til å sette sammen en relevant presentasjon av oppgaven og prosjektplanen.

[US-020303]	4	16:00
Som ingeniør ønsker jeg at alle medlemmene skriver et manus til 1.Presentasjon slik at et annet gruppemedlem kan steppe inn i tilfelle sykdom.		
AC-020303-01: Alle gruppemedlemmer har utarbeidet et kortfattet manus.		
AC-020303-02: Alle Gruppemedlemmers manus er lagt i 1.presentasjonsmappen.		

[US-020304]	5	36:00
Som ingeniør ønsker jeg at vi felles øver på hva vi skal si på 1.Presentasjon slik at vi felles kan sikre kvaliteten på det som sies.		
AC-020304-01: Øvinger er gjennomgått.		
AC-020304-02: Gruppen har godkjent resultatet.		

Vi satt av hele dagen før 1.presentasjon til gjennomgang øving og forberedelser.

[US-020305]	3	16:00
Som høyskole ønsker jeg at første prototype tas med til første presentasjon slik at vi får vist frem vår konsept ide.		
AC-020305-01: Prototype er vist frem på presentasjonen.		

ISSUE		
I arbeidet med ide og design av første prototype har vi også gjort endel research relatert til videre utvikling i de neste sprinter.		
Det er brukt 7 timer til research og "oppvarming" i SolidWorks. Differanse på 6 timer, brukt på prosjektplan		

[US-021502]	4	1:00
Som oppdragsgiver ønsker jeg at dokumentasjonen skal inneholde mekaniske 3D-tegninger slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.		
AC-021502-01: 3D filer er laget å lagt til i prosjektmappe.		

Denne oppgaven er gjentakende, det vil være behov for å ha den med i alle sprinter der det skal tegnes 3d.

[US-021702]	4	16:00
Som ingeniør ønsker jeg at dokumentasjonen skal inneholde USE CASE for alle brukerooperasjoner slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.		
AC-021702-01: USE CASE er utarbeidet.		
AC-021702-02: USE CASE er lagt til i prosjektmappe.		

ISSUE		
Her bruker vi bare 16 av 32 timer. Vi har brukt denne posten til å kartlegge eksisterende kode fra ACT prosjektet og tegnet et enkelt USE case, men tiden har i hovedsak vært brukt til systemoversikt.		

Utvikling av UML til software design vil inngå i sprint 2.

[US-043001]	5	1:00
Som oppdragsgiver ønsker jeg at EE skal kunne festes på UR5 slik at de to skal kunne brukes sammen.		
AC-043001-01: EE er kompatibel med festene på UR5.		
AC-043001-02: EE er festet på UR5.		

ISSUE		
Ikke gjennomført, tas med ved utvikling av neste prototype.		

Vi finner det mer hensiktsmessig og ta dette med senere. Det koster og printe og vi har ikke behov for disse detaljene på nåværende tidspunkt.

[US-050101]	5	1:00
Som UR5 robot ønsker jeg at EE skal ikke overstige en egenvekt på 5kg slik at UR5 ikke skal få problemer med å løfte den		
AC-050101-01: Prototypens vekt er målt til under 5 kg.		

Dette kravet skal følge alle prototypene vi produserer. Derfor er det merket som gjentakende.

[US-070101]	5	4:00
Som prosjektgruppe ønsker vi god kommunikasjon med oppdragsgiver slik at vi kan sikre et godt samarbeide.		
AC-070101-01: Oppdragsgiver har fått statusrapport til rett tid.		
AC-070101-02: Alle gruppemedlemmer har møtt til avtalt tid til møter.		

Gjentagende oppgave da vi igjennom hele prosjektet skal levere statusrapport hver 14. dag

[US-070201]	5	16:00
Som prosjektgruppe ønsker vi møter med oppdragsgiver slik at vi sikrer et godt grunnlag til user stories.		
AC-070201-01: Møter er gjennomført		

Timene er postert på møtet vi hadde hos oppdragsgiver den gangen vi startet med sprint 1.

[US-070301]	5	16:00
Som Scrum master ønsker jeg at det lages et prosjekthåndteringsverktøy slik at vi kan se hvor vi er i sprinten til enhver tid.		
AC-070301-01: Verktøyet er utarbeidet		
AC-070301-02: Verktøyets bruk er gjennomgått med gruppen		
AC-070301-03: Verktøyet er godkjent av gruppen		

Det vil tilkomme noen forbedringer og mindre endringer etter hvert som vi danner oss erfaringer. Men verktøyet gjør det enkelt og holde oversikt over informasjon tilknyttet oppgavene, vi oppdaterer tidsforbruk og endringer underveis.

[US-070401]	5	2:00
Som Scrum master ønsker jeg at gruppa har et budsjett som oppdateres fortløpende slik at vi har god økonomisk kontroll, slik at det ikke blir brukt for mye penger.		
AC-070401-01: Budsjett er utarbeidet AC-070401-02: Budsjett er godkjent av oppdragsgiver		

Vi fortsetter å oppdatere etter hvert som vi får mer konkrete tall.

[US-070501]	5	8:00
Som Scrum master ønsker jeg at gruppa har et timebudsjett som oppdateres fortløpende slik at vi har oversikt over hvem som har brukt timer, og på hva.		
AC-070501-01: Tidsbudsjett er utarbeidet AC-070501-02: Tidsbudsjettets bruk er gjennomgått med gruppa AC-070501-03: Tidsbudsjettet er godkjent av gruppa		

Vi estimerer tiden vi tror vi vil bruke på de forskjellige user storiene i planleggingsfasene før hver sprint, I tillegg re-estimerer vi gjestående tid flere ganger underveis. Denne posten ble brukt til et møte for fullstendig gjennomgang av resterende tid.

[US-070601]	2	36:00
Som høyskole ønsker jeg at gruppa lager en nettside slik at vi kan følge med der, og få et inntrykk av hva oppgaven går ut på.		
AC-070601-01: Innholdet er godkjent av arbeidsgiver AC-070601-02: Nettsiden er publisert		

Et førsteutkast er publisert. Vi kommer til å kjøre oppdateringer underveis i prosjektet.

[US-070602]	5	3:00
Som prosjektgruppe ønsker vi at presentasjonene lages og legges på Websidene slik at presentasjonene alltid er tilgjengelig på nett.		
AC-070602-01: Presentasjonen er implementert i nettsiden		

Dette er et krav som gjelder for alle presentasjonene, derfor legger vi den tilbake så vi får den med i de aktuelle sprintene.

7. Konklusjon

Vi har lært mye av å jobbe sammen. Det har kommet mange gode innspill og vi er et sterkt lag. Roller er fordelt og vi har funnet plassene våre. Vi har utviklet egne verktøy for å spore arbeidet

og holde oversikt over Scrum. Vi har hatt møter meg oppdragsgiver og vi har definert oppgavens omfang. Det har blitt noen små endringer i forhold til planen, vi ser det er en utfordring og estimere tid, men vi tar til oss innspill i forhold til reestimering. Hvis vi fortsetter sånn som dette kommer det til å bli veldig bra.

Sprint retrospektiv 2

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAGSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Sprint retrospektiv 2
VERSJON / DATO	2.0 / 20. Mai 2016
FORFATTER	Thomas Wegener
SIDEANTALL	15



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



i. Abstrakt

Dette dokumentet beskriver hvordan vi har gjennomført sprint 2. Det inneholder en analyse av samarbeid og kommunikasjon, og en fullstendig gjennomgang av oppgaver knyttet til sprint 2.

ii. Innholdsfortegnelse

i. Abstrakt.....	2
ii. Innholdsfortegnelse	2
iii. Oversikt over figurer og tabeller	2
iv. Dokumenthistorie.....	3
1 Innledning	4
2 Samarbeid	4
3. Deadlines.....	4
4. User Story gjennomgang.....	4
5. Konklusjon	15

iii. Oversikt over figurer og tabeller

Figur 1: Burndown chart.....	4
Tabell 1 Dokumenthistorie	3

Vi har bevisst valgt å ikke liste alle tabellene i User Story gjennomgang kapittelet, der hver User Story vi har jobbet med i denne sprinten er representert i en egen tabell.

iv. Dokumenthistorie

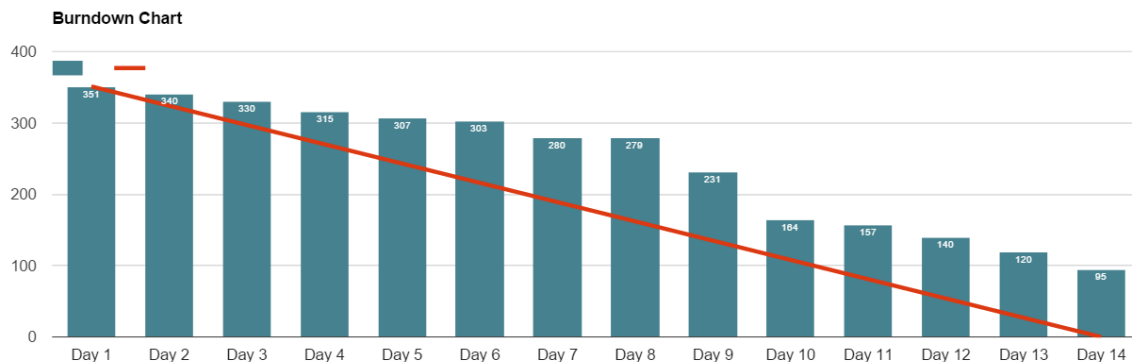
Tabell 1 Dokumenthistorie

Versjon 0.1	20.02.16	Mal av Thomas.
Referat	16.02.16	Referat fra møtet av Aina
Versjon 0.2	24.02.16	Tekst av Thomas, publisering for korrektur.
Versjon 1.0	26.02.16	Korrektur av Thomas og Monica
Versjon 1.1	15.03.16	Feilretting og klargjøring for innlevering.
Versjon 1.2	17.05.16	Lagt til Abstrakt og Konklusjon.
Versjon 2.0	20.05.2016	Espen: Korrektur og ferdigstilling

1 Innledning

Vi har gjennomført en ny sprint igjen, denne gangen kom vi raskt frem til at vi hadde tatt med flere userstories en vi kom til å rekke. Vi har fått kjenne på at tiden har vært knapp. For at vi skal lykkes, har vi behov for å gjøre gode valg mellom sprintene, det at vi valgte å dele en 4 ukers sprint i 2 har gitt noen negative konsekvenser. Når det er sagt har det vært positivt og få slutført enda en sprint, nå som vi skal lage et project burndown dokument, og en prioriteringsliste over de resterende oppgavene.

Dette dokumentet beskriver som sist; en evaluering av arbeidet og en gjennomgang av user stories.



Figur 1: Burndown chart

2 Samarbeid

Vi er enige om at gruppen fungerer bra. Vi tar opp ting med en gang, og er flinke til å lufte meninger. Det er viktig å holde fokus oppe rundt kommunikasjon. Vi vil fortsette i samme spor videre.

3 Deadlines

Vi går igjennom tidslinjen baklengs, for å stadfeste tidsfrister i prosjektet. Vi ser at det er mest hensiktsmessig om vi får prototypen (i deler) printet til sprint 4. Da skal delene settes sammen, og den skal fysisk testes, og implementeres med kode og elektriske komponenter.

4 User Story gjennomgang

Etter gjennomført sprint sitter vi nå igjen med 3 kategorier av de user storiene vi har hatt med.

- De som er utført har grå bakgrunn.
- Der det er behov for gjentakelse senere er merket med grønn bakgrunn.
- De som ikke ble gjennomført er merket med rød.

Vi oppretter issues der vi oppdager at vi må avvike fra den opprinnelige planen. Det kan være med hensyn på tidsbruk, problemer som oppstår, eller at noe ikke blir gjennomført. Issues blir ført inn i et regneark med korresponderende nummer. Vi har listet dem sammen med tilhørende user story i tabeller med gul farge.

[US-020102]	5	2:00
Som intern veileder ønsker jeg at prosjektgruppa sender ukentlig skriftlig rapport slik at jeg kan holde oversikt over fremgang.		
AC-020102-01: Skriftlig rapport er produsert.		
AC-020102-02 : Gruppa har godkjent rapporten.		
AC-020102-03 : Rapporten er sendt til intern veileder.		

Oppfølgingsdokument 5 og 6 er sendt i sprint 2.

[US-020103]	5	8:00
Som oppdragsgiver ønsker jeg at prosjektgruppa sender skriftlig status rapport hver 14.dag slik at vi blir holdt oppdatert.		
AC-020103-01: Skriftlig rapport er produsert.		
AC-020103-02 : Gruppa har godkjent rapporten.		
AC-020103-03 : Rapporten er sendt til ekstern veileder.		

Vi produserte en utfyllende sprint retrospektiv, og oppdaterte risk dokumentet så det ble sendt sammen med status rapporten.

[US-070101]	5	1:00
Som prosjektgruppe ønsker vi god kommunikasjon med oppdragsgiver slik at vi kan sikre et godt samarbeid.		
AC-070101-01: Oppdragsgiver har fått statusrapport til rett tid.		
AC-070101-02: Alle Gruppemedlemmer har møtt til avtalt tid til eventuelle møter.		

Tidsfristen ble overholdt og avtalte tider har blitt overholdt av alle gruppens medlemmer.

[US-070401]	4	3:00
Som Scrum master ønsker jeg at gruppa har et Budsjett som oppdateres fortløpende slik at vi har god økonomisk kontroll, slik at det ikke blir brukt for mye penger.		
AC-070401-01: Det er gått en ny runde med alle fagretninger der de har hatt mulighet til å komme med input.		
AC-070401-02: Informasjonen er bearbeidet og lagt inn i budsjettdokumentet.		
AC-070401-03: Informasjonen er oppdatert på alle relevante steder.		

Budsjettet oppdateres fortløpende og i sprint 2 har vi konkretisert noen av postene. Budsjettet er inkludert i selve statusrapporten.

[US-070501]	4	1:00
Som Scrum master ønsker jeg at gruppa har et Timebudsjett som oppdateres fortløpende slik at vi har oversikt over hvem som har brukt timer, og på hva.		
AC-070501-01: Alle har oppdatert timelistene sine.		

Timelistene oppdateres flere ganger i uken.

[US-070701]	5	4:00
Som oppdragsgiver ønsker jeg at det lages en Burndown-chart for ferdigstilte User stories slik at vi kan spore progresjonen på ferdigstilling av User Stories		
AC-070701-01: Dokumentet er utarbeidet.		
AC-070701-01: Burndown-chart er tilpasset og lagt til i Statusrapport		
Issue		
Denne må ferdigstilles mellom sprintene. Burndown-chart er utarbeidet.		

For å ha tilstrekkelig med data for å mene noe om hvor mye vi rekker i hver sprint, har det vært bra og komme igjennom sprint 2. Nå ser vi i løpet av 14 dager med sprint klarer mellom 25 og 17 User stories. Og denne erfaringen hjelper oss å planlegge mer realistisk når vi nå skal estimere hvor mange user stories vi kan gjennomføre i løpet av 2 sprinter i april og mai.

[US-070801]	5	2:00
Som Product Owner ønsker jeg at User Stories dokumentet holdes ved like slik at dokumentet alltid vil være oppdatert		
AC-070801-01: Dokumentet er oppdatert med de siste endringene før Sprints slutt		

Også i denne sprinten har vi reestimert gjestående tid ved flere anledninger. Vi kom ganske raskt frem til at vi hadde tatt med flere user stories en vi kom til å rekke.

[US-010701]	5	1:00
Som oppdragsgiver ønsker jeg at det skal vises prototyper slik at vi kan se hva gruppen har tenkt, og for tidlig og fortløpende V&V		
AC-010701-01: Prototype godkjent av Product Owner.		
AC-010701-02: Prototype er ferdig produsert.		
AC-010701-03: Alle mål og dimensjoner er notert for videre bruk.		
Issue		
Det vises ikke til en fysisk prototype i sprint 2. Vi følger opp design fra første sprint, Prototype 2 vises i sprint 3 på 2. pres.		

Tid, kostnad og prioriteringer har gjort at vi har valgt å avbryte alle user-storiene som omhandler en fysisk prototype i sprint 2. Vi flytter dem til montering i begynnelsen av sprint 4. Og fokuserer på ferdigstilling av tegningene i sprint 3.

[US-02150101]	4	4:00
Som bachelorgruppe ønsker vi at dokumentasjon skal inneholde mekaniske 2D tegninger for prototype sprint 2 slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.		
AC-02150101-01: 2D tegninger er lagt ved i teknologidokumentet		
Issue		
2D tegninger utsettes til vi er ferdige med å gjøre tilpasninger på designet.		

[US-02150201]	4	1:00
Som bachelorgruppe ønsker vi at dokumentasjon skal inneholde mekaniske 3D filer for prototype sprint 2 slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.		
AC-02150201-01: 3D filer er lagret på hensiktsmessig plassering.		

[US-021505]	5	16:00
Som bachelorgruppe ønsker vi at det skal utarbeides et grunnlag for et mekanisk teknologidokument slik at det tilrettelegges for dyptgående forståelse av oppbygging og hensikt.		
AC-021505-01: Ramme for dokumentet er produsert med oppsett og nødvendig punkter for innhold		

AC-021505-02: Informasjon funnet i Sprint2 er lagt inn i dokumentet der det er relevant

[US-043001]	5	1:00
Som oppdragsgiver ønsker jeg at EE skal kunne festes på UR5 slik at de to skal kunne brukes sammen.		
AC-043001-01: Interface er ferdig designet AC-043001-02: Interface er godkjent av PO AC-043001-01: Interface er ferdig printet AC-043001-03: Interface er fysisk testet koblet på UR5		
Issue		
Parter er modellert og planlagt, vi produserer når flere deler skal printes. Har opprettet dialog med ansvarlig person ved HSN for printing av deler.		
[US-043002]	5	4:00
Som ingeniør ønsker jeg at det fysiske grensesnittet UR5/EE skal bestå av to deler før kobling mot EE slik at EE kan monteres på/av på en hensiktsmessig måte		
AC-043002-01: Koblingen er designet og printet AC-043002-02: Koblingen er prøvd i praksis		
Issue		
Parter er modellert og planlagt, vi produserer når flere deler skal printes. Har opprettet dialog med ansvarlig person ved HSN for printing av deler.		
[US-043003]	5	2:00
Som ingeniør ønsker jeg at koblingen til UR5 tilrettelegges for festing med 4 M6 skruer rundt origo slik at den skal passe korrekt og være festet stødig uten å skade gjengene på UR5		
AC-043003-01: Design av interface er tilpasset monteringshull på UR5		
[US-043004]	5	2:00
Som ingeniør ønsker jeg at det skal tas en faglig vurdering på valg av løsningene for US-043001, US-043001, US-043001 og begrunnes i teknologidokumentet for EE slik at vi sikrer at skrueforbindelser og grensesnitt ikke overbelastes		
AC-043004-01: Det begrunnes kort i tekst som føres inn i teknologidokumentet.		
[US-050401]	3	2:00

Som ingeniør ønsker jeg at EE har en luke for å komme til komponenter slik at EE er lett å gjøre vedlikehold og tilpasninger på.

AC-050401-01: Luken er designet

AC-050401-02: Luken er prøvd i praksis og gir enkel tilgang til komponenter

[US-044001]	5	1:00
Som oppdragsgiver ønsker jeg at EE skal kunne betjene CG slik at tester som inkluderer betjening av CG kan automatiseres.		
AC-040001-01: EE er designet og godkjent av begge fra maskin		
AC-040001-02: EE skal printes og settes sammen		
AC-040001-03: EE og CG er kjørt sammen i simulering		
AC-040001-04: EE og CG er testet sammen fysisk og fungerer		
Issues		
Design og simulering videreføres til sprint 3. Fysiske deler til sprint 4		

[US-044002]	5	8:00
Som oppdragsgiver ønsker jeg at EE skal kunne holde inne palmknapp slik at tester som inkluderer betjening av palmknapp kan automatiseres.		
AC-044002-01: EE holder palmknapp inne ved simulering		
AC-044002-02: EE holder palmknapp inne ved fysisk test		
Issues		
Design og simulering videreføres til sprint 3. Fysiske deler til sprint 4		

[US-04400201]	5	24:00
Som ingeniør ønsker jeg at EE tilpasses til å holde inne palmknapp og modelleres detaljert i SolidWorks slik at det klargjøres for neste fase, der elektronikk skal lage styring.		
AC-04400201-01: 3D modell detaljert med klart definert funksjon for betjening av Palmknapp		
AC-04400201-02: Modellen er designet med nødvendige volum for elektronikk og komponenter		

Det er tatt høyde for at den skal kunne holde inne palmknapp. Elektronikk er bestilt, og det er tilrettelagt for disse.

[US-044004]	5	8:00
Som ingeniør ønsker jeg at EE skal ha en anordning i håndflaten til å passe CG slik at man oppnår stødighet i grepet		
AC-044004-01: EE utstyres med «formbar» anordning for best friksjon i grep		
AC-044004-02: EEs anordning i håndflaten gir stødig grep om CG ved fysisk test		

[US-080102]	5	8:00
Som ingeniør ønsker jeg at vi simulerer CG og EE-Palmtrigger sammen slik at vi kan verifisere at de vil fungere i teorien, før de produseres.		
AC-080102-01: Modell tilsvarende CG er modellert opp i SW		
AC-080102-02: Modell tilsvarende CG og EE er simulert sammen og fungerer		
Issue		
Design og simulering videreføres til sprint 3. Fysiske deler til sprint 4		

[US-081001]	5	3:00
Som ingeniør ønsker jeg at det tas fram en 3D modell av CG slik at de kan simuleres sammen		
AC-081001-01: Modell tilsvarende CG er modellert opp i SW		

Vi har tegnet en forenklet modell av CG, den vil kunne fungere som en test CG. Bevegelse/begrensninger i CG og knapper kan omgjøres/tillegges etter behov. Mål fra 2D CG er brukt der de har vært tilgjengelige.

[US-051202]	5	4:00
Som ingeniør ønsker jeg at kravene til kontroller kartlegges og det gjøres et valg av kontroller slik at den kan bestilles så fort som mulig		
AC-051202-01: Kontrollkort skal velges etter behov for applikasjoner til EE og WS.		
AC-051202-02: Kontrollkort skal bestilles.		
Issue		
Denne videreføres til sprint 3. Vi bruker Arduino mega kortet vi eier privat, for testing frem til det er bestilt egen til prosjektet. Dette vil skje i sprint 3.		

[US-051401]	5	32:00
-------------	---	-------

Som PO ønsker jeg at det kartlegges hvilke komponenter vi planlegger å bruke for å styre EE slik at disse kan bestilles så fort som mulig, og designet kan lages med hensyn på komponentene som skal inni

AC-051501-01: Det skal velges et system som kan regulere spenningen fra 24V til 6V for å kunne levere spenning til Servoer og Arduino.

AC-051501-02: Det skal vises en krets for teoretisk verifisering.

AC-051501-03: Komponent(er) researched for bestilling i eventuell sprint 4.

[US-051001]	5	2:00
Som UR5 robot krever jeg at komponenter i EE ikke bruker mer enn 600mA slik at jeg er sikker på at jeg kan forsyne nok strøm til EE.		
AC-051001-01: Beregninger er gjort ved valg av komponenter så dette er på plass.		
Issue		
Det gjenstår en test av denne US-en, for å se om det holder med bruk av 2 servoer av gangen. Foreslår å tilføye en [AC-051001-02].		

[US-051101]	5	1:00
Som UR5 robot krever jeg at elektronikken i EE skal kunne kjøre på 24V eller 12V slik at jeg ikke dreper elektronikken, da dette er det jeg kan levere.		
AC-051101-01: Valg av kraftoverføringer er foretatt med hensyn til "ramme" på spenning lik 12 el 24 volt.		

[US-050103]	5	10:00
Som End-Effector krever jeg at jeg har komponenter som er sterke nok til å fullføre bevegelser slik at også tester som krever disse bevegelsene kan gjøres.		
AC-050103-01: Komponenter som tilfører kraft, skal resultere i en styrke på minst 25N i angrepspunktet		
AC-050103-02: Det skal vises til utregninger som legges i teknologidokument		

[US-02160101]	4	16:00
Som bachelorgruppe ønsker vi at WS-sensors oppbygging dokumenteres med kretstegninger slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.		
AC-021604-01: Det er produsert korrekte skjemaer over kretsene.		
AC-021604-02: Skjemaer er lagret på hensiktsmessig sted.		
Issue		

Ikke mulig å fullføre før informasjonen som mangler i [US-046102] er på plass.

[US-02160201]	3	4:00
Som bachelorgruppe ønsker vi at BOM lages for WS-sensoren slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.		
AC-021605-01: BOM er produsert og lagt i teknologidokument		
Issue		
Ikke mulig å fullføre før informasjonen som mangler i [US-046102] er på plass.		

[US-02160301]	2	4:00
Som bachelorgruppe ønsker vi at dokumentasjonen skal inneholde data ark for alle 3.parts komponenter brukt på WS-sensor slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.		
AC-021606-01: Dataark er samlet så langt det lar seg gjøres å lagt i egen mappe		
Issue		
Ikke mulig å fullføre før informasjonen som mangler i [US-046102] er på plass.		

[US-046101]	5	1:00
Som oppdragsgiver ønsker jeg at sensor skal registrere alle avfyringer slik at alle avfyringer som skjer registreres.		
AC-046101-01: US-046102 er gjort.		
AC-046101-02: US-046103 er gjort.		
AC-046101-03: US-046104 er gjort.		
Issue		
Ikke mulig å fullføre før informasjonen som mangler i [US-046102] er på plass.		

[US-046102]	5	4:00
Som ingeniør ønsker jeg at tilkoblingsmulighetene på WS for sensor kartlegges slik at sensor kan designes til å passe overens med denne utgangen.		
AC-046102-01: Møte gjennomføres/informasjon skaffes		
AC-046102-02: Informasjonen er lagret på hensiktsmessig sted uten sporbarhet tilbake til WS		
Issue		

Mangelfull informasjon fra arbeidsgiver, avventer til vi får mer informasjon.

[US-046103]	5	16:00
Som ingeniør ønsker jeg at det designes en krets som tolker avfyrings signalet, og oversetter dette til en 5V puls slik at signalet kan mottas av Arduinoen.		
AC-046103-01: Ideen til krets er notert ned AC-046103-02: Konseptet er simulert i for eksempel OrCad for å teste om det er mulig AC-046103-03: Konseptet er testet i praksis		
Issue		
AC-046103-01: Ideen til krets er notert ned. AC-046103-02: Konseptet er delvis simulert i OrCad. Usikker på resultatet, må tas opp igjen. AC-046103-03: Ikke mulig å fullføre før informasjonen som mangler i [US-046102] er på plass.		

[US-046104]	5	4:00
Som ingeniør ønsker jeg at Arduino kodes til å sende beskjed til skjerm ved høyt signal på inn porten slik at vi kan verifisere at kretsen fungerer og fyringer kan registreres.		
AC-046104-01: Kode skrevet slik at ved merket avfyring vil det umiddelbart bli sendt beskjed til testmaskin om dette.		

AC-046104-01: Interrupt er laget, slik at ved høyt signal sendes det beskjed til skjerm, denne er lett å endre til ønsket beskjed/teknologi for kommunikasjon med testmaskin.

[US-046202]	5	4:00
Som ingeniør ønsker jeg at det klargjøres for at Arduino skal kommunisere med testmaskin slik at dette er klart for når software lager loggsystemet		
AC-046202-01: Kode er laget på Arduino for håndtering av kommunikasjon til maskin. AC-046202-02: Kode er testet så langt det lar seg gjøre før loggsystemet er på plass.		

"AC-046202-01: Kode er laget på Arduino for håndtering av kommunikasjon til maskin. Gjort via USB, seriell kommunikasjon.

AC-046202-02: Kode er testet så langt det lar seg gjøre før loggsystemet er på plass. Siden det ikke er noe loggsystem på plass, så er det ikke definert akkurat hva som skal sendes, men dette er en lett endring når det blir mulig."

[US-046302]	5	16:00
-------------	---	-------

Som ingeniør ønsker jeg at det klargjøres for at Arduino skal kunne kommunisere trådløst slik at dette er klart for når software lager loggsystemet

AC-046302-01: Kode er laget på Arduino for håndtering av trådløs kommunikasjon.

AC-046302-02: Kode er testet så langt det lar seg gjøre før loggsystemet er på plass.

"AC-046202-01: Kode er laget på Arduino for håndtering av kommunikasjon til maskin. Gjort via Bluetooth, seriell kommunikasjon.

AC-046202-02: Kode er testet så langt det lar seg gjøre før loggsystemet er på plass. Siden det ikke er noe loggsystem på plass, så er det ikke definert akkurat hva som skal sendes, men dette er en lett endring når det blir mulig."

[US-021702]	2	14:00
Som ingeniør ønsker jeg at dokumentasjonen skal inneholde USE CASE for alle brukeroparasjoner slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.		
AC-021702-01 Teknologidokumentet inneholder use-case for alle brukeroparasjoner.		
AC-021702-02 Use-case er godkjent av begge fra software		

[US-021706]	5	4:00
Som bachelorgruppe ønsker vi at det lages en mal for software teknologidokument med disposisjoner slik at vi kan begynne på teknologidokumentet i denne sprint 2		
AC-021706-01 Teknologidokument er opprettet.		
AC-021706-02 Teknologidokument er satt opp med disposisjon.		
AC-021706-03 Det er skrevet abstrakt og innledning.		

[US-040101]	5	28:00
Som oppdragsgiver ønsker jeg at software skal kunne kommunisere med robotarm via TCP slik at signaler kan sendes og mottas mellom disse to.		
AC-040101-01 Software kan sette opp tilkobling med robotarm.		
AC-040101-02 Det er programmert en listener loop på robotarm.		
AC-040101-03 Det kjører en testsekvens når tilkobling skjer for å vise at tilkobling skjer.		

Vi har modifisert eksisterende kode, så den ikke prøver å skrive til noen database. Det var dette som gjorde at vi fikk problemer med software avhengigheter. Vi programmerer muligheten for å skrive til fil isteden. Så blir hele systemet mye enklere og konfigurere. Vi har skilt ut

våpenstasjonen så den ikke er en nøkkel i nettverket. Bruker resten av dagen på og modifisere koden og bli kjent i systemet.

[US-070603]	3	8:00
Som høyskole ønsker jeg at nettsidene inkluderer en Blogg slik at vi enklere kan følge med på hva som skjer med prosjektet.		
AC-070603-01 Blogg er implementert på hjemmesiden		
AC-070603-02 En første post er lagt ut		

[US-070604]	3	1:00
Som høyskole ønsker jeg at nettsidene inkluderer en presentasjon av gruppens medlemmer slik at det er lett å se både for oss, men også for andre, hvem som er en del av gruppa.		
AC-070604-01 Presentasjon av gruppemedlemmer er lagt ut		
AC-070604-02 Alle har fått muligheten til å oppdatere hva som skal stå der.		

[US-070901]	5	14:30
Som oppdragsgiver ønsker jeg at riskdokumentet oppdateres fortløpende og brukes underveis i prosjektet slik at risiko hele tiden er i fokus for prosjektgruppen.		
AC-070901-01: Risk ansvarlig har gått runden og samlet inn ny info om risk fra alle fagansvarlige		
AC-070901-02: Riskdokument er oppdatert		
AC-070901-03: Eventuelt annet risk ansvarlig føler må gjøres		

Det legges ut nye føringer for oppdatering av risiko i løpet av siste halvdel uke 8.

5 Konklusjon

Vi har fått lære at god planlegging krever tid. Vi har også brukt mye tid i starten på ferdigstilling av Sprint 1. Vi satser på at vi får bedre rutiner når vi har vært igjennom noen ganger. Sprint 2 har vært spennende. Vi har begynt å fordype oss i eksisterende kildekode. Vi kommer til å skrive mye på nytt, men tar med oss mange tanker og inspirasjon. Vi har begynt arbeidet med teknologi og designdokumenter. Vi tar med oss erfaringer som; at noen må ta ansvar for at vi holder fokus og at vi skal effektivisere sprint evalueringene. Vi har også tatt til oss innspillet om å lage et «project burndown diagram» som viser hvor mange User Stories vi får oppfylt i hver sprint og hvor mange vi tror vi ikke får gjennomført, for så å sette opp en prioritert liste for gjennomgang med oppdragsgiver.

Sprint retrospektiv 3

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAGSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Sprint retrospektiv 3
VERSJON / DATO	2.0 / 20. Mai 2016
FORFATTER	Thomas Wegener
SIDEANTALL	12



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



i. Abstrakt

Dette dokumentet beskriver hvordan vi har gjennomført sprint 3. Det inneholder en analyse av samarbeid og kommunikasjon, og en fullstendig gjennomgang av oppgaver knyttet til sprint 3.

ii. Innholdsfortegnelse

i. Abstrakt.....	2
ii. Innholdsfortegnelse	2
iii. Oversikt over figurer og tabeller	2
iv. Dokumenthistorie.....	3
1. Innledning	4
2. Samarbeid	5
3. Kommunikasjon	5
4. Administrasjon	6
4. User Story gjennomgang.....	6
5. Konklusjon	11

iii. Oversikt over figurer og tabeller

Figur 1	4
---------------	---

Tabell 1 Dokumenthistorie	Error! Bookmark not defined.
---------------------------------	-------------------------------------

Vi har bevisst valgt å ikke liste alle tabellene i «User Story» gjennomgang kapittelet, der hver «User Story» vi har jobbet med i denne sprinten er representert i en egen tabell.

iv. Dokumenthistorie

Tabell 1 Dokumenthistorie

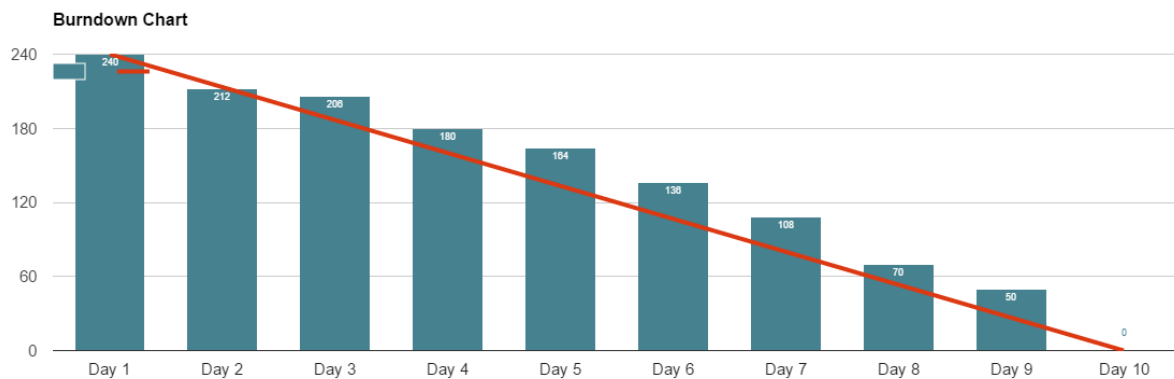
Versjon 0.1	20.02.16	Mal av Thomas.
Referat	19.03.16	Spørreskjema av Thomas
Versjon 0.9	13.04.16	Flytte over alle felt fra Sprint 3
Versjon 1.0	14.02.16	Innledning og evalueringstekst Thomas
Versjon 1.1	19.05.16	Ferdigstilt av Thomas
Versjon 2.0	20.05.16	Korrektur og gjennomlesing av Carl Martin

1. Innledning

Vi avsluttet før påske med å ferdigstille sprint 3. Under eksamensperioden ble det sendt ut et spørreskjema for å hente inn en evaluering fra hver enkelt. I etterkant av evalueringen har hvert enkelt gruppe medlem hatt en medarbeidersamtale med gruppeleder, og alle de tingene som kom opp i forbindelse med spørsmålene er drøftet. På denne måten har vi spart masse tid på møter, det var verdt å prøve det. Det kan til og med hende at vi gjør det igjen.

Kapitelene *Samarbeid*, *Kommunikasjon* og *Administrasjon* inneholder svar direkte fra spørreundersøkelsen. Vi har valgt å beholde tabellene i «review» – dokumentene, så det ikke blir så mye frem og tilbake mellom rapport og referansetabell.

Dette dokumentet beskriver som sist; en evaluering av arbeidet. Det er en gjennomgang av «user stories» som ble planlagt i sprinten, og hvordan vi har arbeidet med dem.



Figur 1: «Burndown chart»

2. Samarbeid

Syns vi har klart oss ganske bra, men vi har forbedringspotensiale. Er hyggelig å jobbe med dere.

Samarbeidet i gruppen er stort sett meget bra, men sprint 3 preges av en i overkant hektisk periode hvor kommunikasjonen ikke er på topp.

Sprint 3 ble noe hektisk for enkelte, dette kunne vært gjort annerledes.

Sprint 3 gikk for det meste bra. Det dukket opp noen «issues» helt mot slutten, men det har blitt ordnet opp i interne møter, og vi ser fremover mot ny «sprint».

Ønsker mer kjernetid (4/5 hverdager) 9-15.

Samarbeidet har fungert strålende, alt fungerer godt når vi følger modellen, har daglige møter, konsentrerer oss om det som er planlagt, har fokus og jobber som et lag. I sprint 3 har vi hatt litt problemer med flere av disse punktene oppunder innleveringsfristen.

3. Kommunikasjon

Syns vi har klart oss ganske bra, men vi har forbedringspotensiale. Vi må bli flinkere til å kommunisere klart og tydelig med en gang noe kommer opp, og ta saker med hele gruppa oftere. Her er jevnlig kommunikasjonsmøter bra.

Denne sprinten har vi fått kjenne på hvor viktig kommunikasjon egentlig er, og har som mål å finne tilbake til rett spor når det gjelder dette.

I denne sprinten hadde vi et svært bra møte, hvor vi fikk snakket om ting som har med det mellommenneskelige i gruppa å gjøre. På tross av noen konflikter, vil jeg likevel si det var bra på bakgrunn av det møtet.

Skriftlig avtale slik at det ikke oppstår noen misforståelser, viktig at man sier ifra når man ikke forstår hva andre mener. Eventuelt skriftlig bekreftelse på enighet. (Sikkerhet for ingen misforståelser og kan peke tilbake på hva som er blitt enighet om)

Gruppe kommunikasjonen fungerte fint i Sprint 3, litt issues med kommunikasjon mellom medlemmer, men det er lite vi kan gjøre i fellesskap med det. Hver enkel må bare passe på at de gjør seg forstått og at den andre parten har forstått det samme.

Kommunikasjonen har vært i fokus, allikevel har vi hatt noen misforståelser. Vi trenger fortsette det gode fokuset og trekke til oss erfaringer, på den måten vi vi blir bedre.

4. Administrasjon

Alltids forbedringspotensiale!

Administrasjonen taklet det hektiske bra.

Her synes jeg alt gikk fint, verktøyene gjør jobben og for det meste klarte vi å estimere greit.

Ingenting å påpeke, synes det fungerer greit.

Det kan være utfordrende å produsere mengder med teknisk dokumentasjon, og i tillegg holde styr på alt som skjer i prosjektgruppa. Syntes teamet fungerer godt, vil fortsette å gjøre så godt jeg kan.

5. User Story gjennomgang

Etter gjennomført sprint sitter vi nå igjen med 3 kategorier av de user storiene vi har hatt med.

- De som er utført har grå bakgrunn.
- Der det er behov for gjentakelse senere er merket med grønn bakgrunn.
- De som ikke ble gjennomført er merket med rød.

Vi oppretter «issues» der vi oppdager at vi må avvike fra den opprinnelige planen. Det kan være med hensyn på tidsbruk, problemer som oppstår, eller at noe ikke blir gjennomført. «Issues» blir ført inn i et regneark med korresponderende nummer. Vi har listet dem sammen med tilhørende «user story» i tabeller med gul farge.

[US-010701]	5	1:00
Som oppdragsgiver ønsker jeg at det skal vises prototyper slik at vi kan se hva gruppen har tenkt, og for tidlig og fortløpende V&V		
AC-010701-01 : Prototype godkjent av Product Owner.		
AC-010701-02 : Prototype er ferdig illustrert i 3d tegning		

"Det ble vist til en 3D modell av prototypen på presentasjonen og i testverifikasjonsdokumentet. Men - her er det klare mangler i designdokumentet for mekanikk, samt at det ikke ble levert filer

fra Solid Works. Dette burde vært bedre, og rettes opp til neste og siste hovedrapport skal leveres. "

Referanse fra testverifikasjon: T-M5-8

[US-020102]	5	2:00
Som intern veileder ønsker jeg at prosjektgruppa sender ukentlig skriftlig rapport slik at jeg kan holde oversikt over fremgang.		
AC-020102-01 : Skriftlig rapport er produsert.		
AC-020102-02 : Gruppa har godkjent rapporten.		
AC-020102-03 : Rapporten er sendt til intern veileder.		

Blir kun ett oppfølgingsdokument denne sprinten, Ligger klart for utfylling i mappe. Ferdigstilles og sendes inn mandag 14.03. Rettelse; lager et oppfølgingsdokument for mulig innsending 21.03, som vil inneholde foreløpig foregående uke.

AC-020102-01: rapport(er) produsert. Gruppen har fått beskjed, for selv å fylle inn timer.

AC-020102-02: Hver og en har lest igjennom tekst og fylt inn eget timebruk. Gruppen har godkjent dokument.

AC-020102-03: Scrum Master har sendt rapport til intern veileder.

[US-020103]	5	16:00
Som oppdragsgiver ønsker jeg at prosjektgruppa sender skriftlig status rapport hver 14.dag slik at vi blir holdt oppdatert.		
AC-020103-01 : Skriftlig rapport er produsert.		
AC-020103-02 : Gruppa har godkjent rapporten.		
AC-020103-03 : Rapporten er sendt til ekstern veileder.		

[US-070101]	5	6:00
Som prosjektgruppe ønsker vi god kommunikasjon med oppdragsgiver slik at vi kan sikre et godt samarbeide.		
AC-070101-01 : Oppdragsgiver har fått statusrapport til rett tid.		
AC-070101-02 : Alle Gruppemedlemmer har møtt til avtalt tid til eventuelt møter.		

Vi har sendt «Sprint review» fra «sprint» 2 sammen med statusrapport. Det er blitt avholdt møte med oppdragsgiver vedrørende reduksjon i antall userstories. Gruppen har besluttet at denne prioriteringen skal gjøres på nytt. Se vedlegg til neste statusrapport.

[US-070401]	4	1:00
Som Scrum master ønsker jeg at gruppa har et Budsjett som oppdateres fortløpende slik at vi har god økonomisk kontroll, slik at det ikke blir brukt for mye penger.		
AC-070401-01 : Det er gått en ny runde med alle fagretninger der de har hatt mulighet til å komme med «input».		

AC-070401-02 : Informasjonen er bearbeidet og lagt inn i budsjett - dokumentet.
AC-070401-03 : Informasjonen er oppdatert på alle relevante steder..

Budsjett har blitt oppdatert, og gruppemedlemmer har fått mulighet for «input».

[US-070501]	4	1:00
Som Scrum master ønsker jeg at gruppa har et Timebudsjett som oppdateres fortløpende slik at vi har oversikt over hvem som har brukt timer, og på hva.		
AC-070501-01 : Alle har oppdatert timelistene sine.		

Alle gruppens medlemmer har ført timer fortløpende.

[US-070702]	5	19:00
Som intern sensor krever jeg at «Project burndown» skal fremstilles på en måte, så oppdragsgiver kan få muligheten til å se hvilke userstories vi ikke rekker. og være med på å prioritere userstories slik at vi har en felles oppfatning av målet, og at de kan hjelpe oss å prioritere de som er viktigst.		
AC-070702-01 : Godkjent av gruppen		
AC-070702-02 : Godkjent av oppdragsgiver		

Det ble foretatt en prioritering med halvparten av gruppen, de andre medlemmene fikk mulighet til å komme med innspill i etterkant. Ser at det var uheldig at alle ikke var med på dette møtet og tar med oss den erfaringen videre slik at alle blir inkludert. Gruppen har besluttet at denne prioriteringen skal gjøres på nytt. Se vedlegg til neste statusrapport

[US-070801]	5	19:00
Som Product Owner ønsker jeg at «User Stories» dokumentet holdes ved like slik at dokumentet alltid vil være oppdatert		
AC-070801-01 : Dokumentet er oppdatert med de siste endringene før Sprints slutt		

AC-070801-01: Dokumentet er oppdatert med de siste endringene før Sprints slutt, og lagt til i prosjektplan - dokumentet.

[US-070901]	5	13:00
Som oppdragsgiver ønsker jeg at Riskdokumentet oppdateres fortløpende og brukes underveis i prosjektet slik at RISK hele tiden er i fokus for prosjektgruppen.		
AC-070901-01 : Risk dokumentet er oppdatert		
AC-070901-02 : Det er avholdt et møte der vi gjennomgår middels og høy risk		

Møte med oppdragsgiver gjennomført, fått råd og veiledning. Har oppdatert risikoanalysen med (herfra) fokus på rød kode da hoved - analysen blir for omfattende å håndtere. Jobbes frem nytt verktøy for håndtering i Excel, dette vil gjøre håndteringen mest mulig effektiv og oversiktlig. Oppdateringsmøte (1 time) er avholdt torsdag 10.03.

[US-020401]	5	6:00
Som høyskole krever jeg at gruppen holder en 2.presentasjon slik at vi kan bedømme fremgang i prosjektet.		
AC-020401-01 : Presentasjon nr. 2 er holdt		
[US-020402]	5	12:00
Som ingeniør ønsker jeg at 2.presentasjon inkluderer slides slik at vi kan ha noe visuelt å vise til.		
AC-020402-01 : Slides er produsert		
AC-020402-02 : Slides er godkjent av gruppen		
[US-020403]	5	6:00
Som ingeniør ønsker jeg at alle medlemmene skriver et manus til 2.presentasjon slik at et annet gruppemedlem kan steppe inn i tilfelle sykdom.		
AC-020403-01 : Alle har lagt manus inn på Google disk		
[US-020404]	5	36:00
Som ingeniør ønsker jeg at vi felles øver på hva vi skal si på 2.presentasjon slik at vi felles kan sikre kvaliteten på det som sies.		
AC-020404-01 : Øvinger er gjennomgått		
[US-020801]	5	1:00
Som høyskole ønsker jeg at gruppen produserer MINST et Teknologidokument for hvert fagfelt slik at vi kan se at gruppen har gjort overveide valg i løpet av prosessen med tanke på produktets oppbygging og kapasitet.		
AC-020801-01 : Minst et teknologidokument er produsert		
AC-020801-02 : Minst et teknologidokument er levert		
[US-021001]	5	25:00
Som høyskole ønsker jeg at det lages et testverifikasjons dokument slik at testresultater leveres samlet i dokumentasjonen.		
AC-021001-01 : Testverifikasjonsdokument er produsert		
AC-021001-02 : Testverifikasjonsdokument er levert		

I dette dokumentet ble det brukt endel mer tid enn estimert da det var uklarheter om hvor grundig innhold skulle være. Det var behov for en tekst rundt test og gangen i test, samt testvekting. I tillegg knyttet vi dokumentet direkte til risiko, som krevde litt mer tid enn først antatt [Aina]. 1. Grunnlag/forklaring for test og vekting er produsert. Testing knyttet opp mot risiko er produsert. 2. Tabeller fylt inn og godkjent av gruppen. 3. Dokumentet er ferdigstilt og levert.

[US-021506]	5	30:00
Som oppdragsgiver ønsker jeg at Det skal utarbeides et EE teknologidokument slik at det gis en dyptgående innsikt/forståelse av oppbygging og hensikt		
AC-02150601: EE teknologidokument er produsert		
AC-02150602: EE teknologidokument er levert		
AC-02150603: EE teknologidokument har med både elektro og mekanisk materiale		

Dokumentet ble endret til separerte dokumenter for elektroniske komponenter, mekanikk og design.

[US-051202]	5	4:00
Som ingeniør ønsker jeg at kravene til kontroller kartlegges og det gjøres et valg av kontroller slik at den kan bestilles så fort som mulig		
AC-051202-01 : Kontrollkort skal velges etter behov for applikasjoner til EE og WS.		
AC-051202-02 : Kontrollkort skal bestilles.		

Ved litt «research» av type Kontrollere som finnes endte vi opp med å bruke en Arduino ATmega 2560. Vi er kjent med dette systemet fra tidligere og kontrolleren som er blitt valg dekker behovet vårt godt.

[US-051501]	5	4:00
Som Elektro ansvarlig ønsker jeg at det kartlegges en løsning for transformering av spenningen ut fra UR5 slik at komponentene og kontrolleren får korrekt spenning		
AC-051501-01 : Det skal velges et system som kan regulere spenningen fra 24V til 6V for å kunne levere spenning til Servoer og Arduino.		
AC-051501-02 : Det skal vises en krets for teoretisk verifisering.		
AC-051501-03 : Komponent(er) «researched» for bestilling i eventuell sprint 4.		

Timene til [US-051501] vil bli brukt tid til møte med servo - avdelingen til KPS for å oppklare hvordan dette skal håndteres. AC-051501-01. AC-051501-02. AC-051501-03: Faller bort inntil videre. Trenger behov for å kunne kartlegge forbruk av systemet ved testing før dette kan bli bestemt. "AC-051202-01: Det vil bli brukt en Arduino ATmega 2560 AC-051202-02: Ikke oppfylt siden det ikke er bestilt egent for vårt bruk."

[US-021701]	5	16:00
Som oppdragsgiver ønsker jeg at dokumentasjonen skal inneholde UML diagrammer for all egenutviklet Software slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon. /dele opp denne når vi har bestemt hvilke diagrammer vi skal ha med		
AC-021703-01 : Klassediagrammer er utarbeidet		
AC-021703-02 : Klassediagrammer blir vist frem og forklart i teknologidokumentet		

UML og planer for Software produsert og dokumentert.

[US-021705]	4	20:00
Som oppdragsgiver ønsker jeg at det lages et Software teknologidokument slik at oppdragsgiver har dette til bruk i videreføring av prosjektet		
AC-021701-01 : Software teknologidokument er korrekturlest		
AC-021701-02 : Godkjent av gruppen		

Software teknologidokument produsert innen fristen.

[US-070605]	3	1:00
Som bachelorgruppe presentasjon 2 er tilgjengelig på nettsiden slik at vi lett kan informere våre interessenter		
AC-070605-01 : 2 presentasjon er lagt tilgjengelig på nettsiden		

[US-070607]	3	1:00
Som bachelorgruppe ønsker jeg at blogg oppdateres ukentlig slik at vi lett kan informere våre interessenter		
AC-070607-01 : Webansvarlig har lagt ut noe om hva vi driver med		

Nettsidene ble oppdatert med en bloggpost under denne sprinten.

[US-080102]	5	16:00
Som ingeniør ønsker jeg at vi simulerer CG og EE-Palm trigger sammen slik at vi kan verifisere at de vil fungere i teorien, før de produseres.		
AC-080102-01 : Modell tilsvarende CG er modellert opp i SW		
AC-080102-02 : Modell tilsvarende CG og EE er simulert sammen og fungerer		

Det ble brukt endel timer i Solid Works for en forenklet visuell presentasjon av UR5 tilkoblet interface, komponentrom og EE sammenstilt mot CG. Dette fordi simulering manglet fullstendige parametere med hensyn på kraft (Og utfordring med 2016-utgaven, ting på menyen og funksjoner manglet). Forenklet presentasjon var ment å vises i "eDrawings". Ser fremover og jobber videre til 3.presentasjon

Referanse fra testverifikasjon: Tabell 9: T-M7

6. Konklusjon

Planleggingen har gått mye bedre denne sprinten. Vi har i sprint 3 jobbet målrettet med justeringer, valg av deler, og skrevet på design og teknologidokumenter til innlevering 2. I tillegg har det vært et stort fokus på 2 presentasjon. Denne sprinten avsluttes inn i påske og eksamensperiode etter at andrepresentasjon ender med et oppvaskmøte. Gruppen er positiv ut av dette møtet og er bestemt på å fortsette den gode trenden, vi velger å lære av det som har

vært og tar meg oss at vi skal kommunisere ekstra tydelig når det er noe vi forventer av hverandre.

Sprint retrospektiv 4

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAGSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Sprint retrospektiv 4
VERSJON / DATO	2.0 / 20. Mai 2016
FORFATTER	Thomas Wegener
SIDEANTALL	23



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



i. Abstrakt

Dette dokumentet beskriver hvordan vi har gjennomført sprint 3. Det inneholder en analyse av samarbeid og kommunikasjon, og en fullstendig gjennomgang av oppgaver knyttet til sprint 3.

ii. Innholdsfortegnelse

i. Abstrakt.....	2
ii. Innholdsfortegnelse	3
iii. Oversikt over figurer og tabeller	4
iv. Dokumenthistorie.....	5
1 Innledning	6
2 Data	7
3 Elektro.....	7
4 Maskin	7
5 Samarbeid	7
6 Kommunikasjon	8
7 Administrasjon	8
8 User Story gjennomgang	9
9 Konklusjon	23

iii. Oversikt over figurer og tabeller

Figur 1: Burndown chart..... 6

Tabell 1 Dokumenthistorie 5

Vi har bevist valgt å ikke liste alle tabellene i User Story gjennomgang kapitlet, der hver User Story vi har jobbet med i denne sprinten er representert i en egen tabell.

iv. Dokumenthistorie

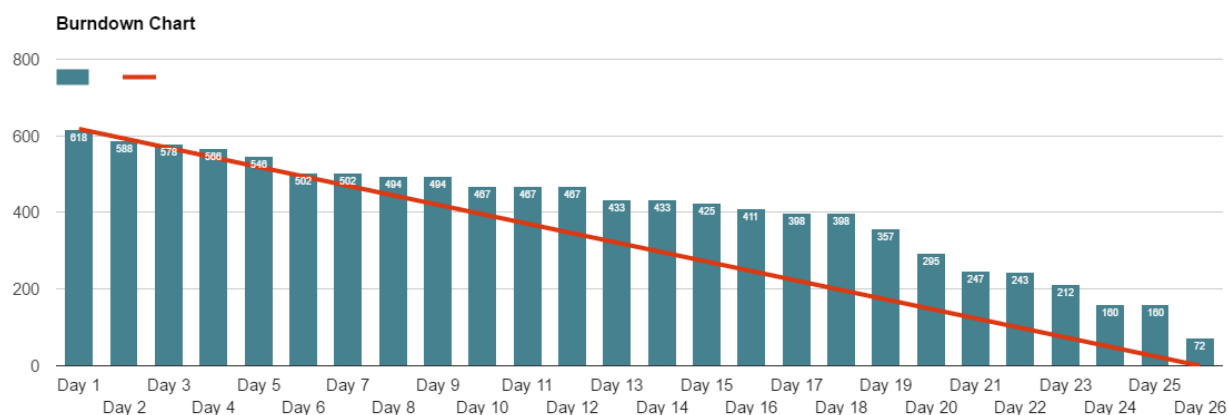
Tabell 1 Dokumenthistorie

Versjon 0.1	20.02.16	Mal av Thomas.
Versjon 0.9	09.05.16	Flytte over alle felt fra Sprint 4
Versjon 1.0	10.05.16	Innledning og evalueringstekst Thomas med input fra alle. Espen: Korrektur.
Versjon 1.1	19.05.16	Thomas opprettet versjon 1.1
Versjon 2.0	20.05.16	Carl Martin korrektur.

1 Innledning

I etterkant av evalueringen har hvert enkelt gruppe medlem hatt en medarbeidersamtale med gruppeleder, og det er lagt en realistisk plan for hvordan vi skal jobbe med dokumentasjon og hvordan vi skal forberede siste presentasjon. Også denne gangen har vi foretatt et effektivt sprint retrospektiv. Der evalueringer fra hver enkelt utgjør en del av dette dokumentet.

Kapitlene *Data*, *Elektro*, *Maskin*, *Samarbeid*, *Kommunikasjon* og *Administrasjon* inneholder svar direkte fra en spørreundersøkelse. Vi har igjen valgt å beholde tabellene i «review» – dokumentene, så det ikke blir så mye frem og tilbake mellom rapport og referansetabell.



Figur 1: Burndown chart

2 Data

I Sprint 4 har vi utarbeidet software til testmaskinen og programmert på UR 5 robotarm. Alt ligger klart for nettverkskommunikasjon mellom de to enhetene og kjøring av sekvenser på armen. Loggsystemet skriver rett til en loggfil, denne kan enkelt lagres og arkiveres. Brukeren har tilgang til og redigere testsekvenser i programmet så disse kan lagres som XML filer, for senere bruk. Systemet kobler seg til og mottar data fra en sensor som skal monteres på våpenstasjonen. Kommunikasjon fungerer trådløst. Har brukt en del tid på «event-listeners» og «on property changed» konfigurasjon for lese og skrive funksjonen til loggfilen. Vi har virkelig hatt en produktiv sprint, og er ganske fornøyd med resultatet.

3 Elektro

Arbeidet med Sprint 4 har vært lærerikt og moro. Endelig har vi fått fokusert mer på teknisk enn på dokumentasjon og planlegging. Fremgangen har vært fantastisk, på tross av litt utfordringer i starten av Sprinten. Igjen fikk vi merke avhengighetene i prosjektet som resulterte i litt venting, mens ting ble gjort ferdig, før andre fagretninger fikk sluppet til med sine egne ting. Når delene var tilgjengelig ble de 2 siste ukene ganske hektisk med montering og tilpassing, samt utvikling av kabler slik at EE skulle bli klart for test, dette ble vellykket og EE er klar for eventuelle sekvenser.

4 Maskin

I denne sprinten har vi tatt frem vår fungerende prototype, og mye tid har gått med til CAD og print. Parallelt med testing av system og subsystem, har vi generert mye innhold til dokumentasjonen.

Mange små «selvfølgeligheter» i form av ulike mål, standarder og små tilpasninger tar endel tid. Opplever at slike detaljer rundt montering og tilpasning av ulike deler burde vært bedre planlagt, forsøk på rask klargjøring slik at testing av signaler og bevegelser for data/elektro kan starte, kan føre til mangler. Opplever også (ikke uventet) våre ulike fokus på bakgrunn av fagretning, og hvor lærerikt dette er for å se systemet fra ulike innfallsvinkler.

5 Samarbeid

Samarbeidet fungerer bra, noe frem og tilbake rundt montering og hva som haster mest. Opplever bra håndtering av ting som oppstår underveis.

I sprint 4 har det vært en hektisk test og monterings periode, vi har fått vært mye sammen, og jeg synes det har vært bra, slik at vi er flere fysisk på en sak.

Samarbeid i gruppen denne sprinten har fungert godt.

Samarbeidet fungerer for det meste greit, vi har denne sprinten fått testet hvordan avhengighetene spiller en stor rolle på et slikt prosjekt.

Samarbeidet har fungert fint, vi jobber bra sammen og innspurten av sprint 4 har virkelig testet hvordan vi har samarbeidet, de forskjellige subsystemene har blitt sammenstilt.

Siste tiden nå er hektisk og det er viktig å beholde et godt fokus for å forsikre om oss at vi får tatt med oss det gode samarbeidet vi har hatt helt frem til sommeren, synes det har gått greit.

6 Kommunikasjon

Bra opprettholdelse av kommunikasjon for fokus mot felles mål.

Etter eksamensperioden har vi fått mer tid sammen med bacheloroppgava, kommunikasjonen har vært god slik at man får et bedre samspill om oppgaven.

Kommunikasjonen innad i gruppen har fungert fint i denne perioden. Vi har for det meste vært samlet i ukedagene og da har det vært lett å holde kommunikasjonen i gang. Når noen har vært andre steder og jobbet har vi fortsatt å bruke Messenger til å formidle informasjon og passe på at alle beskjeder har nådd frem. Det virker som at alle nå har vendt seg til å bruke disse verktøyene, og kommunikasjonen her fungerer dermed mye bedre enn tidligere.

Kommunikasjonen i gruppa denne sprinten har stort sett vært god, men det er fortsatt rom for forbedringer her.

Vi har hatt en stort sett tilfredsstillende kommunikasjon.

7 Administrasjon

God informasjon og samling av tråder. Bra styrking av fellesskap med en sosial dag for team building

Administrasjonen fungerer utmerket i gruppa, ingen ting å utsette på den.

Administrasjonen har vært bra i denne perioden.

Administrasjonen har fungert helt greit, vi valgte å korte ned på møter vi ikke så noen direkte verdi av der og da, og kunne derfor fokusere mer på sprinten.

De administrative oppgavene har vært effektivt løst i denne sprinten.

Administrasjonen fungerer bra.

8 User Story gjennomgang

Etter gjennomført sprint sitter vi nå igjen med 3 kategorier av de user storiene vi har hatt med.

- De som er utført har grå bakgrunn.
- Der det er behov for gjentakelse senere er merket med grønn bakgrunn.
- De som ikke ble gjennomført er merket med rød.

Vi oppretter «issues» der vi oppdager at vi må avvike fra den opprinnelige planen. Det kan være med hensyn på tidsbruk, problemer som oppstår, eller at noe ikke blir gjennomført. «Issues» blir ført inn i et regneark med korresponderende nummer. Vi har listet dem sammen med tilhørende user story i tabeller med gul farge.

[US-010701]	5	64:00
Som oppdragsgiver ønsker jeg at det skal vises prototyper slik at vi kan se hva gruppen har tenkt, og for tidlig og fortløpende V&V.		
AC-010701-03: Første del av EE design printes		
AC-010701-04: Montering og integrering av servomotorer sammen med printet EE.		
AC-010701-05: Dokumentering av AC-010701-03/04		
AC-010701-06: Test av system sammen (EE og servomotorer)		
AC-010701-07: Dokumentasjon av AC-010701-04.		

Vi har tatt 29 timer fra dokumentasjon i denne fasen for å ferdigstille prototypen.

Brukt en dag (8t) på modifisering og generell klargjøring av en finger til print (Aina).

Klargjøring og forberedelse til print (4 t).

Videre brukt 14 timer på modifisering, små endringer, sammenstilling og print (Aina).

Siste uke brukt 9 timer til på prototype (Aina)

AC-010701-01: Samarbeid print, og henting, 4t.(Monica)
 Kommunikasjon, samarbeid 16,5t (Carl M)
 AC-010701-02: Plater og div til festing av finger og komponentrom. 1t(Monica)
 Modifisering av første print, 19t (Carl M)
 AC-010701-03: Ny sammenstilling i Solid Works. For bruk i dok.6t(Monica).
 Innhold dokument generert av sammenstilling brukt for endelig print (Aina)
 AC-010701-04: Monica: 6t bistand montering + skruer, deler, kabinett.
 AC-010701-05: Tatt bilder og filmet (Aina)

[US-020102]	5	6:00
Som intern veileder ønsker jeg at prosjektgruppa sender ukentlig skriftlig rapport slik at jeg kan holde oversikt over fremgang.		
AC-020102-01: Skriftlig rapport er produsert.		
AC-020102-02 : Gruppa har godkjent rapporten.		
AC-020102-03 : Rapporten er sendt til intern veileder.		
AC-020102-04 : Referat er skrevet og sendt intern veileder.		

Uke 1: Oppfølgingsdokument 11 er laget, gruppen har fylt inn og godkjent, Sendt 12.04.16.
 Uke 2: Oppfølgingsdokument 12 er laget og sendt. (Referat veiledermøte 0,5 t på denne)
 Uke 3: Oppfølgingsdokument 13 laget og sendt.
 Uke 4: Oppfølgingsdokument 14 er laget og sendt

[US-020103]	5	6:00
Som oppdragsgiver ønsker jeg at prosjektgruppa sender skriftlig status rapport hver 14.dag slik at vi blir holdt oppdatert.		
AC-020103-01: Skriftlig rapport er produsert.		
AC-020103-02 : Gruppa har godkjent rapporten.		
AC-020103-03 : Rapporten er sendt til ekstern veileder.		

Statusrapp 1 sprint 4: innhold risiko og vedlegg er laget (1 time - Aina)
 Statusrapp 2 Sprint 4: innhold laget, tabell legges ved (1 time – Aina)
 AC-020103-01: Skriftlig rapport er produsert.
 AC-020103-02: Gruppa har godkjent rapporten.
 AC-020103-03: Rapporten er sendt til ekstern veileder.

[US-070101]	5	6:00
Som prosjektgruppe ønsker vi god kommunikasjon med oppdragsgiver slik at vi kan sikre et godt samarbeide.		

AC-070101-01: Oppdragsgiver har fått statusrapport til rett tid.
AC-070101-02: Alle Gruppemedlemmer har møtt til avtalt tid til møter.

AC-070101-01: Oppdragsgiver har fått statusrapport til rett tid.
AC-070101-02: Alle Gruppemedlemmer har møtt til avtalt tid til møter.

[US-070401]	4	1:00
Som Scrum master ønsker jeg at gruppa har et budsjett som oppdateres fortløpende slik at vi har god økonomisk kontroll		
AC-070401-01: Det er gått en ny runde med alle fagretninger der de har hatt mulighet til å komme med input.		
AC-070401-02: Informasjonen er bearbeidet og lagt inn i budsjettdokumentet.		
AC-070401-03: Informasjonen er oppdatert på alle relevante steder.		

[US-070501]	4	1:00
Som Scrum master ønsker jeg at gruppa har et timebudsjett som oppdateres fortløpende slik at vi har oversikt over hvem som har brukt timer, og på hva		
AC-070501-01: Alle har oppdatert timelistene sine.		

[US-070607]	3	1:00
Som bachelorgruppe ønsker jeg at blogg oppdateres regelmessig slik at vi lett kan informere våre interessenter		
AC-070607-01: Oppdateringer er lagt ut på blogg.		

En blogg post lagt ut. Skal legge ut en til med oppdatering på at vi har mottatt servoer og konstruksjon er underveis.

[US-070702]	5	2:00
Som intern sensor krever jeg at project burndown skal fremstilles på en måte, så oppdragsgiver kan få muligheten til å se hvilke User Stories vi ikke rekker å være med på å prioritere userstories slik at vi har en felles oppfatning av målet, og at de kan hjelpe oss å prioritere de som er viktigst.		
AC-070702-01: Godkjent av gruppen		
AC-070702-02: Godkjent av oppdragsgiver		

Vi lagde en ny i forrige planleggingsfase. Denne ble sendt som vedlegg til forrige status rapport. Og det kom ingen innvendinger. I all hovedsak gjorde vi det for å lære noe og bruke de innspill vi fikk fra oppdragsgiver under forrige møte. Dette ble mer ryddig og vi har fått gjort litt mer.

[US-070801]	5	8:00
Som Product Owner ønsker jeg at User Stories dokumentet holdes ved like slik at dokumentet alltid vil være oppdatert		
AC-070801-01: Dokumentet er oppdater med de siste endringene før Sprints slutt		

[US-070901]	5	32:00
Som oppdragsgiver ønsker jeg at riskdokumentet oppdateres fortløpende og brukes underveis i prosjektet slik at RISK hele tiden er i fokus for prosjektgruppen		
AC-070901-01: Risk ansvarlig har gått runden og samlet inn ny info om risk fra alle fagansvarlige		
AC-070901-02: Riskdokument er oppdatert		
AC-070901-03: Riskdokument er gjennomgått og godkjent i plenum		

1. oppdatering sprint 4: 13.04: Oppdatering/ klargjøring dokument (4t Aina), Muntlig oppfordring i gruppa om å forberede risiko for gjennomgang. 14.04: Gjennomgang i plenum (5t total). Oppdatering hoveddokument og arbeidsverktøy (6t Aina).
2. oppdatering sprint 4: 28.04: Klargjøring 1 time, Gjennomgang i plenum 5 timer, oppdatering og vedlikehold (3 timer).

[US-030101]	5	6:00
Som oppdragsgiver ønsker jeg at samtlige gruppemedlemmer setter seg inn i konsernets sikkerhetsrutiner slik at alle gruppemedlemmene har dette klart for seg, og ikke bryter disse.		
AC-030101-01: Alle gruppens medlemmer har vært igjennom sikkerhets brif.		
AC-030101-02: Alle gruppens medlemmer har fylt ut papirene som kreves for å få tilgang til parken/bygningen.		

Denne har blitt utført i sprint 1. Vi har valgt og ta den med nå for å dokumentere ferdigstilling.

[US-021507]	5	72:00
Som oppdragsgiver ønsker jeg at end effector teknologidokument holdes ved like Slik at det er mest mulig "up to date"		
Denne US-en vil bli nøyere dokumentert i sprint 5. Gjennom sprint 4 vil det bare genereres dokumentasjon.		
AC-021507-01: EE teknologidokument generere innhold		
AC-021507-02: Teknologidokument for Design		
AC-021507-03: Teknologidokument for Mekanikk		
AC-021507-04: Teknologidokument for Elektronikk		

Det som ikke er blitt gjort på dokumentasjon går videre til neste sprint.

AC-021507-01: Ny sammenstilling for rett generering i Solid Works. Forts. tektdok mek til EE.

AC-021507-02/03: Ting fra sprint 4 er med og dokumentasjonen vokser litt og litt.

AC-021507-04: Teknologi dokument for elektronikk blir generert. Carl M (22t + 2t)

Her har vi feilestimert tid, 29 timer av timene er isteden brukt i US-010701 til prototypen.

[US-021002]	5	32:00
Som høyskole ønsker jeg at testverifikasjon dokumentet vedlikeholdes Slik at det er mest mulig "up to date"		
AC-021002-01: Komponenter og grensesnitt er styrkeberegnet/analysert der dette er naturlig.		
AC-021002-02 : Komponenter og grensesnitt er fysisk testet.		
AC-021002-03 : Testresultater er dokumentert og samlet for innskrivning i testverifikasjonsdokument.		

Aina: kjørt FEM, beregnet lineære krefter på innfestning motor. Startet dokumentsamling til testverifikasjonsdokumentet. Bård: Lagt til en test. Carl M: Har Bilder av test. Espen: Fylt inn noen datatester. Dokumentet er under utarbeidelse, og har til nå blitt klargjort for innfylling i dokumentasjonssprinten. Tabeller er klargjort, US og AC er satt inn, samt noe tekst er satt inn der dette har vært naturlig. Føringer for innfylling i plenum vil bli gitt, samt tilknytning til risiko vil settes inn. (Aina 14 timer)

[US-02160101]	4	18:00
Som bachelorgruppe ønsker vi at WS-sensors oppbygging dokumenteres med kretstegninger slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.		
AC-021604-01: Det er produsert korrekte skjemaer over kretsene.		
AC-021604-02: Skjemaer er lagret på hensiktsmessig sted.		

AC-021604-01: Det er produsert korrekte skjemaer over kretsene, de er simulert og lagret, samt lagt til i testverifikasjons dokumentet.

AC-021604-02: Skjemaer er lagret på hensiktsmessig sted for videre bruk i dokumentasjonen.

[US-02160201]	3	2:00
Som bachelorgruppe ønsker vi at BOM lages for WS-sensoren slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.		
AC-021605-01: BOM er produsert og lagt i teknologidokument		

BOM grunnlag er funnet, men ikke lagt til i dokumentasjonen ennå, videreført til neste sprint

[US-02160301]	2	4:00
Som bachelorgruppe ønsker vi at dokumentasjonen skal inneholde data ark for alle 3.parts komponenter brukt på WS-sensor slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.		
AC-021606-01: Data-ark er samlet så langt det lar seg gjøres å lagt i egen mappe		

Utsatt, til dokumentasjons fasen, med lav prioritet.

[US-046101]	5	1:00
Som oppdragsgiver ønsker jeg at sensor skal registrere alle avfyringer slik at alle avfyringer som skjer registreres		
AC-046101-01: US-046102 er gjort.		
AC-046101-02: US-046103 er gjort.		
AC-046101-03: US-046104 er gjort.		

Ferdig med unntak av at US-046103 ikke kan fullføres på grund av manglende mulighet til praktisk testing.

AC-043101-01: Statemaskinen utgår, det blir en litt annen løsning.
AC-043101-02: Arduino tar imot instrukser fra UR5 (test) (Dette fungerer nå, ikke ideelt, men greit nok, Bård)
AC-043101-03: Dokumentasjon av AC-043101-01/02.(Ikke gjort ennå, gjøres i neste Sprint

[US-046102]	5	4:00
Som ingeniør ønsker jeg at tilkoblingsmulighetene på WS for sensor kartlegges slik at sensor kan designes til å passe overens med denne utgangen.		
AC-046102-01: Møte gjennomføres/informasjon skaffes		
AC-046102-02: Informasjonen er lagret på hensiktsmessig sted uten sporbarhet tilbake til WS		

Informasjonen er hentet inn og benyttet i design. Lagret uten sporbarhet.

[US-046103]	5	30:00
Som ingeniør ønsker jeg at det designes en krets som tolker avfyrings signalet, og oversetter dette til en 5V puls slik at signalet kan mottas av Arduino.		
AC-046103-01: Ideen til krets er notert ned		
AC-046103-02: Konseptet er simulert i f.eks. OrCad for å teste om det er mulig		
AC-046103-03: Konseptet er testet i praksis		

Lagt til å testverifikasjonsdokument, under T-E6., Praktisk test gjennomført, feilet dessverre da vi ikke var klar over at det må oppgis en Våpen ID for at WS kan gjøre en avfyring. Dette utredes videre.

[US-046201]	5	2:00
Som oppdragsgiver ønsker jeg at sensor skal kommunisere med Software på testmaskin slik at alle hendelser som registreres kan rapporteres tilbake til test Software.		
AC-046201-01: Sensor kommunikasjon er satt opp og denne er testet		

AC-046201-01: Sensor kommunikasjon er satt opp, Avklart med SW kommunikasjonsprotokoll, og dette er testet og i orden

[US-046301]	4	2:00
Som oppdragsgiver ønsker jeg at kommunikasjon mellom sensor og Software på testmaskin fungerer trådløst slik at vi slipper kabler mellom stasjonær testmaskin og bevegelig WS.		
AC-046301-01: Sensor kommunikasjon er satt opp og denne er testet		

AC-046301-01: Sensor kommunikasjon er satt opp, Avklart med SW kommunikasjonsprotokoll, og dette er testet og i orden

[US-021707]		20:00
Som oppdragsgiver ønsker jeg at All egenutviklet Software er dokumentert grundig slik at vi kan forstå hva dere har gjort		
AC-021707-01: Software designdokument er oppdatert		
AC-021707-02: Software teknologidokument er utarbeidet.		

Opprettet disposisjon og notert noen hovedlinjer i dokumentet. Software har planlagt for dokumentasjon i sprint 5.

[US-040201]	5	0:00
Som oppdragsgiver ønsker jeg at Software skal kunne kommunisere med WS slik at signaler kan sendes og mottas mellom disse to.		
AC-060801-01: Software kan ta imot informasjon fra WS og lagre dette på en hensiktsmessig måte		

Grunnet begrenset tid med tilgang til våpenstasjonene og robotarmen har denne blitt droppet. Vi har prioritert robotarmen.

[US-040301]	5	20:00
Som oppdragsgiver ønsker jeg at Software skal kunne kjøre bevegelsessekvenser på UR 5 robotarm slik at Software kan kontrollere arm.		
AC-040301-01: Software kan sende innlastet XML fil med instruksjoner til ur 5 robotarm og de blir forstått og fulgt.		
AC-040301-02: Det er lagt til funksjonalitet i programmet slik at brukeren kan lage egne testsekvenser utfra de mulige funksjonene på DCP og CG		
AC-040301-03: Det er opprettet en funksjon for hver bevegelsessekvens som er mulig på UR 5		
AC-040301-04:		

Skjelettet til programmet som lar brukeren velge testsekvenser og lagre som XML er utarbeidet. Det mangler og lage funksjoner for hver bevegelse. Det mangler også å legge til de faktiske navnene på sekvensene som skal kjøres.

[US-040401]	5	16:00
Som oppdragsgiver ønsker jeg at UR5 må kunne kommunisere med Arduino som styrer hånd slik at Software kan kontrollere hånd.		
AC-040601-01: Det er satt opp ett statemachine diagram er utarbeidet og overlevert elektro		

Elektro har valgt å gå for en annen teknisk løsning. Tiden avsatt til denne oppgaven har derfor blitt brukt til forbedring av trådfunksjonalitet i koden.

[US-040601]	5	10:00
Som oppdragsgiver ønsker jeg at Software skal kunne ta imot fyringssignal slik at Software kan verifisere fyring.		
AC-040601-01: Software tar imot en variabel som bestemmer om fyring skjer eller ikke		
AC-040601-02: Software kan vise om fyring skjer i GUI		

[US-040701]	5	16:00
Som oppdragsgiver ønsker jeg at Software skal kunne lese testinstruksjoner fra fil slik at Software kan utføre testsekvenser		
AC-040701-01: Software leser inn en XML fil og lagrer den i et string array		

Software kan lese testinstruksjoner fra fil, denne blir ikke lagret i et array men, filnavnet på den åpne filen blir lagret slik at den kan brukes andre steder.

[US-042001]	5	12:00
-------------	---	-------

Som oppdragsgiver ønsker jeg at det opprettes et loggsystem som skal registrere hendelser slik at det kan lese ut av loggen hva systemet har gjort.

AC-042001-01: Loggsystem er opprettet og testet

Loggsystemet er klart. Det er laget for å håndtere skrijving fra flere tråder og avlesing. Loggen kan eksporteres til fil. (Manger testing med sensor men systemet er ferdig.)

[US-042002]	5	4:00
Som oppdragsgiver ønsker jeg at loggsystemet skal notere instruksjon i loggen slik at det kan leses ut av loggen hvilke instruksjoner Software gir til systemet.		
AC-042002-01: Det er satt opp en funksjon som skriver til fil		

AC-042002-01: Det er satt opp en funksjon som skriver til fil

[US-042003]	5	4:00
Som oppdragsgiver ønsker jeg at loggsystemet skal verifisere bevegelse ved UR5 arm slik at det kan leses ut av loggen at det systemet ble gitt instruksjoner om, faktisk ble gjort		
AC-042003-01:		
AC-042003-02:		

Det er lagt til skrivinger til logg i løkken som kjører tester i test Software. Denne fungerer ved at det kommer en OK streng tilbake fra roboten når den valgte bevegelsen er utført.

[US-042004]	5	4:00
Som oppdragsgiver ønsker jeg at loggsystemet skal verifisere bevegelse ved EE slik at det kan leses ut av loggen at det systemet ble gitt instruksjoner om, faktisk ble gjort		
AC-042004-01:		
AC-042004-02:		

[US-042006]	3	0:00
Som oppdragsgiver ønsker jeg at loggsystemet skal sammenligne forventet resultat med faktisk resultat slik at systemet automatisk kan kontrollere om testen er godkjent		
AC-042006-01:		
AC-042006-02:		

Da vi ikke henter data fra våpen stasjonen blir testverifikasjon i systemet begrenset.

[US-042007]	2	4:00
Som oppdragsgiver ønsker jeg at loggsystemet skal markere uventet utfall i loggen slik at tester med feil er lett avdekke.		
AC-042007-01: AC-042007-02:		

Da vi ikke henter data fra våpenstasjonen blir testverifikasjon i systemet begrenset.

[US-045101]	5	16:00
Som oppdragsgiver ønsker jeg at UR5 skal kommunisere tilbake til Software på testmaskin slik at test Software kan lagre bekreftelse på at instruksjoner er utført i loggen.		
AC-045101-01: Det er mulig for Software og ta imot informasjon fra UR5 AC-045101-02: Informasjon fra UR5 lagres i Software. AC-045101-03 : Det er tilrettelagt at dette kan sendes videre til logg		

Denne er mer eller mindre ferdig. Når en bevegelse er utført vil det sendes en streng til Software som sier "OK" Denne gjør at løkken i Software går videre til neste steg og da kan det skrives til logg at bevegelsen er ferdig. Er ikke fullstendig implementert men alle forutsetninger ligger der.

[US-045201]	5	20:00
Som oppdragsgiver ønsker jeg at UR5 skal kunne sette opp en socket listener loop slik at UR5 kan lytte etter tilkobling fra test Software.		
AC-045201-01: Socket listener loop med bevegelsessekvenser som inkluderer gripeverktøy er opprettet		

Det er satt opp en listener loop på UR 5 som lytter etter tilkobling. Det er også satt opp noen testsekvenser som det fungerer og kalle fra Software slik at denne kommunikasjonen er testet. Det står igjen og lage sekvenser som også har med gripeverktøyet.

[US-060201]	5	8:00
Som oppdragsgiver ønsker jeg at testmaskinen skal kunne motta informasjon fra UR5 slik at informasjon som er viktig for loggen kommer frem.		
[AC-060201] Testmaskinen kan motta strenger fra UR 5 og lagre de om nødvendig		

Det er satt opp en løkke på UR 5 som sender strenger med nødvendig informasjon på gitte punkter. Disse mottas og skrives til loggen i Software. Noe står igjen her, blant annet å skrive om på noen av strengene og hvor de skal i Software.

[US-060301]	5	16:00
Som oppdragsgiver ønsker jeg at UR5 skal kunne gi instruksjoner til EE slik at vi har en sentralisert styring av systemet.		
[AC-060301] Instruksjoner til EE blir lagret i en bevegelsessekvens på UR 5 og blir kjørt og utført korrekt.		

Dette må vi gjøre i sprint 5 og 6, venter på ferdigstilling av prototypen.

[US-060401]	5	0:00
Som ingeniør ønsker jeg at det tilrettelegges for å lagre informasjon fra EE slik at informasjon som er viktig for loggen kommer frem.		
AC-060401-01: Det er opprettet en funksjon som lagrer informasjon fra EE som et string array		
AC-060401-02: Funksjonen er testet og det er dokumentert i testverifikasjons dokumentet		

Insteadet for å lagre strings i et array så kjører vi all informasjon fra testene direkte til loggfil.

[US-060501]	5	16:00
Som oppdragsgiver ønsker jeg at testmaskinen skal kunne gi instruksjoner til sensorer slik at vi har en sentralisert styring av systemet.		
AC-060501-01: Det er opprettet en funksjon som kan sende informasjon til sensor		
AC-060501-02: Funksjonen er testet og det er dokumentert i testverifikasjons dokumentet		

Har ikke fått det inn i testverifikasjonsdokumentet. Tar det i neste sprint.

[US-060701]	5	0:00
Som oppdragsgiver ønsker jeg at testmaskinen skal kunne gi instruksjoner til WS slik at testsekvenser som krever dette kan gjennomføres.		
AC-060701-01: Software kan sende informasjon til våpenstasjon og få et svar som viser at dette er testet.		

Grunnet begrenset tid med tilgang til våpenstasjonene og robotarmen har denne blitt droppet. Vi har prioritert robotarmen.

[US-060801]	5	0:00
Som oppdragsgiver ønsker jeg at testmaskinen skal kunne motta informasjon fra WS slik at testsekvenser som krever dette kan gjennomføres.		

AC-060801-01: Software kan ta imot informasjon fra WS og lagre dette på en hensiktsmessig måte

Grunnet begrenset tid med tilgang til våpenstasjonene og robotarmen har denne blitt droppet. Vi har prioritert robotarmen.

[US-043001]	5	8:00
Som oppdragsgiver ønsker jeg at EE skal kunne festes på UR5 slik at de to kan brukes sammen.		
AC-043001-01: Interface er designet		
AC-043001-02: Interface er godkjent av PO		
AC-043001-04: Interface er printet		
AC-043001-03: Interface er fysisk testet koblet på UR5		

Ting er litt modifisert, men med en hensikt i å minske mellomledd som kan være kilde til svakheter.

[US-043002]	5	8:00
Som ingeniør ønsker jeg at det fysiske grensesnitt UR5/EE skal bestå av to deler før kobling mot EE slik at EE kan monteres på/av på en hensiktsmessig måte		
AC-043002-01: Koblingen er printet		
AC-043002-02: Koblingen er fysisk testet		

[US-043101]	5	32:00
Som oppdragsgiver ønsker jeg at EE skal kunne motta instruksjoner fra Software via UR5 slik at den kan gjøre de bevegelsene test Software instruerer den om		
AC-043101-01: Statemaskinen er kodet		
AC-043101-02: Arduino tar imot instrukser fra UR5 (test)		
AC-043101-03: Dokumentasjon av AC-043101-01/02.		

Espen trenger et diagram over hvilke digitale signaler som skal sendes fra ur5

Bård: Kommunikasjon fungerer nå, men ikke helt ideelt. Litt siste testing gjenstår for å sjekke robustheten til systemet.

AC-043101-01: Statemaskinen er kodet

AC-043101-02: Arduino tar imot instrukser fra UR5 (test) (Under arbeid, Bård)

AC-043101-03: Dokumentasjon av AC-043101-01/02.

AC-043101-01: Statemaskinen utgår, det blir en litt annen løsning.

AC-043101-02: Arduino tar imot instrukser fra UR5 (test) (Dette fungerer, men ikke optimalt)

AC-043101-03: Dokumentasjon av AC-043101-01/02.(Ikke gjort ennå, gjøres i neste Sprint

Ferdig med unntak av at US-046103 ikke kan fullføres på grund av manglende mulighet til praktisk testing.

[US-044002]	5	32:00
Som oppdragsgiver ønsker jeg at EE skal kunne holde inne palm knapp slik at tester som inkluderer betjening av palm knapp kan automatiseres.		
AC-044002-01: Beregninger er gjort slik at det skal er nok styrke i utforming av EE		
AC-044002-02: Servomotorer er programmert til å utføre sekvensene for å aktivere palm knapp.		
AC-044002-03: Dokumentasjon av AC-044002-01/02.		
[US-044005]	5	32:00
Som oppdragsgiver ønsker jeg at EE skal kunne betjene triggerknapp slik at tester som inkluderer betjening av triggerknapp kan automatiseres.		
AC-044005-01: Beregninger er gjort slik at det skal er nok styrke i utforming av EE		
AC-044005-02: Servomotorer er programmert til å utføre sekvensene for å aktivere triggerknapp.		
AC-044005-03: Dokumentasjon av AC-044005-01/02.		
[US-044102]	5	16:00
Som oppdragsgiver ønsker jeg at EE skal kunne betjene de forskjellige vippebryterne på DCP slik at tester som inkluderer betjening av vippebrytere på DCP kan automatiseres.		
AC-044102-01: Beregninger er gjort slik at det skal er nok styrke i utforming av EE		
AC-044102-02: Programmering av EE skal være ferdig slik at Software kan programmere UR5 til å gjøre sekvens.		
AC-044102-03 : Software programmerer sekvens for bevegelse av UR5 slik at EE utfører test for betjening av vippebrytere til DCP		
AC-044102-04 : Dokumentasjon av AC044102-01/02/03		
[US-044103]	5	16:00
Som oppdragsgiver ønsker jeg at EE skal kunne trykke på alle knappene på DCP slik at tester som inkluderer betjening av alle knapper på DCP kan automatiseres.		
AC-044103-01: Beregninger er gjort slik at det skal er nok styrke i utforming av EE		
AC-044103-02: Programmering av EE skal være ferdig slik at Software kan programmere UR5 til å gjøre sekvens.		
AC-044103-03 : Software programmerer sekvens for bevegelse av UR5 slik at EE utfører test for betjening av Knapper til DCP		
AC-044103-04 : Dokumentasjon av AC044103-01/02/03		

Espen trenger et diagram over hvilke digitale signaler som skal sendes fra ur5

Bård: Kommunikasjon fungerer nå, men ikke helt ideelt. Litt siste testing gjenstår for å sjekke robustheten til systemet.

AC-043101-01: Statemaskinen er kodet

AC-043101-02: Arduino tar imot instruksjoner fra UR5 (test) (Under arbeid, Bård)

AC-043101-03: Dokumentasjon av AC-043101-01/02.

Bård: laget Konfigurasjons program til Arduino for å kartlegge motortilstander for gjennomføring av bevegelse. Laget kode for kontrollerkort.

Aina: AC - 044002-01 er gjort

Carl M. Testet og montert fast EE mot UR5 slik at test for palm samt DCP og trigger er klar. Palm-knapp aktiveres og klar for test via EE, posisjon notert. / Etter montering av kabinett trengs det å endre posisjon til servomotorer.

Monica: Vært med på oppmåling, vurderinger og testing for at det hele skal fungere mot knappene.

Bård: Laget konfigurasjonsprogram til Arduino for å kartlegge motortilstander for gjennomføring av bevegelse. Laget kode for kontrollerkort

Carl M: Laget kabler for EE slik at test kan gjennomføres på alle servomotorer, og testet at EE kan utføre trykk på trigger og palm knapp.

CarlM: EE trykker inn Triggerknapp testet, posisjon er skrevet ned./endring etter montering av nytt kabinett.4t / Etter montering av kabinett trengs det å endre posisjon til servomotorer.

Monica: Vært med på oppmåling, vurderinger og testing for at det hele skal fungere mot knappene.

Bård: Laget konfigurasjonsprogram til Arduino for å kartlegge motortilstander for gjennomføring av bevegelse. Laget kode for kontrollerkort

Carl M: Testet EE mot DCP. Trenger en endring på en av fingertuppene slik at den klarer å trykke inn knapper. /endring etter montering av nytt kabinett.4t / Etter montering av kabinett trengs det å endre posisjon til servomotorer.

Bård: Laget konfigurasjonsprogram til Arduino for å kartlegge motortilstander for gjennomføring av bevegelse. Laget kode for kontrollerkort

Carl M: Loddet og gjort klar strømkabel slik at UR5 kan bevege seg med EE mye friere. 1.5t /endring etter montering av nytt kabinett.4t / Etter montering av kabinett trengs det å endre posisjon til servomotorer.

På grunn av endring fra funksjonell posisjon til EE blir disse US'ene slått sammen til en US i sprint 5, slik at vi kan kartlegge nye posisjoner for å utføre sekvensene til EE mot CG ang DCP. Det vil bli laget en egen AC for hver av de gamle US'ene: som ACxx palm knapp, ACxx triggerknapp etc.

Sammenslått og ferdigstilt ny US i Project backlog som [US-010101]

-	5	0:00
Som End-Effector krever jeg at jeg har en egen kontroller for å kunne styre komponentene jeg er bygget av slik at jeg vet hva jeg skal gjøre, selv om UR5 har en begrenset mengde output signaler.		
AC-051201-01: Anskaffelse av Arduino ATmega 2560		
AC-051201-02: Test av Arduino ATmega 2560		

AC-051201-01: Arduino er mottatt og tatt i bruk,
AC-051201-01: Arduino er testet.

9 Konklusjon

Sprint 4 har vært den lengste og mest omfattende sprinten. Vi var ambisiøse under planleggingen og det har fått full uttelling. Vi har kommet nesten i mål med det meste og svært fornøyd med arbeidet som har blitt lagt ned de siste ukene. Det er en del vi ikke rekker, men i smidige prosjekter som er låst til tid og kost er det funksjonaliteten som må ofres. Vi har måttet utelate en del oppgaver knyttet til verifikasjon fra våpenstasjon og detaljert verifikasjon fra systemet. De siste dagene har samarbeid fungert godt og vi har fått montert hånden. Noen steder har det blitt jobbet en del overtid.

Software designdokument

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAKSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Software designdokument
VERSJON / DATO	2.0 / 11. Mai 2016
FORFATTER	Espen S. Kraglund og Thomas Wegener
SIDEANTALL	27



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



i. Abstrakt

Dette dokumentet beskriver behovet for, og hvordan vi skal utvikle Software for å styre et komplett testsystem. Testsystemet skal erstatte en manuell gjennomgang av kombinasjoner av funksjoner som iverksettes for å teste endringer gjort på CROWS systemet. Dette dokumentet skal vise hvordan systemet modelleres og hvordan implementering prioriteres.

ii. Innholdsfortegnelse

i. Abstrakt.....	2
ii. Innholdsfortegnelse	3
iii. Oversikt tabeller.....	4
iv. Oversikt figurer	5
v. Dokumenthistorie	5
vi. Forkortelser	6
1 Innledning	7
2 Systembeskrivelse	7
2.1 Robotdatamaskin.....	8
2.2 Robotarm.....	8
2.3 Gripeverktøyet.....	8
2.4 DCP.....	8
2.5 CG.....	8
2.6 MPU	8
2.7 Våpenstasjon.....	9
2.8 Fyringssensor.....	9
2.9 Testdatamaskin	9
3 User Stories Software	9
3.1 Designkrav	9
3.2 Avgrensninger	12
4 Use case diagram	16
4.1 Testprogramvare use case.....	16
4.1.1 Koble opp testsystemet	17
4.1.2 Start test	17
4.1.3 Velg en test	17
4.1.4 Avbryt test	17
4.1.5 Reset test	17
5 Rediger testsekvens.....	17

5.1 Legge til testsekvens	18
5.2 Redigere testsekvens	18
5.3 Slette testsekvens	18
6 Klassediagram	18
6.1 Main	20
6.2 View	20
6.3 Controller	20
6.4 Sensor	20
6.5 WeaponStation	20
6.6 Robot	20
6.7 Log	20
7 Sekvensdiagram Run test	21
7.1 CreateTest	22
8 Konklusjoner etter implementering	23
8.1 Klassediagram	23
8.2 Konklusjon	25
Kilder	26

iii. Oversikt tabeller

Tabell 1: Dokumenthistorie	5
Tabell 2: Forkortelser	6

iv. Oversikt figurer

Figur 1: Deploymentdiagram	8
Figur 2: Testprogramvare use case.....	16
Figur 3: Rediger testsekvens use case	18
Figur 4: Klassediagram	19
Figur 5: Run test sekvensdiagram.....	21
Figur 6: CreateTest sekvensdiagram	22
Figur 7: Klassediagram etter implementasjon	24

v. Dokumenthistorie

Tabell 1: Dokumenthistorie

v 0.1 22.02.16	Espen: Skrevet innledning og abstrakt, satt opp hovedpunkter
v 0.2 01.03.16	Espen: Lagt til use case diagrammer
v 0.3 10.03.16	Thomas: Deploymentdiagram, systemforklaring, forkortelser
v 0.4 12.03.16	Espen: Div. tilføyelser
v 0.5 14.03.16	Thomas: Rensket og kommet med flere forslag til innhold.
v 0.6 15.03.16	Espen & Thomas: Lagt til sekvensdiagrammer, klassediagram
v 1.0 16.03.16	Espen: Ferdigstilt
v 2.0 11.05.16	Espen: Skrevet konklusjon etter implementering.

vi. Forkortelser

Tabell 2: Forkortelser

Forkortelser	Forklaring
KPS	Kongsberg Protech Systems
HSN	Høgskolen i Sørøst-Norge
ACT	Automated CROWS Testing
ACTEE	Automated CROWS Testing End Effector
EE	End Effector, gripeverktøy
CG	Control Grip, styringsjoystick for våpenstasjon
DCP	Display Control Panel
CROWS	Common Remotely Operated Weapon Station
UR5	Universal Robots 5 robotarm
US	User Stories
MPU	Main Processing Unit
XML	eXtensible Markup Language

1 Innledning

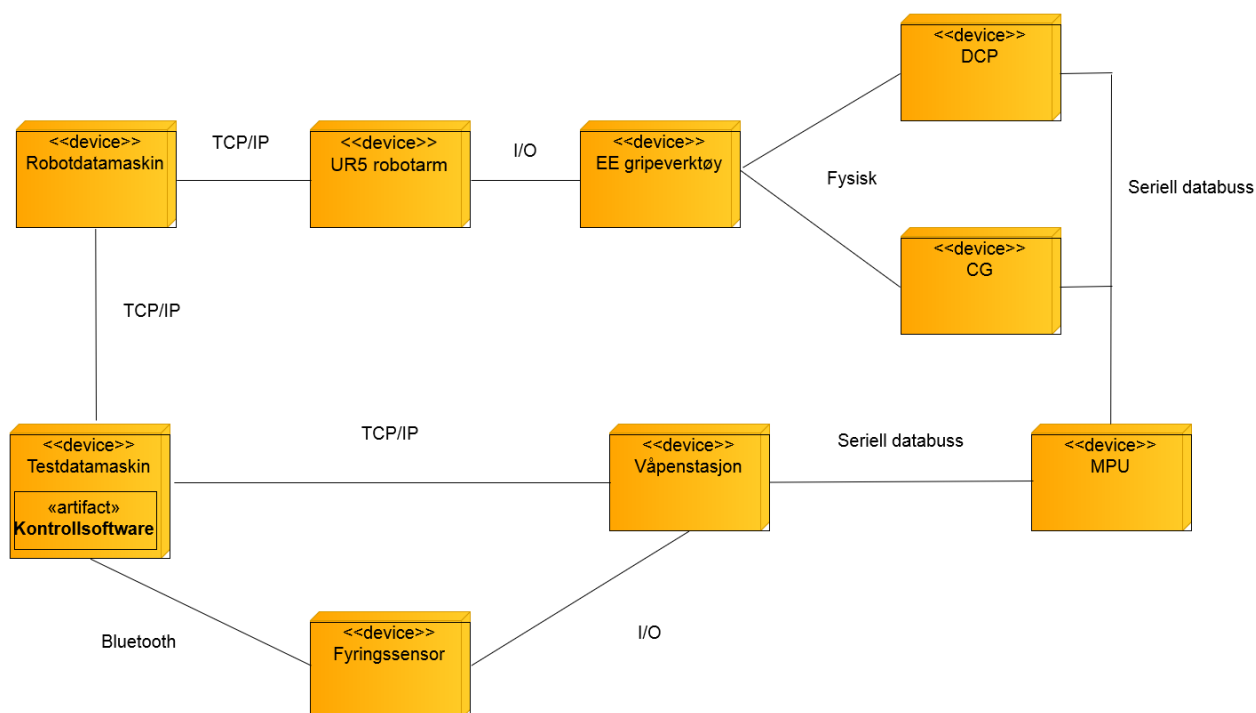
ACTEE er en bachelorgruppe som utfører en oppgave for Kongsberg Protech Systems. Vi skal automatisere testing av ny programvare på en CROWS M153 våpenstasjon.

Dette vil gjennomføres ved at vi lager ett gripeverktøy som skal plasseres på en UR5 robotarm, denne skal kunne utføre de funksjoner som er nødvendige for å kjøre testsekvenser automatisk.

Dette dokumentet inneholder en beskrivelse av Software design for et fullstendig testsystem, som kan utføre alle de nødvendige funksjonene for å automatisere testene.

Det første som blir forklart i dette dokumentet er systemet i sin helhet, og en beskrivelse av de enkelte enheter og kommunikasjonsveier. Vi vil så forklare hvordan Software skal designes, med hjelp av UML diagrammer. Til slutt er det et implementeringsperspektiv hvor vi tar for hvilken prioritet den forskjellige funksjonaliteten har.

2 Systembeskrivelse



Figur 1: Deploymentdiagram

Før vi kan begynne å forklare Software systemet som skal produseres er det nødvendig å forklare systemet som en helhet og hvordan det kommuniseres mellom de mange enhetene. For å lettere visualisere dette har vi lagd et deployment diagram.

Vårt prosjekt omhandler og ta det foregående prosjektet ACT[1] et skritt videre ved å designe et gripeverktøy og programvaren som er nødvendig for å styre denne.

Hensikten med dette er og kunne lage et fullstendig automatisert testsystem for ny Software på CROWS M153 våpenstasjon. Dette fullstendige testsystemet er det som er modellert ovenfor.

2.1 Robotdatamaskin

Dette er styringsenheten for UR 5 robotarmen [2] som kan betjenes ved bruk av en medfølgende touchskjerm. På denne touchskjermen kan bevegelsessekvenser programmeres ved å sette opp en serie med waypoints. Dette gjøres enkelt ved å bevege armen til den ønskede posisjonen. Disse sekvensene vil så bli en del av en løkke som vil vente på et funksjonskall fra testdatamaskinen, da vil den aktuelle bevegelsessekvensen kjøres. Denne maskinen kommer til å kommunisere med robotarmen og testmaskin med TCP/IP.

2.2 Robotarm

Denne kommer til å kommunisere med robotdatamaskinen via TCP/IP og med EE via I/O.

2.3 Gripeverktøyet

Gripeverktøyet er det fysiske produktet vi skal fremstille. Denne skal kunne betjene de ønskede funksjoner på DCP og CG. Bevegelsene som gripeverktøyet skal utføre vil programmeres på robotdatamaskinen og vår test Software vil ha funksjoner som kaller disse bevegelsene avhengig av hvilken test som skal kjøres.

2.4 DCP

Dette er kontrollpanelet som kontrollerer noen av funksjonene til våpenstasjonen. Her er det knapper og brytere som gripeverktøyet må kunne betjene.

2.5 CG

Dette er en joystick som styrer bevegelsene til våpenstasjonen. Det er også flere knapper på den som må kunne betjenes av gripeverktøyet.

2.6 MPU

Kontrollenhet for våpenstasjon.

2.7 Våpenstasjon

En CROWS M153 våpenstasjon. Det vil kommuniseres med denne med telnet. Kommunikasjon med denne er nødvendig for verifikasjon av tester.

2.8 Fyringssensor

Dette er en sensor utviklet av oss, som sjekker signalet fra våpenstasjonen og kommuniserer tilbake til testdatamaskinen om firing skjer. Dette er for å verifisere tester hvor firing er en del av testen.

2.9 Testdatamaskin

Her vil testprogramvaren kjøres fra, denne maskinen må ha mulighet for å sette opp tilkobling til robotarmen[3] for å kunne kalle bevegelsessekvensene som er programmert der. Brukeren skal ha mulighet til å lage egne testsekvenser fra et utvalg av bevegelser.

3 User Stories Software

3.1 Designkrav

Nedenfor følger en liste over Software relaterte User Stories. De er inndelt i 5 kategorier. Overordnede, Brukermanual, Dokumentasjon, Testprogram Software og Interface. De grå og blå er utførte og løpende prosesser.

3.1.1 Overordnede

US-010301		Score: 9.00
Som oppdragsgiver ønsker jeg at systemet skal kunne betjene CROWS M153 hardware, slik at testing av disse kan håndteres mest mulig automatisk.		
US-010401		Score: 9.00
Som oppdragsgiver ønsker jeg at systemet skal designes med mulighet for fremtidig utvidelse, slik at vi kan gjøre flere endringer senere ved behov.		
US-010501		Score: 9.00
Som oppdragsgiver ønsker jeg at systemet skal tilrettelegge for implementering av lavere prioriterte User Stories, slik at vi kan gjøre flere endringer senere ved behov.		
US-021801		Score: 8.33
Som oppdragsgiver ønsker jeg at dokumentasjonen skal inkludere en brukermanual, slik at dagens eller eventuelt nye operatører lett skal kunne forstå bruken av produktet.		

3.1.2 Brukermanual

US-021806		Score: 9.67
Som testoperatør ønsker jeg at brukermanualen skal beskrive alle forutsetninger som kreves for å kjøre en test, slik at disse er lett tilgjengelig for testoperatør, så man ikke støter på problemer underveis i testen.		
US-021803		Score: 8.67
Som oppdragsgiver ønsker jeg at brukermanual skal beskrive hvordan man legger inn nye sekvenser, slik at testoperatør kan gjøre dette, uten at noen må lære han opp.		
US-021805		Score: 8.33
Som testoperatør ønsker jeg at brukermanualen skal beskrive bruk av Software, slik at man kan være mindre avhengig av opplæring for å kunne bruke testsystemet.		
US-021804		Score: 8.00
Som testoperatør ønsker jeg at brukermanual skal være på engelsk, slik at vi ikke trenger brukermanualer på flere språk.		

3.1.3 Dokumentasjon

US-021703		Score: 9.00
Som ingeniør ønsker jeg at dokumentasjonen skal inneholde klassediagrammer for operatørprogramvaren, slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.		
US-021701	UTFØRT	Score: 2.00
Som oppdragsgiver ønsker jeg at dokumentasjonen skal inneholde UML diagrammer for all egenutviklet Software, slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.		
US-021705	UTFØRT	Score: 2.00
Som oppdragsgiver ønsker jeg at det lages et Software teknologidokument, slik at oppdragsgiver har dette til bruk i videreføring av prosjektet		
US-021702	UTFØRT	Score: 1.00
Som ingeniør ønsker jeg at dokumentasjonen skal inneholde USE CASE for alle brukeroperasjoner, slik at vi kan bruke disse senere for eventuell videreutvikling/produksjon.		
US-021706	UTFØRT	Score: 1.00
Som bachelorgruppe ønsker vi at det lages en mal for Software designdokument med		

disposisjoner, slik at vi kan begynne på teknologidokumenter i sprint 2

3.1.4 Testprogram Software

US-040301		Score: 9.67
Som oppdragsgiver ønsker jeg at Software skal kunne kjøre bevegelsessekvenser på UR 5 robotarm, slik at Software kan kontrollere arm.		
US-021704		Score: 9.00
Som oppdragsgiver ønsker jeg at kildekode er strukturert og lettleselig, slik at eventuelle andre utviklere kan forstå og legge til funksjonalitet senere.		
US-060101		Score: 8.67
Som oppdragsgiver ønsker jeg at testmaskinen skal kunne gi instruksjoner til UR5, slik at vi har en sentralisert styring av systemet.		
US-010201		Score: 8.33
Som oppdragsgiver ønsker jeg at testsekvenser skal kunne defineres av en testoperatør, slik at han kan velge ut hvilke tester som skal kjøres.		
US-040701		Score: 7.67
Som oppdragsgiver ønsker jeg at Software skal kunne lese testinstruksjoner fra fil, slik at Software kan utføre testsekvenser		
US-060401		Score: 7.00
Som oppdragsgiver ønsker jeg at UR5 skal kunne motta informasjon fra EE, slik at informasjon som er viktig for loggen kommer frem.		
US-060501		Score: 7.00
Som oppdragsgiver ønsker jeg at testmaskinen skal kunne gi instruksjoner til Sensorer, slik at vi har en sentralisert styring av systemet.		

3.1.5 Interface

US-040401		Score: 9.67
Som oppdragsgiver ønsker jeg at UR5 må kunne kommunisere med Arduino som styrer hånd, Slik at Software kan kontrollere hånd.		
US-060301		Score: 9.67
Som oppdragsgiver ønsker jeg at UR5 skal kunne gi instruksjoner til EE, slik at vi har en sentralisert styring av systemet.		

US-045001		Score: 9.33
Som oppdragsgiver ønsker jeg at UR5 sin posisjon skal styres av Software på testmaskin, slik at test Software kan fungere som hjernen i systemet.		
US-045201		Score: 9.30
Som oppdragsgiver ønsker jeg at UR5 skal kunne sette opp socket listener loop, slik at UR5 kan lytte etter tilkobling fra test Software.		
US-043101		Score: 9.00
Som oppdragsgiver ønsker jeg at EE skal kunne motta instruksjoner fra Software via UR5, slik at den kan gjøre de bevegelsene test Software instruerer den om.		
US-040201		Score: 6.67
Som oppdragsgiver ønsker jeg at Software skal kunne kommunisere med WS, slik at signaler kan sendes og mottas mellom disse to.		
US-045101		Score: 6.33
Som oppdragsgiver ønsker jeg at UR5 skal kommunisere tilbake til Software på testmaskin, slik at test Software kan lagre bekreftelse på at instruksjoner er utført i loggen.		
US-040101	UTFØRT	Score: 1.00
Som oppdragsgiver ønsker jeg at Software skal kunne kommunisere med robotarm via TCP, slik at signaler kan sendes og mottas mellom disse to.		

3.2 Avgrensninger

Dette prosjektet skal ta det forrige prosjektet ACT et skritt videre for å automatisere tester for ny Software på en CROWS M153 våpenstasjon. Vi utvikler et gripeverktøy som skal kunne betjene CG og trykke på knapper på DCP. Vi kommer til å prioritere å kunne demonstrere at gripeverktøyet kan utføre de funksjonene på DCP og CG som er ønskelig fra KPS sin side. Vår høyeste prioritet er å lage et program som vil vise at EE kan trykke på de riktige knappene og styre joysticken.

Ideelt sett ville vi utført alle funksjonene beskrevet i dette dokumentet, men bacheloroppgaven er begrenset i tid og det ble tidlig i prosessen avklart med oppdragsgiver at vi skal fokusere på å ta frem et funksjonelt gripeverktøy. Interface det mot eksisterende hardware og utvikle en noe begrenset tilhørende Software. Vi kommer til å fokusere på å legge til bevegelser for et forenklet oppsett. Og tilrettelegge slik at man enkelt kan legge til flere bevegelser. Vi kommer til å hente verifikasjon fra en fyringssensor og skrive resultat fra en test av fyringsknappen til en loggfil.

Og lage et fullstendig testsystem under denne bacheloroppgaven hadde vært det optimale, men er urealistisk med tanke på tiden vi har til rådighet.

3.2.1 Loggsystemet

Vi begrenser oppgaven i forhold til utførelse av loggsystemet. Vi designer rammeverket men implementer ikke en fullstendig løsning, på grunn av tidsbegrensninger.

US-042002		Score: 5.67
Som oppdragsgiver ønsker jeg at loggsystemet skal notere instruksjon i loggen, slik at det kan leses ut av loggen hvilke instruksjoner Software gir til systemet.		
US-042007		Score: 5.67
Som oppdragsgiver ønsker jeg at loggsystemet skal markere uventet utfall i loggen, slik at tester med feil er lett å avdekke.		
US-042001		Score: 4.67
Som oppdragsgiver ønsker jeg at det genereres et loggsystem som skal registrere alle hendelser, slik at det kan leses ut av loggen hva systemet har gjort.		
US-042005		Score: 4.00
Som oppdragsgiver ønsker jeg at loggsystemet skal verifisere hendelse ved WS, slik at det kan leses ut av loggen at det systemet ble gitt instruksjoner om, faktisk ble gjort		
US-042006		Score: 3.33
Som oppdragsgiver ønsker jeg at loggsystemet skal sammenligne forventet resultat med resultat, slik at systemet automatisk kan kontrollere om testen er godkjent		
US-042003		Score: 3.00
Som oppdragsgiver ønsker jeg at loggsystemet skal verifisere bevegelse ved UR5 arm, slik at det kan leses ut av loggen at det systemet ble gitt instruksjoner om, faktisk ble gjort		
US-042004		Score: 3.00
Som oppdragsgiver ønsker jeg at loggsystemet skal verifisere bevegelse ved EE, slik at det kan leses ut av loggen at det systemet ble gitt instruksjoner om, faktisk ble gjort		

3.2.2 Våpen stasjonen

Vi begrenser oppgaven i forhold til å hente verifikasjonsverdier fra våpenstasjonen for å verifisere at en knapp har blitt trykket på. Loggsystem ville skrevet til en fil der verifikasjoner av de individuelle sekvensene skrives og de settes som ok hvis de er ok, og failed hvis ikke.

US-060701		Score: 3.67
-----------	--	-------------

Som oppdragsgiver ønsker jeg at testmaskinen skal kunne gi instruksjoner til WS, slik at testsekvenser som krever dette kan gjennomføres.

US-060801	Score: 3.67
-----------	-------------

Som oppdragsgiver ønsker jeg at testmaskinen skal kunne motta informasjon fra WS, slik at testsekvenser som krever dette kan gjennomføres.

US-047001	Score: 3.00
-----------	-------------

Som oppdragsgiver ønsker jeg at WS status skal overvåkes av Software, slik at Software kan merke eventuelle problemer som oppstår.

3.2.3 Kommunikasjon

Vi har gjort noen nedprioriteringer av Kommunikasjon, det primære formålet med oppgaven er å lage et fungerende gripeverktøy. Derfor ser vi det som nødvendig å fokusere på å gjøre dette best mulig. Derfor ser vi oss nødt til å prioritere ned verifikasjon. Vi tilrettelegger for rammeverket og implementerer noe, hvis vi ser at det blir ekstra tid.

US-046201	Score: 5.33
-----------	-------------

Som oppdragsgiver ønsker jeg at sensor skal kommunisere med Software på testmaskin, slik at alle hendelser som registreres kan rapporteres tilbake til test Software.

US-060201	Score: 5.00
-----------	-------------

Som oppdragsgiver ønsker jeg at testmaskinen skal kunne motta informasjon fra UR5, slik at informasjon som er viktig for loggen kommer frem.

US-060601	Score: 5.00
-----------	-------------

Som oppdragsgiver ønsker jeg at testmaskinen skal kunne motta informasjon fra sensorer, slik at informasjon som er viktig for loggen kommer frem.

US-040601	Score: 4.67
-----------	-------------

Som oppdragsgiver ønsker jeg at Software skal kunne ta imot fyringssignal, slik at Software kan verifisere fyring.

US-046301	Score: 4.67
-----------	-------------

Som oppdragsgiver ønsker jeg at kommunikasjon mellom sensor og Software på testmaskin fungerer trådløst, slik at vi slipper kabler mellom stasjonær testmaskin og bevegelig WS.

US-043201	Score: 4.33
-----------	-------------

Som oppdragsgiver ønsker jeg at EE skal kunne sende informasjon til Software, slik at EE kan verifisere når en instruksjon er gjennomført, og dette kan lagres i loggen.

US-047003		Score: 2.67
Som oppdragsgiver ønsker jeg at Software skal gi feilmelding ved automatisk stopp, slik at testoperatør lettere skal kunne finne feilkilde.		

3.2.4 Endring av design.

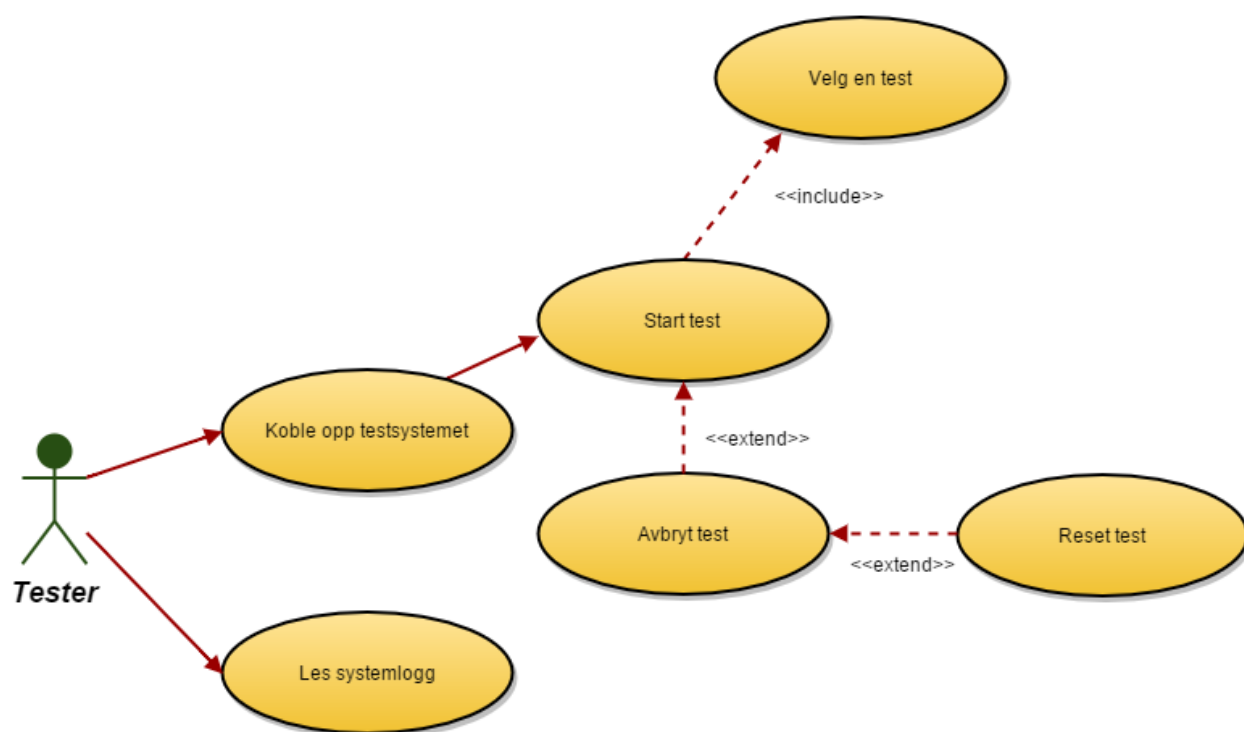
Vi har gjort noen endringer i forhold til hva vi tenkte i utgangspunktet og fjernet denne User Storyen da vi ikke finner hensiktsmessig. Den vil i stedet få en reset knapp så man kan gjøre en kontrollert tilbake stilling av systemet.

US-047004		Score: 4.33
Som testoperatør ønsker jeg at Software burde kunne automatisk tilbakestilles etter stopp, slik at den kan settes klar til ny test etter en feil.		

4 Use case diagram

Et use case diagram opprettes i de tidlige stadiene av et prosjekt. Hensikten med dette er og få oversikt over hva programmet skal utføre, uten å gå i dybden mht. hvordan det skal gjøres.

Vi bruker use case diagram som et hjelpemiddel i utviklingsprosessen, og for å kunne vise våre interessenter hvordan vi tenker. Vi har valgt å dele opp våre use caser i to slik at det blir mer oversiktlig.



Figur 2: Testprogramvare use case

4.1 Testprogramvare use case

4.1.1 Koble opp testsystemet

For å kunne kjøre testsekvenser må programvaren først kobles til robotarm og våpenstasjon. Robotarmen og testmaskinen vil være koblet til et nettverk. For å tilrettelegge for utskifting av utstyr er det nødvendig for bruker å legge inn IP adressen til robotarmen.

4.1.2 Start test

Dette vil være en knapp som kjører den valgte testsekvensen. Denne vil være lagret som en XML fil.

4.1.3 Velg en test

For at det skal være mulig å starte en test må en test være valgt, derfor må denne inkluderes når en test skal kjøres. Brukeren får mulighet til å velge en XML fil som er lagret på disk som skal kjøres.

4.1.4 Avbryt test

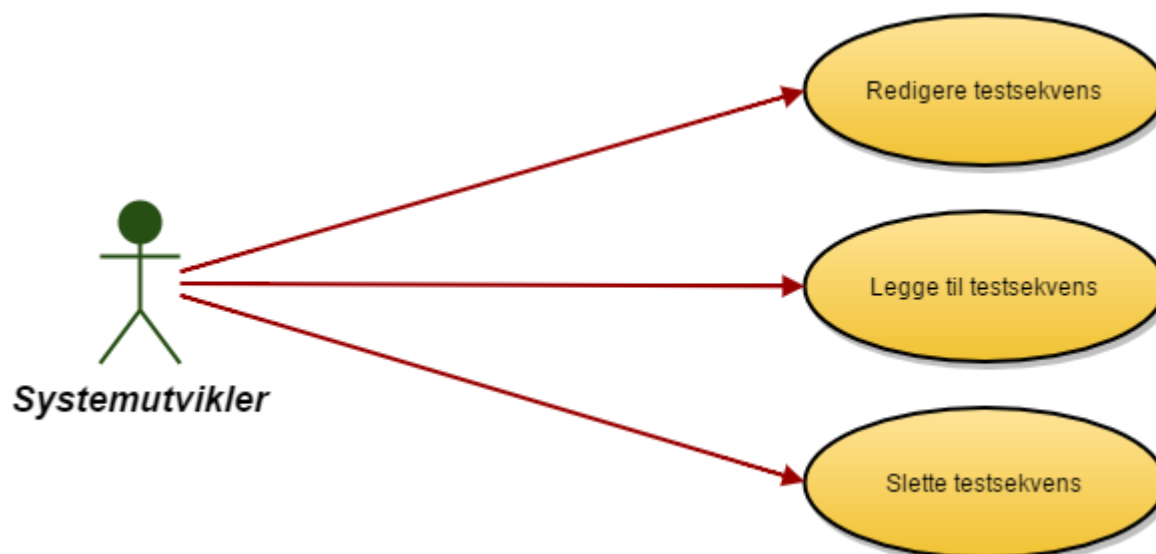
Hvis det er behov for å avslutte den valgte testen må brukeren ha mulighet til dette, dette kan være om det skjer en uventet feil. Denne er en utvidelse fra start test da den ikke er aktuell med mindre en test kjøres.

4.1.5 Reset test

Denne setter programmet tilbake til starten av den valgte testen og gir brukeren mulighet til å starte testsekvensen igjen. Dette tilfellet er en utvidelse av avbryt test.

5 Rediger testsekvens

Hensikten til dette systemet er å kunne redigere, lagre og slette testsekvenser. Ved å gi brukeren muligheten til å velge bevegelser fra en liste skal men enkelt kunne kombinere et sett med bevegelser, disse vil kunne lagres til en XML fil slik at de kan brukes på nytt senere. For at dette systemet skal virke må hver enkelt bevegelse være programmert som en separat bevegelsessekvens på UR5 robotens kontrollmaskin. Når vi kjører en sekvens fra testmaskinen vil det sendes start og stoppsignaler til og fra UR5 systemet over TCP.



Figur 3: Rediger testsekvens use case

5.1 Legge til testsekvens

Når man kjører programvaren, er det ikke definert noen bevegelser. Man må klikke på rediger testsekvens. Der får brukeren tilgang til en forhåndsdefinert liste, over alle programmerte bevegelser. Man kan ved å merke ønsket sekvens og klikke legg til eller fjern, kombinere bevegelsene som skal kjøres etter hverandre. De legges da til i en testliste, som kan lagres som XML eller kjøres direkte.

5.2 Redigere testsekvens

Under rediger test kan man også åpne en tidligere lagret XML fil og laste innholdet inn i programmet. Innholdet vises i testlisten, og man kan legge til flere bevegelser eller slette bevegelser.

5.3 Slette testsekvens

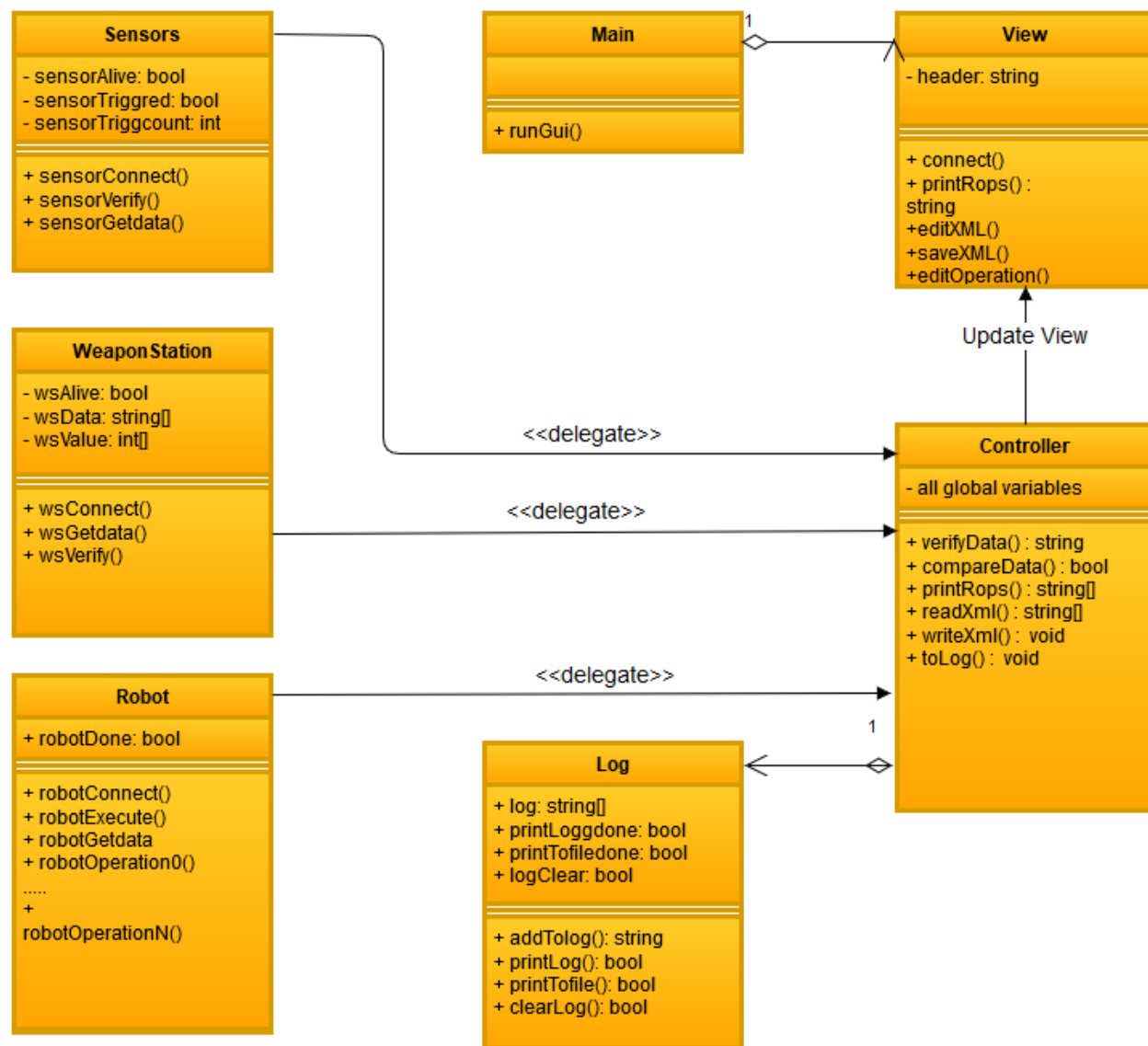
Ved å klikke slett sekvens inne i programmet tømmer man testlisten og kan begynne med blanke ark.

6 Klassediagram

Vi har satt opp et klassediagram for å definere en fornuftig oppdeling av programvaren. Vi har tatt inspirasjon fra Model-View-Controller designmønsteret, siden vårt program kommer til å være mer funksjonsorientert enn objekt orientert. Vi har behov for å gruppere disse funksjonene på 4 områder: Sensor, Våpenstasjon, Robotarm med gripeverktøy og Loggsystemet. Vi har valgt å beskrive brukergrensesnittet i et eget klassediagram. Disse genereres enkelt i Visual

Studio når vi har satt opp layouten med GUI designer verktøyet. Derfor bruker vi ikke tid på å beskrive knappefunksjonene i detalj nå.

Dette klassediagrammet er å betrakte som et utkast til design, det vil danne grunnlaget for hvordan vi deler opp koden. Etterhvert som vi ser hvordan vi trenger å definere rettigheter og får behov for variabler og funksjoner vil vi tilføye dem i diagrammet og oppdatere dette dokumentet.



Figur 4: Klassediagram

6.1 Main

Main kjører i gang det grafiske brukergrensesnittet.

6.2 View

Her kommer vi etterhvert til og ha alle elementer som skal vises til brukeren.

6.3 Controller

Controller har som formål å kommunisere med de forskjellige delene av koden. Sortere testene. Skrive til skjerm og skrive til logg. I Controller kommer vi til og lagre alle de globale variablene, og dermed skjerm data i de andre klassene. Vi vil bruke delegate til å rapportere inn til Controller.

6.4 Sensor

Vi tar utgangspunktet i en sensor som skal registrere avfyringer på våpenstasjonen. Dens fremste formål er å registrere uønskede fyringer. Vi har tenkt at vi trenger variabler som viser oss om det har blitt utløst en eller flere avfyringer. Også vil vi før testen gjøre en bekreftelse på at sensoren er tilkoblet og at den boolske verdien `sensorTriggered` er false. Vi har tenkt vi plasserer en delegat funksjon i denne klassen som rapporterer tilbake til Controller.

6.5 WeaponStation

WeaponStation vil inneholde tilkobling og all kommunikasjon med våpenstasjonen relatert til verifikasjon av testresultatet.

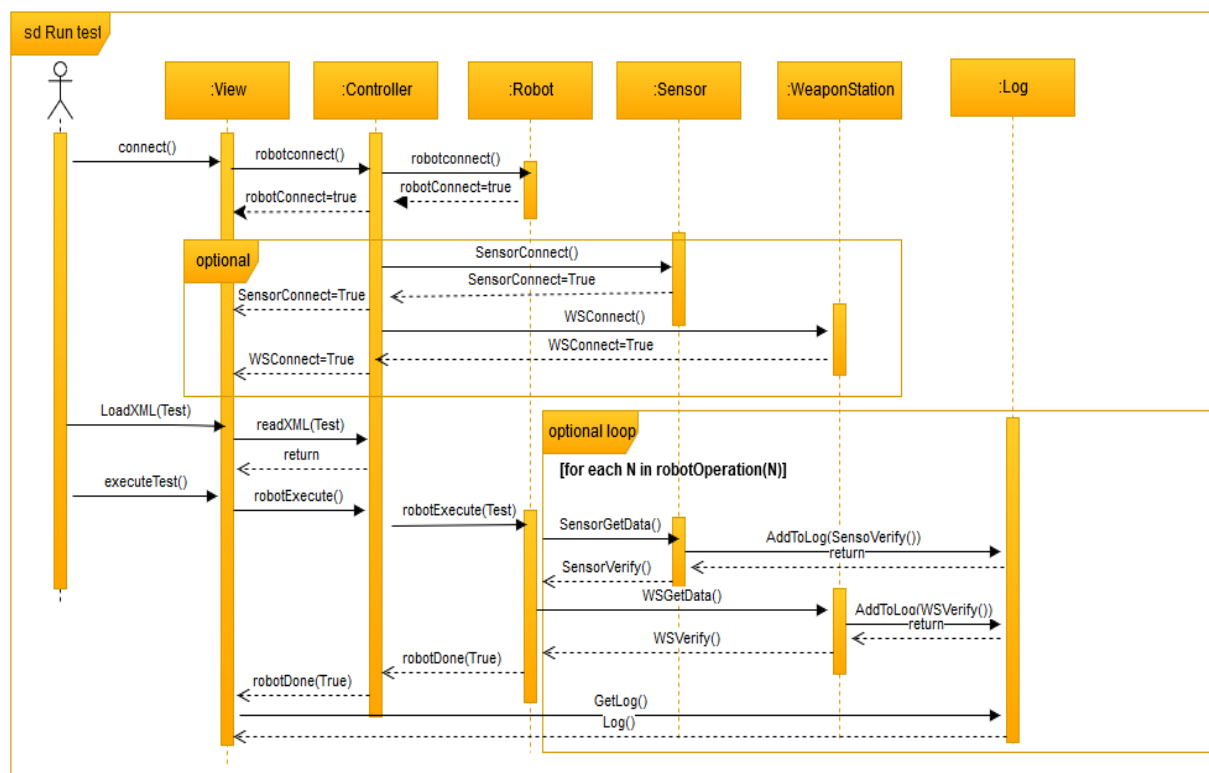
6.6 Robot

Robotklassen skal inneholde alle funksjoner tilknyttet robotarmen og gripeverktøyet. Vi trenger en funksjon for tilkobling, med en verifikasjon av tilkobling. Her vil det også være en funksjon for hver bevegelsessekvens som er programmert på robotarmen som kan kjøre den bevegelsen. Har vil vi også definere rammeverket for resultat og verifikasjon for hver bevegelse. Dette innebærer at vi har behov for en eller flere delegatfunksjoner som rapporterer tilbake til Controller med det som skal skrives til loggen.

6.7 Log

Loggklassen inneholder funksjoner for registrering av alle hendelser. Når vi kaller en bevegelse skal det skrives til log array. Når testen er gjennomført skal log array skrives til en tekstboks i brukergrensesnittet. Her har vi også tenkt at man skal kunne trykke på en knapp og få en PDF fil med loggen klar til arkivering, videresending eller utskrift. Hvis en test blir stoppet og programmet blir tilbakestilt skal `clearlog` funksjonen tømme log array.

7 Sekvensdiagram Run test



Figur 5: Run test sekvensdiagram

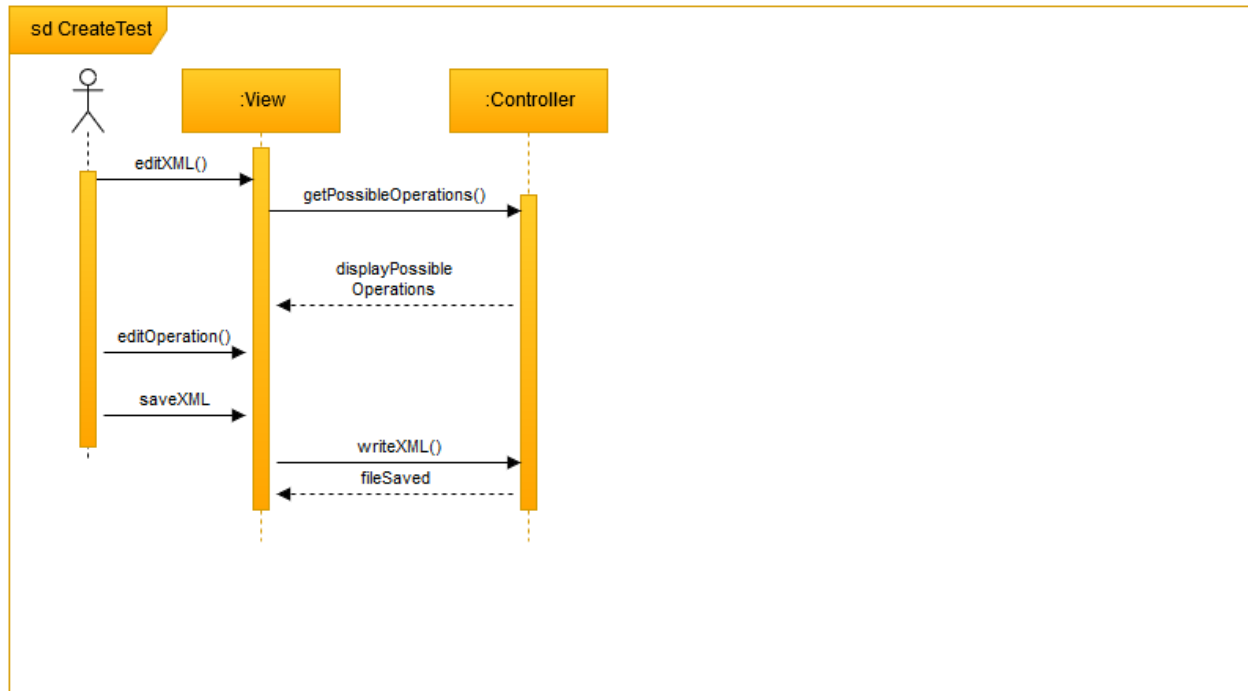
Dette sekvensdiagrammet viser hvordan en test skal kjøres i Software. Det må settes opp en tilkobling til UR 5, en XML fil med testsekvensen må velges, og brukeren må velge å kjøre en test.

Det neste som skjer er at om verifikasjon skal gjøres, må det settes opp en tilkobling til sensor og til våpenstasjon.

Når tilkobling til robotarmen er satt opp vil brukeren få mulighet til å laste inn en XML fil som beskriver testsekvensen som skal kjøres. Denne leses og brukeren får muligheten til å starte den valgte testen, de sekvensene som filen inneholder vil da være navnet på funksjoner i Robot klassen som vil kjøres i den valgte rekkefølgen.

Hvis det skal gjøres verifikasjon på testene må den valgfrie loopen kjøres. Denne sjekker for hver delbevegelse at det som er utført er riktig, og skriver dette til loggen og rapporterer tilbake til Robot klassen.

7.1 CreateTest



Figur 6: CreateTest sekvensdiagram

Dette sekvensdiagrammet viser hvordan en ny testsekvens opprettes. Dette vil være en del av det samme programmet som Run test sekvensdiagrammet. Her har brukeren mulighet til å sette opp en testsekvens fra alle de mulige bevegessekvensene. Tanken er at vi legger inn en bevegessekvens for hver knapp eller funksjon som må betjenes, så kan brukeren sette de sammen til en testsekvens og lagre den som en XML fil.

8 Konklusjoner etter implementering

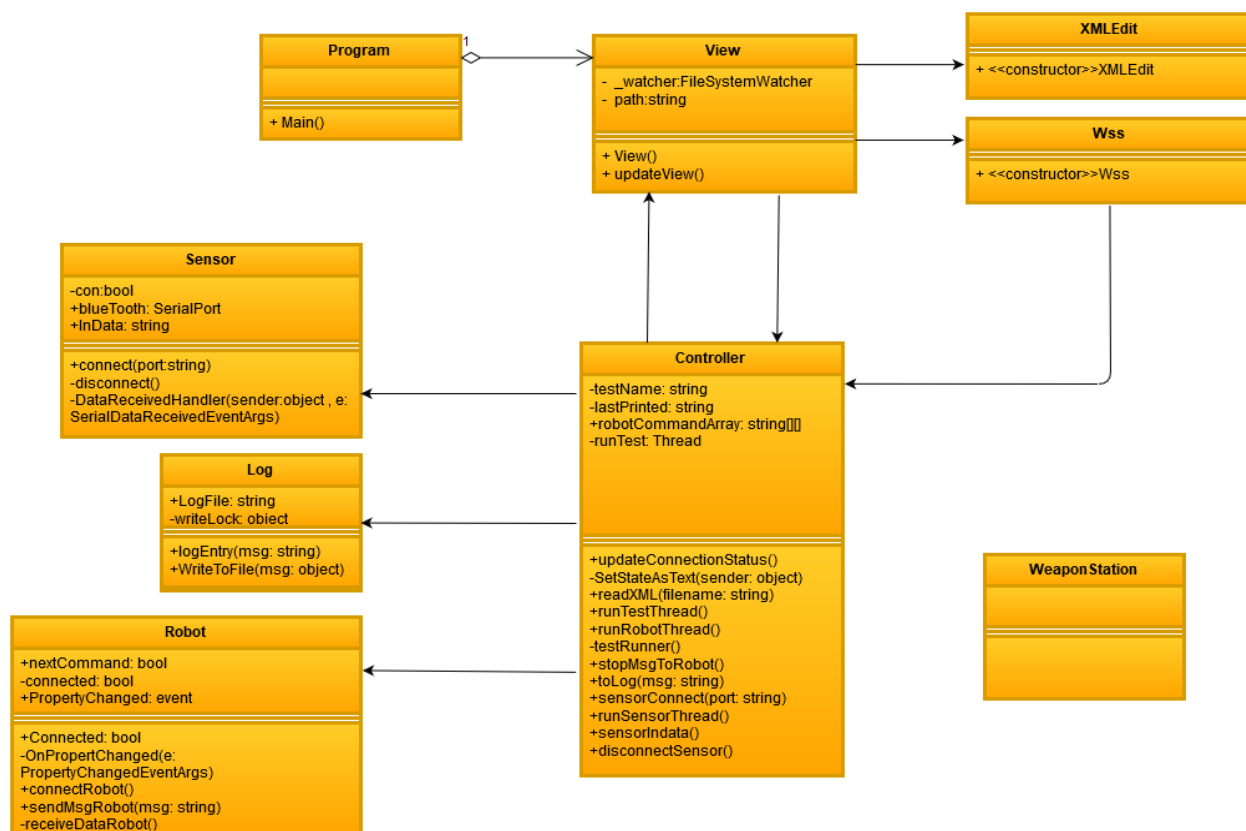
Dette kapittelet er skrevet etter at vi har implementert programmet og vi prøver å se tilbake på designfasen for å lære hva som fungerte som planlagt, og hva som måtte løses på en annen måte.

Vi ser at ettersom kompleksiteten til programmeringen vi skal utføre øker blir det viktigere å utføre en god planleggings og designfase før implementering. Dette er en god erfaring å ha med seg videre til arbeidslivet, slik at vi kan være mest mulig effektive og unngå å gjøre feil som en følge av for dårlig planlegging.

Vi ser at vi fikk implementert slik som vi hadde planlagt med tanke på våre use case diagrammer. Det eneste unntaket er «Slette testsekvens» funksjonen i «rediger testsekvens» diagrammet **[figur 3]** vårt, som ikke ble implementert. Denne funksjonaliteten oppfylles av operativsystemet hvor brukeren kan navigere til XML-fil plasseringen, og slette testsekvensen på den måten.

8.1 Klassediagram

Det første klassediagrammet vi modellerte var en god hjelp for å få kategorisert hvor vi skulle ha de forskjellige funksjonene. Den endelige implementasjonen ble noe annerledes, spesielt ble navnet på funksjoner noe annerledes, men funksjonaliteten til programmet ble som planlagt.



Figur 7: Klassediagram etter implementasjon

Denne figuren viser et klassediagram av programmet vårt slik det ble etter implementasjon. På grunn av tidsbegrensninger så vi oss nødt til å nedprioritere implementeringen av kommunikasjon og medfølgende verifikasjon som vi skulle bruke «Weapon Station» klassen til, derfor står den som en tom klasse på dette diagrammet. Den er opprettet i programmet vårt, men står også der som en tom klasse.

«Wss» og «XMLEdit» er en del av brukergrensesnittet, og er henholdsvis for kommunikasjon med sensor på våpenstasjon og redigering av XML testsekvenser.

Vi kan se at likhetene med det første klassediagrammet vårt [figur 3] er store på «Controller», «Sensor», «Log» og «Robot» klassene. Vi fikk også implementert at «Controller» klassen tar seg av mesteparten av kommunikasjonen mellom klassene i programmet vårt, noe som gjør det mer oversiktlig for utenforstående å forstå og enklere for oss å jobbe med.

8.2 Konklusjon

Implementeringen av programmet gikk bedre enn vi hadde trodd på forhånd, noe som var en god erfaring å ta med seg. Vi tror at dette har en sammenheng med at vi brukte god tid i designfasen og fikk tenkt gjennom problemstillinger som kunne oppstå underveis i programmeringen før vi begynte på selve implementasjonen. Dette sparte oss for en del tid under implementeringen noe som kom godt med da det nesten alltid tar lengre tid å implementere en det som blir planlagt.

Kilder

[1] A.K. Svendsen; E.L. Roa; H.B. Sørums; S. Rudin. "Automated CROWS Testing"
Bachelor Thesis, Fakultet for teknologi og maritime fag, Høgskolen i Buskerud og
Vestfold, Kongsberg, 2014.

Tilgjengelig fra:

<http://hdl.handle.net/11250/216910>

Sist sjekket: 16.03.2016

[2] Universal Robots AS. (2015).

UR5 - robot.

Tilgjengelig fra:

<http://www.universal-robots.com/no/produkter/ur5-robot/>

Sist sjekket: 16.03.2016

[3] Universal Robots support

Ethernet socket communication via URScript

[Nettside]

Tilgjengelig Fra:

<http://www.universal-robots.com/how-tos-and-faq/how-to/ur-how-tos/ethernet-socket-communication-via-urscript-15678/>

Sist sjekket: 16.03.2016

Software teknologidokument

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAKSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Software teknologidokument
VERSJON / DATO	1.0 / 19. Mai 2016
FORFATTER	Espen S. Kraglund og Thomas Wegener
SIDEANTALL	32



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



i. Abstrakt

Dette dokumentet beskriver utviklingen av programvaren for et testsystem for ny programvare på CROWS M153 våpenstasjoner. Det gis en gjennomgang av nødvendig bakgrunns teori for den valgte løsningen, beskrivelse av noen programmer som har vært nyttige under utviklingen av systemet og en gjennomgang av løsningene som ble valgt og hvordan vi har gått frem når vi har utviklet koden.

ii. Innholdsfortegnelse

i. Abstrakt.....	2
ii. Innholdsfortegnelse	3
iii. Oversikt vedlegg	4
iv. Oversikt tabeller	4
v. Oversikt figurer	4
vi. Dokumenthistorie.....	5
vii. Forkortelser	6
1 Innledning	6
2 Bakgrunns teori.....	7
2.1 UR Script Programming Language.....	7
2.2 System.Xml.Linq Namespace.....	7
2.3 TcpListener.....	7
2.4 Windows forms.....	7
2.5 XML.....	7
3 Programmer brukt i utviklingen.....	8
3.1 GIT	8
3.2 SourceTree.....	9
3.3 Hercules SETUP utility	11
3.4 Putty	11
3.5 Visual Studio	12
4 Testprogrammet.....	12
4.1 Klasser	12
4.1.1 View klassen	13
4.1.2 WSS klassen.....	15
4.1.3 XMLEdit klassen	15
4.1.4 Controller klassen	16
4.1.5 Robot klassen	19
4.1.6 Log klassen.....	22

4.1.7 Sensor klassen	23
4.2 UR5 kode	25
5 Konklusjon	29
Kilder	30

iii. Oversikt vedlegg

Vedlegg A: Versjonshistorie fra SourceTree.

iv. Oversikt tabeller

Tabell 1: Dokumenthistorie.....	5
Tabell 2: Forkortelser	6

v. Oversikt figurer

Figur 1: XML Eksempel.....	8
Figur 2: GIT mappe på server	9
Figur 3: SourceTree	10
Figur 4: Visualisering av oppdatering i SourceTree	10
Figur 5: Kommunikasjon med GIT server	11
Figur 6: Kodekart fra Visual Studio.....	12
Figur 7: Brukergrensesnitt.....	13
Figur 8: Eksempel på klikk event.....	14
Figur 9: FileStream	14
Figur 10: MouseDown funksjon.....	15
Figur 11: DragDrop funksjon	16
Figur 12: Objekter i Controller klassen	16
Figur 13: For løkke i testRunner funksjonen.....	17
Figur 14: sensorIndata funksjonen	18
Figur 15: get; set; for Connected.....	19
Figur 16: OnPropertyChanged funksjonen	19
Figur 17: connectRobot funksjonen.....	20
Figur 18: Utdrag fra receiveDataRobot funksjonen.....	21
Figur 19: sendMsgRobot funksjonen.....	21
Figur 20: logEntry funksjonen.....	22

Figur 21: writeToFile funksjonen	23
Figur 22: Brukergrensesnitt for tilkobling til sensor	24
Figur 23: connect funksjon for å koble til sensor.....	24
Figur 24: Handlinger ut ifra mottatte data fra sensor	25
Figur 25: Nettverkstilkobling fra UR 5.....	26
Figur 26: Løkke som kjøres om strengen "item_1" mottas.....	26
Figur 27: Lyttetråd på UR 5.....	27
Figur 28: Subprogram for å gripe joystick.....	28

vi. Dokumenthistorie

Tabell 1: Dokumenthistorie

v 0.1 02.05.16	Thomas: Dokument opprettet
v 0.2 03.05.16	Espen: skrevet noe overordnet og beskrevet noen klasser
v 0.3 03.05.16	Espen: Skrevet på klassekapitelet
v 0.4 12.05.16	Espen: Skrevet abstrakt
v 0.5 15.05.16	Espen: Lagt til noen bilder og skrevet på klasser.
v 0.6 18.05.16	Thomas og Espen: ferdigstilling.
v 0.7 19.05.16	Espen: Skrevet innledning og ferdigstilt.
v 1.0 19.05.16	Espen og Thomas: Ferdigstilt og godkjent av begge.

vii. Forkortelser

Tabell 2: Forkortelser

Forkortelser	Forklaring
KPS	Kongsberg Protech Systems
HSN	Høgskolen i Sørøst-Norge
ACT	Automated CROWS Testing
ACTEE	Automated CROWS Testing End Effector
CG	Control Grip, styringsjoystick for våpenstasjon
DCP	Display Control Panel, kontrollpanel for våpenstasjon
CROWS	Common Remotely Operated Weapon Station
UR5	Universal Robots 5 robotarm
MVC	Model-View-Controller
XML	eXtensible Markup Language

1 Innledning

Dette dokumentet er en del av bacheloroppgaven til gruppen ACTEE, vi har som oppgave å forenkle testing ved ny programvare på CROWS ved å utvikle et gripeverktøy for en UR 5 robot arm med test programvare som kan utføre disse testene.

Dette dokumentet beskriver utviklingsprosessen til denne programvaren.

Vi vil gi en kort innføring i noe nødvendig bakgrunns teori, forklare noen av programmene vi har brukt under utviklingen og gjøre en gjennomgang av koden vi har implementert. Til slutt konkluderer vi med hvordan utviklingsprosessen har vært, og trekker inn det vi har lært underveis.

For mer innsikt i hvorfor vi har gjort de valgene vi har gjort, henviser vi til designfasen vår **[Software designdokument]** hvor vi planla hvordan vi best kunne løse de utfordringene som utviklingen av dette systemet gir. Denne utviklingsprosessen har benyttet seg av SCRUM modellen, for å se hvordan vi har oppfylt de forskjellige user storiene henviser vi til Testverifikasjonsdokumentet **[Testverifikasjon og risiko]**

2 Bakgrunns teori

I dette kapittelet vil vi beskrive teknologier brukt under utviklingen av software til ACTEE systemet som det er nødvendig at leseren setter seg inn i for å forstå deler av dette dokumentet.

2.1 UR Script Programming Language.

Programmering på robotarmen foregår ved hjelp av dette skriptspråket. [1] UR 5 robotarmen[2] har en touch skjerm som bruker det grafiske brukergrensesnittet Polyscope. Der har brukeren mulighet til å opprette programmer som skriptes med UR Script Programming Language. Dette er et programmeringsspråk med de fleste muligheter som er gjenkjennbare fra andre programmeringsspråk slik som variabler, løkker og tråder. Det er utstrakt bruk av ferdige kodeblokker som kan “drag and droppes” i programmet. Dette gjør det veldig enkelt og programmere, om en har en grunnleggende forståelse av programmering.

2.2 System.Xml.Linq Namespace

System.Xml.Linq Namespace[3] er et Namespace i C# som inneholder klasser som kan brukes til å modifisere og opprette XML filer på en enkel og effektiv måte. Her er det klasser som XElement som representerer et XML element og XAttribute som representerer en XML attributt. Disse klassene kan brukes til å lese en XML fil ut ifra hvilke tagger de har, noe som gjør det mulig å ha informasjon om testsekvenser lagret i en XML struktur og velge hvilke elementer vi skal ha med hjelp av dette Namespacet.

2.3 TcpListener

TcpListener[4] er en klasse i C# som, ved hjelp av System.Net.Sockets namespacet kan opprette en tilkobling til et nettverkspunkt. Dette gjøres ved at det settes opp ett lytteobjekt som kan lytte etter innkommende forsøk på å koble seg til en oppgitt IP adresse og port nummer. Dette er en enklere måte enn å bruke Sockets klassen i System.Net.Sockets som er mer kompleks å benytte.

2.4 Windows forms

Windows forms er brukergrensesnittbyggeren i Visual Studio. Ved å bruke denne blir byggingen av brukergrensesnitt betydelig enklere. Det er drag and drop funksjonalitet for å få knapper, lister, tekstbokser og andre elementer inn i brukergrensesnittet.

Disse elementene får funksjonalitet ved at det opprettes en klikk event som beskriver hvilken funksjonalitet som skal kjøres når brukeren bruker det gitte elementet.

Bruken av Windows forms forenkler byggingen av brukergrensesnitt betydelig, da det sparer tid og kunne bygge det med ferdiglagd kode.

2.5 XML

eXtensible Markup Language er et markeringsspråk som er lesbart både for maskin og for mennesker. Det er deskriptive tagger som definerer hvilken type innholdet mellom taggene er.

For våre testsekvenser har vi hentet inspirasjon fra foregående bacheloroppgave, ACT[5] for navnene på taggene. XML dokumentet inneholder tagger som test, sequence og cmd. Testtaggene inneholder hele testsekvensen, sequence inneholder en sekvens av kommandoer som er en del av en hel test og cmd inneholder navnet på bevegelsen som skal sendes som en streng til robotarmen.

```
<test totalSeqs="1" testname="Testname">
  <sequence id = "1">
    <cmd part="1">item_4</cmd>
    <cmd part="2">item_6</cmd>
  </sequence>
</test>
```

Figur 1: XML Eksempel

Denne figuren viser en forenklet testsekvens lagd på måten som beskrevet over hvor det er to kommandoer som kan sendes til UR 5: item_4 og item_6.

3 Programmer brukt i utviklingen

Under utvikling av software var det nødvendig å bruke flere programmer for å forenkle utviklingsprosessen. Dette kapittelet gir en kort innføring i disse programmene.

3.1 GIT

GIT[6] er et versjonskontrollsystem utviklet for å holde styr på versjoner og koordinere samarbeidet med koden. GIT er utviklet av Linux kernel utviklergruppen med blant andre Linus Torvalds.

Vi bruker GIT til versjons og revisjonskontroll. Vi kjører GIT på de maskinene vi utvikler applikasjonen på og i tillegg kjører vi GIT på en privat server. Denne finnes på git.actee.no. Der har vi private brukerkontoer og bruker SSH krypteringsnøkler for å koble oss til. Hver gang vi har skrevet ett tillegg til koden eller gjort endringer dytter vi de opp til serveren. Serveren varsler om nye endringer og disse kan lastes ned og eventuelt flettes sammen med lokale endringer som siden dyttes tilbake. På denne måten holder vi styr på hvem som gjør hva og vi får muligheten til å gå tilbake til en tidligere versjon hvis noe skulle gå galt.

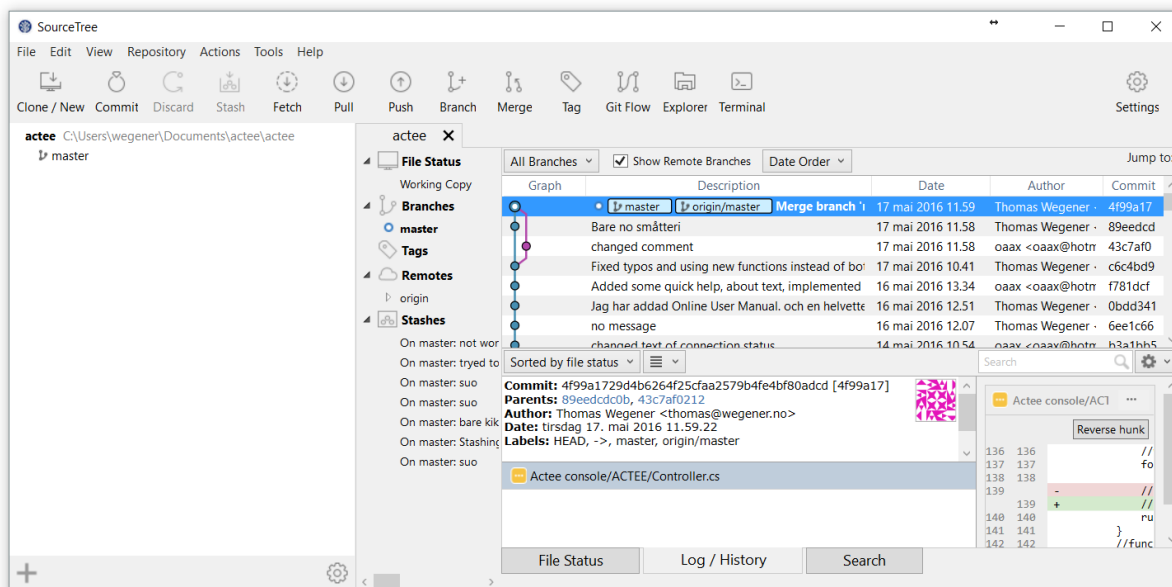

```
wegener@development:/srv/git/project/actee.git$ ls -la
total 44
drwxrwsr-x  7 wegener git 4096 Apr 18 23:10 .
drwxrwsr-x  3 wegener git 4096 Apr 18 22:52 ..
drwxrwsr-x  2 wegener git 4096 Apr 18 22:52 branches
-rw-rw-r--  1 wegener git   66 Apr 18 23:09 config
-rw-rw-r--  1 wegener git   73 Apr 18 22:52 description
-rw-rw-r--  1 wegener git   23 Apr 18 22:52 HEAD
drwxrwsr-x  2 wegener git 4096 Apr 18 22:52 hooks
drwxrwsr-x  2 wegener git 4096 Apr 18 22:52 info
drwxrwsr-x 241 wegener git 4096 May 16 13:34 objects
drwxrwsr-x  4 wegener git 4096 Apr 18 22:52 refs
-rw-rw-r--  1 wegener git    6 Apr 18 23:10 test
wegener@development:/srv/git/project/actee.git$
```

Figur 2: GIT mappe på server

Vi har valgt å kjøre Git på en egen server, det finnes mange som tilbyr versjonskontrolltjenester. Men de er ofte begrenset og koden ligger ofte åpen, ikke bare ubeskyttet. For at vi skulle ha kontroll på tilgang og kjøre prosjektet vårt privat og i tillegg være litt kritiske til bruk av penger valgte vi å investere tid i konfigurasjon av vårt eget system.

3.2 SourceTree

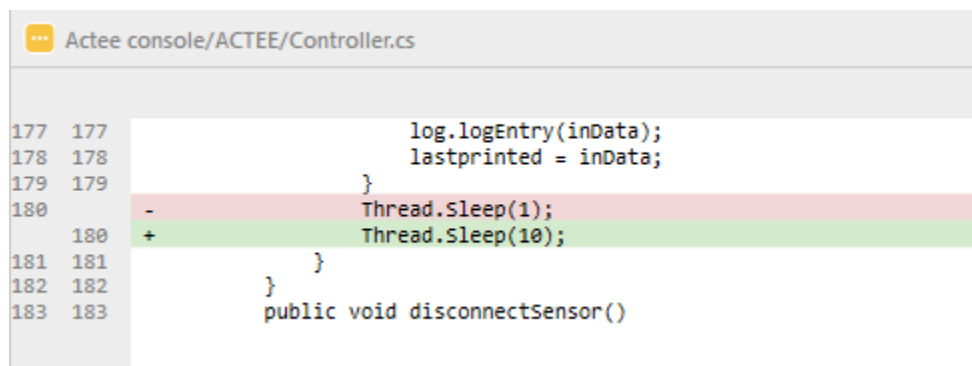
SourceTree er ett enkelt grafisk brukergrensesnitt til GIT som vi har tatt i bruk på maskinene hvor vi har utviklet applikasjonene. Dette inneholder all funksjonaliteten til GIT, men gjør at vi slipper å bruke terminal for å bruke GIT.



Figur 3: SourceTree

Fordelene ved å bruke et grafisk brukergrensesnitt over å bruke terminal er å spare tid, da spesielt ved å slippe å lære oss alle kommandoene for å bruke GIT.

Vi kan se i historien og revidere kode linje for linje, hvor vi også kan se på de enkelte klassefilene og hvilke forandringer denne oppdateringen har gjort.



Figur 4: Visualisering av oppdatering i SourceTree

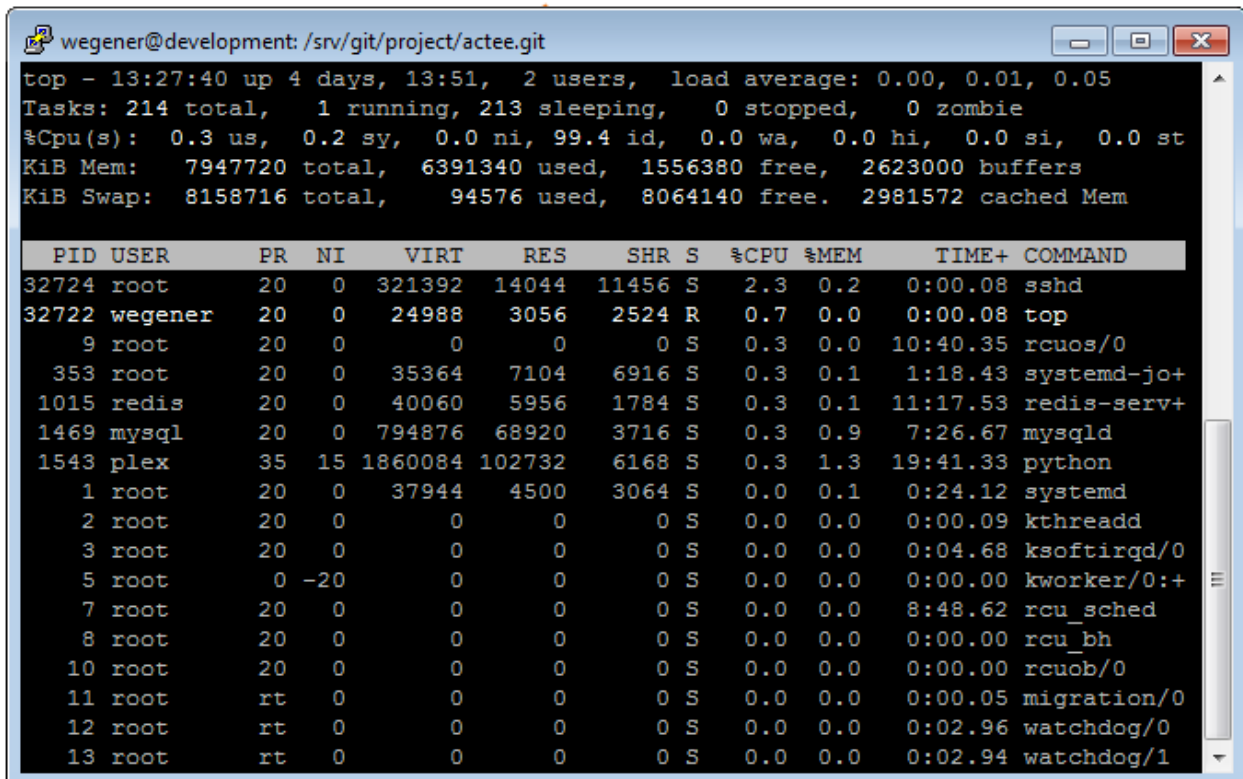
Denne figuren viser hvordan SourceTree viser hvilke forandringer som har kommet i en ny versjon av programmet.

3.3 Hercules SETUP utility

Hercules er et program som inneholder terminaler for forskjellige nettverksstandarder som TCP/IP og UDP. Vi brukte dette for å teste tilkoblingen til ett nettverkspunkt og sending av strenger over dette nettverket. Hercules virker da som klient og kan koble seg til programmet vårt og sende strenger, noe som gjør at vi får testet all funksjonaliteten til nettverksprogrammeringen vår. Dette forenkler testingen av nettverkstilkoblingen betraktelig da det ikke er nødvendig å fysisk være i nærheten av UR 5 eller bruke tid på å koble den opp.

3.4 Putty

Vi har brukt Putty for å kommunisere med en privat linuxserver der vi lagrer og håndterer versjonskontroll for kildekode med GIT, nettsidene til prosjektet og for å overvåke seriellporten under testing.



```
wegener@development: /srv/git/project/actee.git
top - 13:27:40 up 4 days, 13:51, 2 users, load average: 0.00, 0.01, 0.05
Tasks: 214 total, 1 running, 213 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.3 us, 0.2 sy, 0.0 ni, 99.4 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem: 7947720 total, 6391340 used, 1556380 free, 2623000 buffers
KiB Swap: 8158716 total, 94576 used, 8064140 free. 2981572 cached Mem

  PID USER      PR  NI   VIRT    RES    SHR  S  %CPU  %MEM   TIME+ COMMAND
 32724 root        20   0  321392  14044  11456 S   2.3   0.2   0:00.08 sshd
 32722 wegener    20   0   24988   3056   2524 R   0.7   0.0   0:00.08 top
    9 root        20   0     0     0     0 S   0.3   0.0  10:40.35 rcuos/0
   353 root        20   0   35364   7104   6916 S   0.3   0.1   1:18.43 systemd-jo+
  1015 redis      20   0   40060   5956   1784 S   0.3   0.1  11:17.53 redis-serv+
  1469 mysql     20   0  794876  68920   3716 S   0.3   0.9   7:26.67 mysqld
  1543 plex       35  15 1860084 102732   6168 S   0.3   1.3  19:41.33 python
    1 root        20   0   37944   4500   3064 S   0.0   0.1   0:24.12 systemd
    2 root        20   0     0     0     0 S   0.0   0.0   0:00.09 kthreadd
    3 root        20   0     0     0     0 S   0.0   0.0   0:04.68 ksoftirqd/0
    5 root        0 -20     0     0     0 S   0.0   0.0   0:00.00 kworker/0:++
    7 root        20   0     0     0     0 S   0.0   0.0   8:48.62 rcu_sched
    8 root        20   0     0     0     0 S   0.0   0.0   0:00.00 rcu_bh
   10 root        20   0     0     0     0 S   0.0   0.0   0:00.00 rcuob/0
   11 root        rt   0     0     0     0 S   0.0   0.0   0:00.05 migration/0
   12 root        rt   0     0     0     0 S   0.0   0.0   0:02.96 watchdog/0
   13 root        rt   0     0     0     0 S   0.0   0.0   0:02.94 watchdog/1
```

Figur 5: Kommunikasjon med GIT server

3.5 Visual Studio

Visual Studio er Microsofts integrerte utviklingsverktøy for programmering. Vi har programmert i C# og Visual Studio har hjulpet oss med referanser, kompileringsnotater og versjonshistorie som genereres automatisk fra GIT versjonshistorien. Vi kan se endringer, forfatter, liste biblioteker, og ha en komplett oversikt over systemet vi utvikler og alle funksjoner og objekter som lenker det sammen.

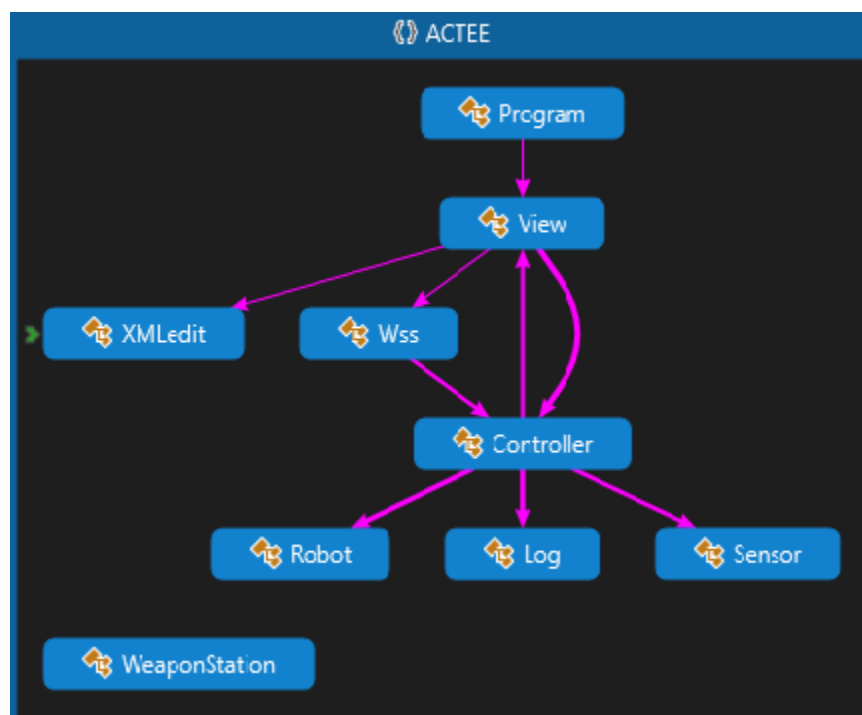
4 Testprogrammet

I dette kapittelet forklares kildekoden til programmet ved å forklare den overordnede strukturen til klassene og så forklares hver enkelt klasse med de funksjoner de innehar.

Testprogrammet er strukturert etter planleggingen vi gjorde i designfasen ved hjelp av UML diagrammer. Funksjoner er lagt til i beskrivende, separate klassefiler. Dette gjør koden vår mer oversiktlig for andre å lese og gjør den enklere for oss å jobbe med.

Hovedfunksjonen til programmet er å la brukeren kjøre test sekvenser og logge de hendelsene som skjer under sekvensene.

4.1 Klasser



Figur 6: Kodekart fra Visual Studio

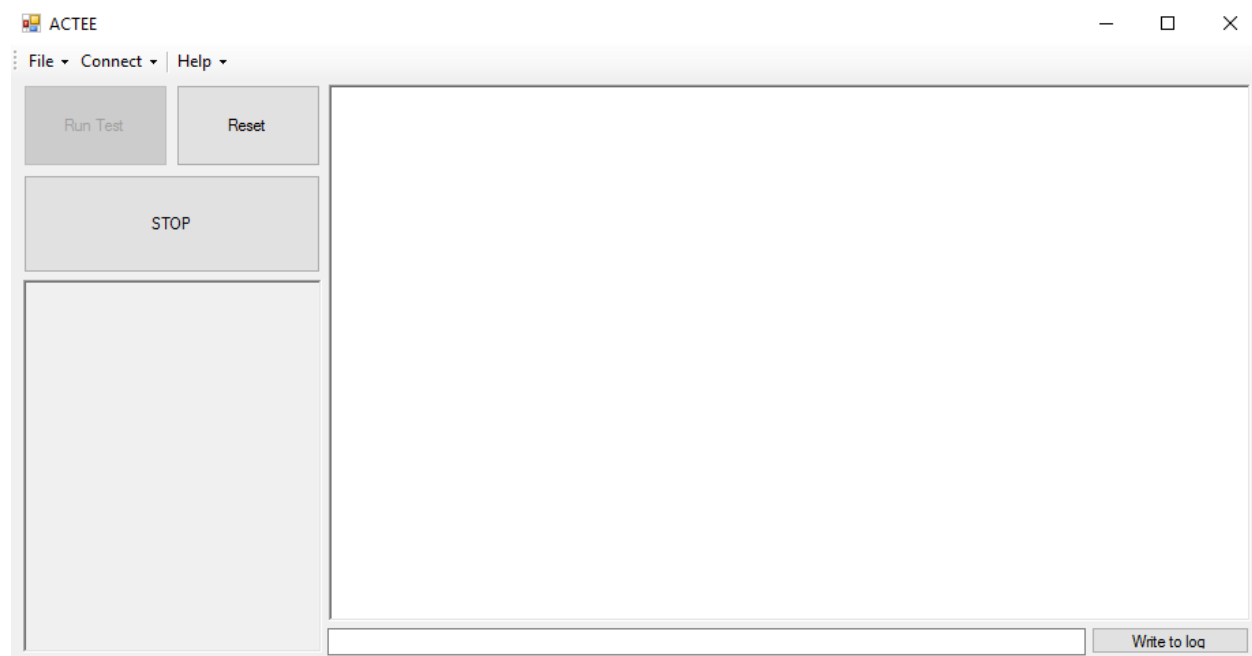
Dette kodekartet viser klassene våre og hvem av de som kommuniserer med hverandre. Vi har hentet inspirasjon fra «Model-View-Controller» designmønsteret når vi designet programvaren. MVC deler opp et program med grafisk brukergrensesnitt i tre komponenter: Model, View og Controller. [7]

Model holder på dataene som skal brukes i programmet. Log, Sensor og Robot klassene våre er en del av Model komponenten. Controller manipulerer og sender data videre i systemet, hos oss er denne komponenten representert med klassen Controller. View viser dataene til brukeren, hos oss er brukergrensesnittet plassert i tre klasser hvor View er hovedvinduet som brukeren får opp når programmet startes.

Program klassen kjøres når programmet starter, denne inneholder en initialisering av loggfilen og et kall til å starte View.

4.1.1 View klassen

View klassen representerer brukergrensesnittet som presenteres til brukeren av programmet ved kjøring. For mer inngående informasjon om hvordan brukergrensesnittet fungerer se kapittel 4 i Brukermanualen.



Figur 7: Brukergrensesnitt

View klassen inneholder hovedsakelig klikk events. Dette er handlinger som programmet skal gjøre om det klikkes på noen av knappene eller velges noe fra drop down menyene.


```
private void connectToUR5ToolStripMenuItem_Click(object sender, EventArgs e)
{
    //set enabled to false so that the user cant open more connections on one port
    //that will result in a exception
    connectToUR5ToolStripMenuItem.Enabled = false;
    //call the function that starts the connection to robot thread
    controller.runRobotThread();
    controller.setForm(this);
}
```

Figur 8: Eksempel på klikk event

Denne figuren er et eksempel på en klikk event, når det trykkes på “Connect” knappen i brukergrensesnittet vil den knappen disables, funksjonen som starter tilkoblingstråden til roboten vil starte og vi sender en View referanse til Controller.

For å oppdatere loggvinduet som vises i programmet er det satt opp en eventlistener som overvåker filen log.txt i programmappen. Denne tekstfilen er der loggen lagres, og den lastes inn på nytt når programmet startes hvis den ikke blir slettet. Dette gir brukeren muligheten til å få tak i data hvis systemet krasjer. Og dette er hovedargumentet for å lagre loggen til fil og ikke beholde den i et array. Eventlistenneren er en filesystem watcher. Som følger med på log.txt i mappen programmet kjører fra, den ser etter skrijving til og overskriving av filen og når det er endring bruker vi et lambda uttrykk for å kjøre en try catch som forsøker å lese filen.

```
//Using filestream handles read and write accsett for the file. We already write to it from the log class.
using (FileStream fileStream = new FileStream(
    Path.GetFileName(Log.LogFile),
    FileMode.Open,
    FileAccess.Read,
    FileShare.ReadWrite))
{ //using streamreader allows us to grab the content of the file,
    using (StreamReader streamReader = new StreamReader(fileStream))
    {
        var logText = streamReader.ReadToEnd();
        if (!txtLogOutput.IsDisposed)
            Invoke(new Action(() => txtLogOutput.Text = logText ));
    }
}
```

Figur 9: FileStream

Siden vi både leser og skrive til denne filen setter vi FileShare.ReadWrite selv om vi bare skal lese i denne funksjonen. Dette gjør at vi kan lese filen med StreamReader uten å havne i konflikt med log klassen som låser og åpner filen for å skrive til den.

Så bruker vi invoke for å skrive logText ut til tekstvinduet. På teststadiet hadde vi problemer med tilfeller hvor log.txt ikke eksisterte og vi laget derfor en exception meldingsboks som forteller om problemer med å lese tekstfilen. Senere laget vi en funksjon som oppretter filen hvis den ikke eksisterer.

4.1.2 WSS klassen

WSS er en utvidelse av View for å åpne et vindu der brukeren velger hvilken port som skal brukes når det skal opprettes en tilkobling til sensoren. Dropdown menyen der brukeren velger port fylles med tilgjengelige porter på den maskinen programmet kjøres fra.

4.1.3 XMLEdit klassen

XMLEdit er også en utvidelse av View klassen, dette åpner et vinduet over hovedbrukergrensesnittet hvor brukeren kan lage sin egen testsekvens ut ifra et utvalg av mulige bevegelser som kan utføres på CG og DCP. Dette kan brukeren lagre som en XML fil som kan kjøres av hovedprogrammet.

Dette gjøres ved at det er lagd to listebokser hvor den ene holder en fullstendig liste over de mulige bevegelsene, og den andre representerer sekvensen brukeren vil lage.

```
//this function is called on mousedown on listBox1 it is needed to implement drag and
//drop functionality from one listBox to another
1 reference | oaax, 4 days ago | 1 author, 2 changes
private void listBox1_MouseDown(object sender, System.Windows.Forms.MouseEventArgs e)
{
    //if the listBox is empty, then exit program
    if (listBox1.Items.Count == 0)
        return;

    int index = listBox1.IndexFromPoint(e.X, e.Y);

    //We do an if check here to make sure that we do not get an indexoutofrange exception
    //because the index is set to -1 if we try to choose a blank spot in the listBox
    if (index != (-1))
    {
        string s = listBox1.Items[index].ToString();
        DragDropEffects ddel = DoDragDrop(s,
            DragDropEffects.All);

        if (ddel == DragDropEffects.All)
        {
            listBox1.Items.RemoveAt(listBox1.IndexFromPoint(e.X, e.Y));
        }
    }
}
```

Figur 10: MouseDown funksjon

Når brukeren velger et element i listen over mulige bevegelser ved å trykke på musen, kjøres det en MouseDownEvent som sjekker hvor i listeboksen brukeren har valgt det og tar det

elementet og kopierer det, dette gjøres ved funksjonen listBox2_DragOver som bare sier at effekten som skal skje er at elementet skal kopieres.

For at brukeren skal kunne se i brukergrensesnittet hvor det er lov å slippe det valgte elementet, er det implementert en funksjon listBox2_DragEnter som skifter musepeker ikonet når brukeren kommer innenfor et område på brukergrensesnittet hvor det mulig å slippe elementet.

```
1 reference | oasx, 20 days ago | 1 author, 1 change
private void listBox2_DragDrop(object sender, System.Windows.Forms.DragEventArgs e)
{
    if (e.Data.GetDataPresent(DataFormats.StringFormat))
    {
        string str = (string)e.Data.GetData(
            DataFormats.StringFormat);

        listBox2.Items.Add(str);
    }
}
```

Figur 11: DragDrop funksjon

Når brukeren da slipper musepekeren i listeboksen hvor sekvensen bygges vil listBox2_DragDrop funksjonen kalles og en kopi av det valgte elementet vil bli lagt på neste plass i listeboksen.

Når brukeren er ferdig med å bygge sekvensen, og ønsker å lagre dette slik at det kan kjøres i hovedvinduet utføres dette av saveTestToolStripMenuItem_Click funksjonen. Denne funksjonen åpner et vindu for å lagre en fil, ved å bruke saveFileDialog. Og skriver til denne filen ved å opprette en instans av System.IO.StreamWriter som skriver innholdet fra listeboksen formatert med XML tagger.

4.1.4 Controller klassen

Tanken bak Controller klassen er at det er den som skal være ansvarlig for kommunikasjonen mellom View og Model komponentene. Dette betyr at det er her det opprettes tråder av funksjoner fra andre klasser, og her data blir modifisert og sendt videre til andre klasser.

```
class Controller
{
    Robot robot = new Robot();           //make object of Robot class
    Log log = new Log();                  //make object of log class
    Sensor sensor = new Sensor();         //make object of sensor class
}
```

Figur 12: Objekter i Controller klassen

Som figuren viser, opprettes det objekter av de klassene vi trenger å bruke, eller modifisere data fra i begynnelsen av Controller klassen.

For at datamaskinen skal kunne lese XML filer som brukeren velger i brukergrensesnittet har vi implementert readXML funksjonen. Denne fungerer ved at fil lokasjonen blir tatt inn som argument når funksjonen kalles.

XML filen leses ved å bruke System.Xml.Linq Namespacet som er forklart i kapittel 1. Ved hjelp av dette kjøres det spørringer mot XML filen som gjør at den deles opp ut ifra hvilke tagger de forskjellige elementene har. Vi itererer så over elementene i XML filen og lagrer dette i samme strukturen som i XML filen i et jagged array som vi kan iterere over senere når vi kjører sekvensen.

TestRunner funksjonen kjører testsekvensen som brukeren har valgt og skriver informasjonen om testsekvensen til loggen.

```
for (int i = 0; i < robotCommandArray.Length; i++)
{
    //write current sequence number to log
    log.logEntry("Running sequence : " + (i+1) );

    for (int j = 0; j < robotCommandArray[i].Length; j++)
    {
        //write current movement to log
        log.logEntry("Running movement: " + robotCommandArray[i][j]);

        //send message to robot with the command we want it to execute
        robot.sendMessageRobot(robotCommandArray[i][j]);

        //the thread will spin until the nextCommand is set to true
        System.Threading.SpinWait.SpinUntil(() => robot.nextCommand);

        //When the thread starts again the movement will be complete so write that to log
        log.logEntry("command : " + robotCommandArray[i][j] + " complete");

        //set the boolean to false so we can wait for an OK to be sent
        robot.nextCommand = false;
    }
    //write to log that the current sequence is completed
    log.logEntry("Sequence " + (i + 1) + " complete");
}
```

Figur 13: For løkke i testRunner funksjonen

Figuren viser hvordan det itereres over elementene fra testsekvensen i testRunner funksjonen. Det er flere kall til Log klassen her, hvor det sendes en string til logEntry funksjonen som gir brukeren informasjon om sekvensen som kjøres.

For hver kommando som ligger i arrayet sendes verdien av denne til sendMessageRobot funksjonen i Robot klassen, som sender informasjonen til roboten. Der vil det bli kjørt en bevegelse som har navn lik den strengen som er i arrayet.

For å forhindre at det sendes en ny kommando til roboten før den forrige er ferdig kalles SpinWait.SpinUntil funksjon som lar tråden spinne til den får beskjed fra roboten at den er klar for neste kommando. Dette gjøres ved at det sendes en "OK" string til test programmet, og i Robot klassen blir den boolske verdien nextCommand satt til sann som gjør at løkken går videre.

Etter dette setter vi nextCommand til usann, slik at løkken også vil stoppe etter den neste kommandoen har blitt sendt til roboten.

For View klassen som bare har et objekt av Controller har det blitt opprettet en funksjon toLog som videresender en streng til Log.logEntry.

Vi har et par sensor funksjoner også i Controller. SensorConnect som oppretter en serial tilkobling i sensor, og kjører runSensorThread som starter en tråd som kjører sensorInData.

```
public void sensorInData()
{
    while (true)
    {
        var inData = sensor.InData;
        if (inData != null && inData != "" && lastprinted != inData)
        {
            log.logEntry(inData);
            lastprinted = inData;
        }
        Thread.Sleep(10);
    }
}
```

Figur 14: sensorInData funksjonen

Funksjonen "sensorInData" kjører så lenge tråden er aktiv. Den har som oppgave og periodisk lese innholdet i sensor.InData og skrive dette til loggen, så lenge den ikke er lik null, tom eller har det samme innhold som den forrige strengen vi skrev til loggen. For at løkken ikke skal spise alle ressursene tar den pause hver gang den har lest sensor.InData.

Funksjonen runThreadSensor() var på et tidspunkt skrevet som en onPropertyChanged EventListener. Hvor den bare kjørte når det var endringer i strengen sensor.InData. Dette er en anbefalt tilnærming som er ressursbesparende. Denne forbedringen ble gjort, men det oppstod noe feil under testing. Da vi ikke hadde mer tid i budsjettet valgte vi helle og benytte en versjon av sensorkoden som fungerte uten feil.

4.1.5 Robot klassen

Robot klassen inneholder funksjonene som står for nettverkskommunikasjonen med UR 5 robotarmen. Dette fungerer ved at vi setter opp programmet som en server, som lytter etter tilkobling, og robotarmen som en klient som kobler seg til over Ethernet.

```
public bool Connected
{
    get { return connected; }
    set
    {
        if (connected == value)
        {
            return;
        }
        else
        {
            connected = value;
            OnPropertyChanged(new PropertyChangedEventArgs("Connected"));
        }
    }
}
```

Figur 15: get; set; for Connected

For at brukeren enkelt skal få en tilbakemelding på om systemet er tilkoblet har vi implementert en eventlistener som lytter etter endringer i variabelen Connected. Vi lager så et event som kalles fra funksjonen OnPropertyChanged. [8]

```
public event PropertyChangedEventHandler PropertyChanged;
1 reference | Thomas Wegener, 12 days ago | 1 author, 1 change
private void OnPropertyChanged(PropertyChangedEventArgs e)
{
    PropertyChangedEventHandler h = PropertyChanged;
    if (h != null)
    {
        h(this, e);
    }
}
```

Figur 16: OnPropertyChanged funksjonen

Når det finnes en handler og et event er satt fra Connected sin set funksjon så sender vi eventet e. Dette fanges opp av en EventListener som kjører en funksjon i Controller klassen hver gang det er endringer i Connected.

Det neste som skjer er at vi har en funksjon som tar seg av selve tilkoblingen. Her opprettes det en instans av klassen TcpListener som tar imot tilkoblinger fra alle IP adresser på port nummer 30002.


```
public void connectRobot()  
{  
    //create a new TcpListener listening from any ip address on port 30002  
    TcpListener server = new TcpListener(IPAddress.Any, 30002);  
  
    server.Start();  
  
    //listening for a connection  
    while(true)  
    {  
        //client is a object of type TcpListener.accepttcpclient  
        client = server.AcceptTcpClient();  
  
        //make a thread object for listening to messages coming from the client  
        Thread tcpReadThread = new Thread(new ThreadStart(receiveDataRobot));  
  
        //start the thread  
        tcpReadThread.Start();  
    }  
}
```

Figur 17: connectRobot funksjonen

Når vi har opprettet denne instansen kaller vi Start funksjonen i TcpListener som gjør at programmet starter å lytte etter tilkoblinger.

Det er så en evig løkke som går så lenge det lyttes etter tilkoblinger og oppretter ett klient objekt av denne tilkoblingen hvis det skjer.

Når en tilkobling er opprettet så oppretter, og starter vi en ny tråd av receiveDataRobot funksjonen som lytter etter data på nettverkstrømmen.

Denne funksjonen er til for å lytte etter aktivitet på nettverksstrømmen.

Dette foregår ved at vi lager et objekt av typen TcpClient.getstream som henter nettverksstrømmen fra UR 5 robotarmen.

Vi sjekker så om strømmen er tilgjengelig for lesing ved å kalle på Can.read funksjonen til stream objektet, hvis den er det opprettes det er byte array for å holde på informasjonen som kommer. Informasjonen som kommer over nettverket må formateres på en måte som er leselig for mennesker så det opprettes en instans av StringBuilder. Denne informasjonen som kommer fra UR5 robotarmen må også konverteres til ASCII tegnsettet.

Etter at denne delen av løkken er ferdig er det en ny while løkke som sjekker om det er tilgjengelig data på nettverkstrømmen ved å kalle DataAvailable funksjonen i NetworkStream klassen. Det første som skjer i denne løkken er at vi sjekker om antallet bytes som er lest fra strømmen er lik 0, hvis den er det betyr dette at koblingen mellom robot og programmet er brutt og den boolske verdien "Connected" settes til false og vi kaller break for å avslutte hele løkken.


```
//robot sends ok message after completing a movement
if (myCompleteMessage.ToString() == "OK")
{
    //if robot send OK, go to next movement
    nextCommand = true;
}
//robot sends connected when TCP/IP connection is complete
if (myCompleteMessage.ToString() == "Connected")
{
    //variable telling the GUI to show connection status
    Connected = true;
}
```

Figur 18: Utdrag fra receiveDataRobot funksjonen

Denne figuren viser det neste som skjer så lenge det er data tilgjengelig for å bli lest fra nettverkstrømmen. Hvis beskjenen som er mottatt er "OK" så setter vi den boolske verdien nextCommand til sann, denne bestemmer om vi skal gå videre til neste bevegelse i sekvensen som kjøres.

Hvis beskjenen som mottas er "Connected" betyr dette at det er tilkobling til robotarmen og vi kan fortelle brukeren dette ved å skrive det til loggen.

Det er også nødvendig å kunne sende strenger tilbake til roboten, dette gjøres ved at det er en funksjon som heter sendMsgRobot som tar inn en streng som argument.

```
//this function takes in any string as a parameter and sends it to the robot
2 references | oaax, 5 days ago | 2 authors, 7 changes
public void sendMsgRobot(string msg)
{
    //if the client is null, that means the connection to the robot is severed
    if (client == null)
        return;
    //create a instance of the networkstream from the client object
    stream = client.GetStream();
    //byte array with ascii encoding
    byte[] Mesg = System.Text.Encoding.ASCII.GetBytes(msg);

    //call the write function of NetworkStream with our message as argument
    stream.Write(Mesg, 0, Mesg.Length);
}
```

Figur 19: sendMsgRobot funksjonen

Her lager vi et objekt av typen Getstream som er en funksjon i Tcpclient klassen. Denne returnerer nettverksstrømmen til det valgte objektet som vi kan kalle Write funksjonen til for å skrive til roboten over Ethernet.

Vi har valgt å bruke en if sjekk for å sjekke om client objektet er lik null, hvis det er det så betyr det at vi ikke har noen tilkobling til roboten lenger og funksjonen avsluttes. Dette gjøres for å unngå NullReferenceExceptions.

Vi oppretter et byte array med ASCII encoding da write funksjonen trenger et byte array som argument. [9]

4.1.6 Log klassen

Logg klassen har til hensikt å lagre strenger i en loggfil. Først settes navnet på den innebygde loggfilen, slik at denne er definert et sted og ikke trenger og defineres hver gang vi skal lese, skrive til den, eller overskrive innholdet i filen. Dette gjør vi ved å definere en public const string som vi kaller LogFile, og gir den verdien @"log.txt".

```
public void logEntry(string msg)
{
    // To enqueue the write, we use the built in threadpool.
    ThreadPool.QueueUserWorkItem(writeToFile, msg);
}
```

Figur 20: logEntry funksjonen

Så har vi laget en funksjon som tar en string og putter den i loggen. Denne funksjonen heter logEntry. Hvis logEntry funksjonen får skrive direkte til filen så får vi problemer med å lese filen, og/eller vi kan få problemer med å skrive når vi leser. Derfor legger logEntry bare strengen i en tråd kø med en beskjed om at den skal kjøres i funksjonen writeToFile.

Videre oppretter vi en låsefunksjon som låser filen når køoperasjonene kjøres. På denne måten forhindrer vi at det blir krøll med andre tråder eller operasjoner i systemet. Låsen opprettes som et objekt og sendes som parameter i writeToFile sammen med teksten vi skal putte i loggen.


```
private static object writeLock = new object();
1 reference | 0 changes | 0 authors, 0 changes
public static void writeToFile(object msg)
{
    lock (writeLock)
    {
        //using the lock bject we now can ensure tha we are the only tread accessing the file
        if ((string)msg == "reset") //the reset string flushes the logfile.
        {
            var strm = File.CreateText(LogFile);
            strm.Flush();
            strm.Close();
        }
        else
        {
            using (var writer = File.AppendText(LogFile)) //adds a string to a new line in the log
            {
                writer.WriteLine((string)msg); //writes the string.
            }
        }
    }
}
```

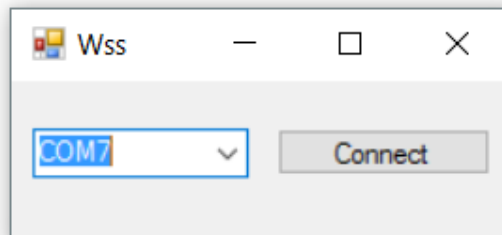
Figur 21: writeToFile funksjonen

Vi har laget unntak som sier at; hvis strengen som sendes til loggen er reset, så skal vi flushe filen. Vi bruker StreamWriter fra IO biblioteket til denne oppgaven. Hvis filen ikke finnes så oppretter vi en ny fil. Hvis strengen er noe annet enn reset så legger vi til strengens verdi som en ny linje i filen. Har bruker vi using og lager en streamwriter som kjører funksjonen File.AppendText fra IO biblioteket, den tar "hvilken fil" som parameter. Vi bruker streamwriters innebygde WriteLine som tar strengen som parameter, for å skrive strengen til filen.

Det er knyttet en eventlister til denne hendelsen, som sier at hver gang det gjøres en endring i loggfilen så skal filen skrives til brukervinduet. Dette skjer i View klassen og er et bedre alternativ enn å kjøre en tråd som leser periodisk.

4.1.7 Sensor klassen

Sensorklassen inneholder funksjoner som blir kjørt i en tråd som opprettes i Controller klassen. Sensoren kobles til systemet ved å opprette en seriellport som håndterer data som kommer over Bluetooth. Seriellporten åpnes ved at brukeren velger COM port og trykker på koble til i et lite vindu som åpnes fra brukergrensesnittet.



Figur 22: Brukergrensesnitt for tilkobling til sensor

Baudrate og time-out er hardkodet, for å være den samme som på sensoren. Vi har vurdert det slik at brukeren ikke skal få flere valg en nødvendig for å effektivisere konfigurasjonsprosessen. Brukeren får dermed heller ikke gjøre ting vi ikke har testet.

```
public void connect(string port) //connect function
{
    blueTooth.PortName = port;
    blueTooth.BaudRate = 9600;
    blueTooth.DataReceived += new SerialDataReceivedEventHandler(DataReceivedHandler);
    blueTooth.WriteTimeout = 500;
    blueTooth.Open();
    resetSensor();
}
```

Figur 23: connect funksjon for å koble til sensor

Først definerer vi en variabel som holder styr på om vi er tilkoblet. En streng som holder verdier sendt fra serial porten. Vi oppretter et Seriellport objekt. I connect-funksjonen settes nødvendige verdier for en tilkobling og porten åpnes. Det første vi gjør på en ny tilkobling er å kjøre en reset funksjon som sender en "r" over seriellporten. Når sensoren mottar denne strengen nullstilles sensoren, og den fortsetter å sende "#" til oss. Her var det en tanke at vi skulle implementere en heartbeat som holder styr på hvor lenge det er siden forrige "#", men det fikk vi ikke tid til.

Vi bruker en eventhandler for å håndtere hva som skal gjøres når vi mottar noe fra seriellporten. Hver gang vi får noe på bufferet så kjøres en readline funksjon, som skriver verdien til en streng "indata", denne strengen analyseres med et sett med if setninger.


```
SerialPort sp = (SerialPort)sender;
string indata = sp.ReadLine();
if (indata != null && indata != "")
{
    if (indata.Contains('0'))
    {
        InData = "Sensor has been reset.";
    }
    else if (indata.Contains('#') && !con)
    {
        InData = "Sensor connected";
        con = true;
    }
    else if (indata.Contains('F'))
    {
        InData = "Weapon has been fired!";
    }
}
```

Figur 24: Handler ut ifra mottatte data fra sensor

Er den null eller tom så gjør vi ingenting.

Hvis den inneholder en '0', betyr det at sensoren er nullstilt.

Hvis den inneholder en '#', så er vi tilkoblet, og så sier vi at variabelen con er sann.

Hvis den inneholder en 'F', betyr det at sensoren har registrert en firing.

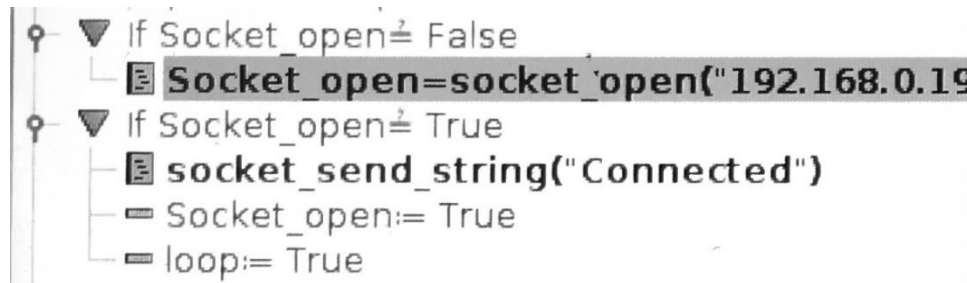
Vi bruker indata.contains for å håndtere tilfeller der bufferet holder på flere verdier og ikke klarer å skille dem fra hverandre.

Disse forskjellige hendelsene skrives til strengen InData. En tråd som kjører sensorovervåkingen fra klassen Controller leser InData periodisk, den ser om vi har skrevet InData til loggen før. Og hvis InData ikke er identisk med det siste vi skrev ut; så skriver vi verdien til InData ut til loggen.

4.2 UR5 kode

For å kunne kommunisere mellom UR 5 og test programvare er det nødvendig å ha ett program som fungerer som klient kjørende på roboten.

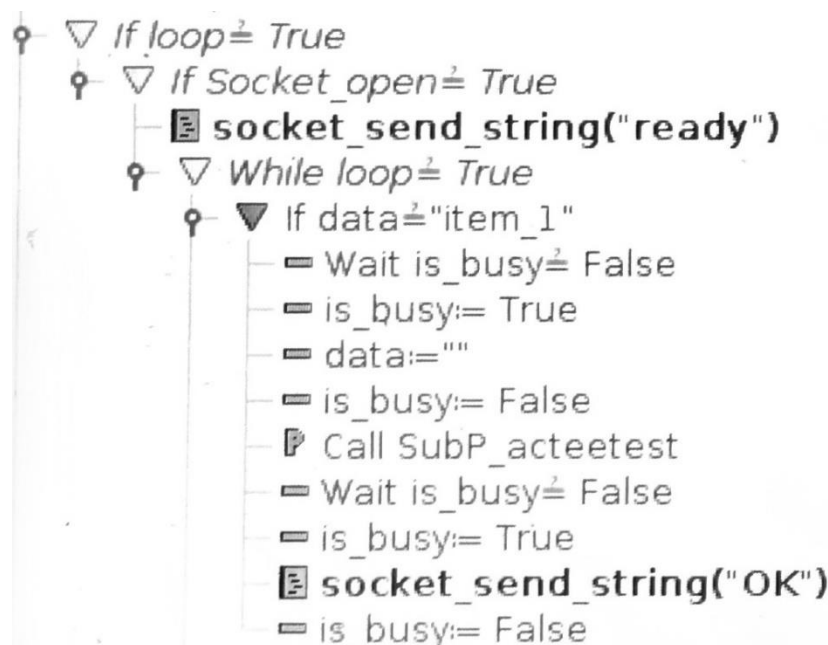
Dette programmet står først i en løkke som prøver å opprette en tilkobling til test programmet ved å åpne en socket, hvis den ikke finner en server på nettverket vil den vente i et halvt sekund før den forsøker igjen.



Figur 25: Nettverkstilkobling fra UR 5

Denne figuren viser hvordan koden på UR 5 kjører scriptkoden `socket_open` for å skape en tilkobling til testprogrammvaren. Når denne tilkoblingen er opprettet vil den sende en "Connected" string over socketen som vil skrives til loggen i test programmet slik at brukeren vet at tilkobling har skjedd.

Etter at tilkobling har skjedd vil det startes en løkke som lytter etter strenger sendt fra test programmet. Ut ifra hvilken streng som sendes vil det kjøres en sekvens på robotarmen.



Figur 26: Løkke som kjøres om strengen "item_1" mottas

Disse bevegelsessekvensene ligger lagret i subprogrammer inne i programmet på robotarmen. Dette er programmer som består av en serie med bevegelser som roboten skal utføre for å f.eks. Kunne trykke på knappene på DCP. Som figur 26 viser vil roboten kjøre sekvensen med navn item_4 når den mottar den beskjeden som en string fra test programmet.

Det kjøres en tråd på robotarmen for å lytte etter beskjeder sendt over nettverket.



Figur 27: Lyttetråd på UR 5

Den må kjøre som en tråd for at vi kan få samtidighet i programmet på UR 5 robotarmen. Denne lytter etter data fra test programmet og hvis data den mottar ikke er lik null, vil den skrive denne verdien til variabelen "data" som vi bruker andre steder i programmet for å bestemme hvilken bevegelse som skal gjøres. Her er det også en stopp funksjon for programmet som lytter etter strengen "stop", og hvis den mottar denne vil den kalle funksjonen halt, som stopper programmet på robotarmen.

Når det skal lages en sekvens for å f.eks. gripe joysticken som styrer våpentårnet må vi også styre gripeverktøyet, dette gjøres ved å sende signaler til Arduino med UR 5 sine digitale ut porter nr. 8 og 9. Vi kan da sende 4 bits koder ved å sette digital out til true eller false fra koden.


```
SubP_grip_joystick
  set_digital_out(8, True )
  Wait: 0.1
  set_digital_out(9, True )
  Wait: 0.1
  set_digital_out(8, False )
  Wait: 0.1
  set_digital_out(9, False )
  Wait: 0.1
  set_digital_out(8, True )
  Wait: 0.1
  set_digital_out(9, True )
  Wait: 0.1
  set_digital_out(8, True )
  Wait: 0.1
  set_digital_out(9, False )
  Wait: 0.1
```

Figur 28: Subprogram for å gripe joystick

Her er ett eksempel på et subprogram som vi gripe joysticken. For hver gang vi setter den digitale ut porten fra koden legger vi inn en Wait for å gi Arduionen tid til å oppfatte signalet.

For å forhindre at det skjer feil under utførelsen av bevegelsen må det legges inn en Wait funksjon, som venter et gitt antall sekunder før den går videre i koden. Dette for å forhindre at roboten begynner å bevege seg mens hånden beveger seg, som kan resultere i feil som f.eks. at robotarmen kommer frem til joysticken før gripeverktøyet har rukket å åpne seg. Disse må legges inn manuelt ettersom vi ser hvor lang tid de forskjellige bevegelsene til gripeverktøyet tar.

Koden på UR 5 er en videreutvikling av koden det foregående prosjektet, ACT[5] lagde. Hvor videreutvikling består av endringer av noe av strukturen og at det er implementert bruk av digitale ut signaler for å styre gripeverktøyet.

5 Konklusjon

Utvikling av software til ACTEE systemet har vært en effektiv og engasjerende prosess. Vi på software har brukt mye tid på analyse av systemet i forkant av implementeringen. Man kan kanskje si at designfasen vår ble unaturlig lang i forhold til implementeringen, men vi har utnyttet mulighetene og benyttet tid med venting til andre ting som gavnet prosjektet. Det totale systemet har mange avhengigheter, og samarbeidet mellom de ulike disiplinene har fungert godt.

C# Har vært et naturlig valg da vi skulle fortsette et prosjekt skrevet i C#. Det har vært mindre utfordrende i forhold til syntaks en vi hadde tenkt. Tilgangen på god dokumentasjon har vært avgjørende for vår suksess i implementeringsfasen. Vi to på software har kommunisert veldig godt, vi har jobbet mye sammen, og det at vi har brukt verktøy som GIT og SourceTree har gjort at vi enkelt har kunnet holde styr på fletting av og samarbeid med koden.

Vi ser at vi gjerne skulle hatt lengre tid til å rydde opp og polere koden, noe vi også kommer til å gjøre etter levering av dokumentasjon. Vi har lært mye underveis som å implementere eventlistenere og byttet ut tråder med nyere mer ressursbesparende funksjonalitet i C#, men det er flere funksjoner i koden vi ikke har fått tid til å endre. Vi kommer med forslag for forbedringer vi ønsket å gjøre, men ikke fikk tid til i fremtidsdokumentet.

Kilder

[1] Universal Robots

«The UR Script Programming Language»

Tilgjengelig fra:

http://www.wmv-robotics.de/home_htm_files/scriptmanual_en_1.5.pdf

Sist besøkt: 02.05.2016

[2] Universal Robots

“Polyscope Manual”

Version 1.7, 2012, side 5.

Tilgjengelig fra:

http://support.universal-robots.com/foswiki/pub/Downloads/SoftwareUpdates/software_manual_en_Global_1.7beta.pdf

Sist besøkt: 03.05.2016

[3] Microsoft Developer Network

“System.Xml.Linq Namespace”

[Nettside]

<https://msdn.microsoft.com/en-us/library/system.xml.linq%28v=vs.110%29.aspx?f=255&MSPPErr=-2147217396>

Sist besøkt: 17.05.2016

[4] Microsoft Developer Network

“TcpListener Class”

[Nettside]

[https://msdn.microsoft.com/en-us/library/system.net.sockets.tcplister\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/system.net.sockets.tcplister(v=vs.110).aspx)

Sist besøkt: 17.05.2016

[5] A.K. Svendsen; E.L. Roa; H.B. Sørums; S. Rudin. "Automated CROWS Testing"

Bachelor Thesis, Fakultet for teknologi og maritime fag, Høgskolen i Buskerud og Vestfold, Kongsberg, 2014.

Tilgjengelig fra:

<http://hdl.handle.net/11250/216910>

Sist besøkt: 16.03.2016

[6] Wikipedia

Git (software)

[Nettside]

[https://www.wikipedia.org/wiki/Git_\(software\)](https://www.wikipedia.org/wiki/Git_(software))

Sist besøkt 18.05.2016

[7] iOS Developer Library

“Model-View-Controller”

[Nettside]

<https://developer.apple.com/library/ios/documentation/General/Conceptual/DevPedia-CocoaCore/MVC.html>

Sist besøkt: 18.05.2016

[8] Microsoft Developer Network

“iNotifyOnPropertyChanged”

[Nettside]

[https://msdn.microsoft.com/en-us/library/ms743695\(v=vs.100\).aspx](https://msdn.microsoft.com/en-us/library/ms743695(v=vs.100).aspx)

Sist besøkt: 17.05.2016

[9] Microsoft Developer Network

“NetworkStream.Write Method”

[Nettside]

<https://msdn.microsoft.com/en-us/library/system.net.sockets.networkstream.write%28v=vs.110%29.aspx?f=255&MSPPErr=-2147217396>

Sist besøkt: 05.05.2016

Vedlegg A: Versjonshistorie fra SourceTree.

	master origin/master Merge branch 'master' of ssh://home.wegener.no	17 mai 2016 11.59	Thomas Wegener <thomas@wegener.no>
	Bare no småtteri	17 mai 2016 11.58	Thomas Wegener <thomas@wegener.no>
	changed comment	17 mai 2016 11.58	oaax <oaax@hotmail.com>
	Fixed typos and using new functions instead of both old and new.	17 mai 2016 10.41	Thomas Wegener <thomas@wegener.no>
	Added some quick help, about text, implemented reset button.	16 mai 2016 13.34	oaax <oaax@hotmail.com>
	Jag har addad Online User Manual. och en helvettes masse kommentarier i kådet	16 mai 2016 12.51	Thomas Wegener <thomas@wegener.no>
	no message	16 mai 2016 12.07	Thomas Wegener <thomas@wegener.no>
	changed text of connection status	14 mai 2016 10.54	oaax <oaax@hotmail.com>
	renamed a button in XMLedit form	14 mai 2016 10.52	oaax <oaax@hotmail.com>
	added delete functio to lsiobox in testeditor, works by pressing delete	13 mai 2016 17.31	oaax <oaax@hotmail.com>
	Merge branch 'master' of ssh://git.actee.no:svr/git/project/actee	10 mai 2016 16.59	oaax <oaax@hotmail.com>
	minor changes, mostly comments	10 mai 2016 16.59	oaax <oaax@hotmail.com>
	Sensorhacking.. It works now.. Moved most of	5 mai 2016 14.34	Thomas Wegener <thomas@wegener.no>
	Merge branch 'master' of ssh://git.actee.no:svr/git/project/actee	4 mai 2016 21.23	oaax <oaax@hotmail.com>
	Merge remote-tracking branch 'origin/master'	4 mai 2016 21.18	Thomas Wegener <thomas@wegener.no>
	Fiksa event listner??	4 mai 2016 21.18	Thomas Wegener <thomas@wegener.no>
	changed one line	4 mai 2016 15.01	oaax <oaax@hotmail.com>
	comments and cleaning	4 mai 2016 14.34	oaax <oaax@hotmail.com>
	cleaned and commented in the robot class	4 mai 2016 11.33	oaax <oaax@hotmail.com>
	removed some unused functions	4 mai 2016 11.13	oaax <oaax@hotmail.com>
	Fixing	4 mai 2016 10.54	Thomas Wegener <thomas@wegener.no>
	Har oppdatert nesten alle filer. Fiksa en git ignorefil så vi ikke trenger og pushe alle fil	3 mai 2016 23.39	Thomas Wegener <thomas@wegener.no>
	removing binary files and ignoring extra stuff	3 mai 2016 21.40	Thomas Wegener <thomas@wegener.no>
	Merge branch 'master' of ssh://home.wegener.no:svr/git/project/actee	3 mai 2016 21.32	Thomas Wegener <thomas@wegener.no>
	LOTS OF UPDATES	3 mai 2016 21.24	Thomas Wegener <thomas@wegener.no>
	changed logfile location to the actee folder	3 mai 2016 15.01	oaax <oaax@hotmail.com>
	Fiksa reset log	2 mai 2016 16.11	Thomas Wegener <thomas@wegener.no>
	Cleaned some code, added functionality to the reset button.	2 mai 2016 14.05	oaax <oaax@hotmail.com>
	Cleaned View.cs , made the connect toolbar button be disabled after click.	2 mai 2016 10.38	oaax <oaax@hotmail.com>
	small changes, not much was accomplished this day...	29 apr 2016 15.23	oaax <oaax@hotmail.com>
	Sensor implemented selecting only shots fired. progressbar replaced by a connection	29 apr 2016 0.32	Thomas Wegener <thomas@wegener.no>
	Merge branch 'master' of ssh://home.wegener.no:svr/git/project/actee	28 apr 2016 21.54	Thomas Wegener <thomas@wegener.no>
	sensor.cs wss.cs har small updates.. merging with master.	28 apr 2016 21.53	Thomas Wegener <thomas@wegener.no>
	Cleaned code. Commented more. Added stop button and checked while running sequ	28 apr 2016 14.46	oaax <oaax@hotmail.com>
	Implemented some write functionality to the log. It will get testname and say when th	26 apr 2016 14.55	oaax <oaax@hotmail.com>
	Fixed alot of bugs. Progressbar replace square. Two threads may now write to same fil	26 apr 2016 7.49	Thomas Wegener <thomas@wegener.no>
	Connection status	25 apr 2016 15.30	oaax <oaax@hotmail.com>
	Adding content to sensors.cs	25 apr 2016 15.28	Thomas Wegener <thomas@wegener.no>
	Merge remote-tracking branch 'origin/master'	25 apr 2016 14.19	Thomas Wegener <thomas@wegener.no>
	no message	25 apr 2016 14.19	Thomas Wegener <thomas@wegener.no>
	templog	25 apr 2016 13.32	oaax <oaax@hotmail.com>
	fixing errors??	25 apr 2016 13.29	Thomas Wegener <thomas@wegener.no>
	no message	25 apr 2016 13.22	Thomas Wegener <thomas@wegener.no>
	Lagt til logsystemet.. Endret på gui. Leser fra templogfile.	25 apr 2016 13.09	Thomas Wegener <thomas@wegener.no>
	changed names on listbox variables for sending to ur 5	25 apr 2016 10.10	oaax <oaax@hotmail.com>
	Updated textlogger	25 apr 2016 9.28	Thomas Wegener <thomas@wegener.no>
	Cleaned and added new	21 apr 2016 13.22	oaax <oaax@hotmail.com>
	updated textlogger	21 apr 2016 13.11	Thomas Wegener <thomas@wegener.no>
	Dynamic resizing of testlogger complete.	20 apr 2016 11.32	Thomas Wegener <thomas@wegener.no>
	Laster opp nyeste versjon med sending av strings over nettverk	20 apr 2016 9.49	oaax <oaax@hotmail.com>
	Lagt ut det som lå i google driven 19.04. 08:00 og lagt til textlogger. Dette er nå mitt i	19 apr 2016 8.00	Thomas Wegener <thomas@wegener.no>
	Merge branch 'master' of ssh://home.wegener.no:svr/git/project/actee	18 apr 2016 23.28	Thomas Wegener <thomas@wegener.no>
	lagt til et bilde	18 apr 2016 23.21	Thomas Wegener <thomas@wegener.no>
	laqt til et bilde	18 apr 2016 23.16	Thomas Wegener <thomas@wegener.no>

Teknologi dokument WSS

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAAGSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	TEKNOLOGI DOKUMENT WSS
VERSJON / DATO	1.0 / 19. mai 2016
FORFATTER	Bård Simen Hamborg Enget
SIDEANTALL	24



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



i. Abstrakt

Dette dokumentet handler om WSS (Weapon Station Sensor), som er en del av ACTEE sitt bachelor prosjekt Vår 2016. Vi vil her gå igjennom valgene som har blitt gjort underveis, teknologiene vi har brukt og funksjonaliteten til enheten.

ii. Innholdsfortegnelse

i. Abstrakt.....	1
ii. Innholdsfortegnelse	2
iii. Figur oversikt	3
iv. Tabell oversikt	3
v. Dokumenthistorie	4
vi. Forkortelser	5
1 Innledning	6
2 WSS	7
2.1 Bakgrunns teori	7
2.1.1 Arduino	7
2.1.1.1 Arduino Nano.....	7
2.1.1.2 Arduino Uno.....	8
2.1.2 Komparator	8
2.1.2.1 MAX931CPA+	8
2.2 Funksjonalitet	9
2.2.1 Fyringskretsen	9
2.2.2 Våpen ID krets	12
2.2.3 Blåtann modul	13
2.2.4 Arduino	14
2.2.4.1 Arduino tilkoblinger	14
2.2.4.2 Arduino Scetch	17
2.2.4.2.1 Bibliotek og variabler	17
2.2.4.2.2 SoftwareSerial.....	17
2.2.4.2.3 void setup()	18
2.2.4.2.4 void fire()	18
2.2.4.2.5 void loop()	19
2.2.4.2.6 void setWid()	21
3 Konklusjon	22
4 Referanser	23

iii. Figur oversikt

Figur 1: Arduino Nano V3.1	7
Figur 2: Arduino Uno	8
Figur 3: Op-Amp komparator.....	8
Figur 4: MAX931CPA+ typisk bruk	8
Figur 5: Wss fyringskrets.....	10
Figur 6: WSS fyringskrets simulering 32V	10
Figur 7: WSS fyringskrets simulering 8V	11
Figur 8: Kobling Arduino mot WS	12
Figur 9: HC-06 Blåtann modul.....	13
Figur 10: Tilkobling Arduino mot Blåtann modul	13
Figur 11: Fyringskrets	15
Figur 12: Bibliotek og variabler	17
Figur 13: SoftwareSerial.....	17
Figur 14: void setup()	18
Figur 15: void fire()	18
Figur 16: void loop()	19
Figur 17: void setWid()	21

iv. Tabell oversikt

Tabell 1: Dokumenthistorie.....	4
Tabell 2: Forkortelser	5
Tabell 3: Tilkoblinger Arduino Nano	16
Tabell 4: Bill of materials WSS	16

v. Dokumenthistorie

Tabell 1: Dokumenthistorie

Versjon 0.1	10.05.16	Oppsett av Bård Simen Bakgrunns teori skrever av Bård Simen
Versjon 0.2	11.05.16	Innledning skrevet om av Bård Simen Påbegynt Hoveddel av Bård Simen
Versjon 0.3	13.05.16	Hoveddel fylt ut av Bård Simen
Versjon 0.4	15.05.16	Hoveddel utfylt av Bård Simen
Versjon 0.5	16.05.16	Espen: Korrektur.
Versjon 0.6	18.05.16	Oppdatert kapittel om void fire() av Bård Simen
Versjon 1.0	19.05.16	Espen: Ordnet en figur og ferdigsjekk.

vi. Forkortelser

Tabell 2: Forkortelser

Forkortelser	Forklaring
ACT [1]	Automated CROWS Testing (foregående bachelorgruppe)
ACTEE	Automated CROWS Testing End Effector
CROWS	Common Remotely Operated Weapon Station
UR5 [2]	Universal Robots 5 robotarm
IDE	Integrated Development Enviroment
WSS	Weapon Station Sensor

1 Innledning

Ved prosjektets start så vi tidlig at det ville være et behov for deteksjon av enkelte hendelser på WS under automatisk testing. Vi så også at det ville være viktig at dette foregikk så uavhengig av WS sine indre funksjoner som mulig.

WSS ble derfor utviklet for å kunne oppfylle dette formålet. Første mål for sensoren var muligheten til å detektere et høyt trigger signal og kunne rapportere dette tilbake til test software uten noen påvirkning av WS Software. Ved å holde disse separert er tanken at vi vil kunne rapportere alle avfyringer på WS, selv eventuelle utilsiktede avfyringer, og med dette øke validiteten av testingen. På denne måten vil et loggsystem kunne verifisere at avfyringer skjer når dette er en del av testen, samtidig som det kan rapportere avfyringer som skjer som ikke er en del av testen.

Vi vil i dette dokumentet forklare nærmere om mikrokontrolleren vi har brukt til formålet, kretsen vi har designet for deteksjon av fyrings signal, tilkoblingen mot både WS og testsoftware og koden skrevet på mikrokontrolleren for å muliggjøre dette.

2 WSS

WSS er en sensor som plasseres på en CROWS M153 våpenstasjon og detekterer om det skjer firing. Hvis det er tilfelle blir denne informasjonen sendt videre til test software via Blåtann kommunikasjon.

Dokumentets hoveddel vil bli todelt, først vil vi ta for oss bakgrunns teori for bedre forståelse av de individuelle delene av WSS, før vi beveger oss over til hvordan enheten faktisk fungerer.

2.1 Bakgrunns teori

I dette kapitlet tar vi for oss den viktigste bakgrunns kunnskapen leseren behøver for å forstå det tekniske innholdet.

2.1.1 Arduino

Arduino[3] er en åpen kilde plattform laget for enkel styring av elektroniske prosjekter. Ideen ble tatt fra Master oppgaven Wiring fra 2004 som i korte trekk handlet om å få tilgjengeliggjort mikrokontrollere som et verktøy for hobby utviklere. Tanken var å tilgjengeliggjøre teknologien for allmenheten med å øke tilgjengeligheten og senke kostnaden.

Prosjektet tilbyr en hel rekke forskjellige ferdige kort med forskjellige mikrokontrollere, sammen med en gratis IDE (Integrated Development Environment) som støtter Processing, C og C++ kode. Arduino IDE tilbyr også en god del eksempelkode gratis, som fritt kan brukes i egne prosjekter. Det har også vokst frem et relativt stort miljø med DIY utviklere rundt dette prosjektet som gladelig deler kode, biblioteker og prosjekter på nett.

2.1.1.1 Arduino Nano

Arduino Nano[4] er ett av ferdigkortene man kan bruke, det er flere versjoner av dette kortet, men de siste kommer med en Atmega328 mikrokontroller.



Figur 1: Arduino Nano V3.1

Med sin størrelse på 45mm x 18mm er den godt egnet til små prosjekter. Begrensningen kommer naturlig fra antall tilkoblingsmuligheter du har, men med 14 digitale I/O, og 8 analoge innganger har man allikevel mange muligheter. Kortet kan kjøre på strøm fra micro-USB tilkoblingen, eller eksternt spenningskilde via Vin pin på 6-20V.

2.1.1.2 Arduino Uno

Et annet ferdig kort fra Arduino er Uno[5], dette er gjerne standard kortet for nybegynnere, da det er svært robust. Designet muliggjør også at om man er så uheldig å ødelegge mikrokontrolleren, kan man bare kjøpe en ny, da dette er en utbyttbar Integrated Circuit.



Figur 2: Arduino Uno

Formfaktoren er noe større på dette kortet med sine 68.6mm x 53.4mm.

Kortet har samme antall digitale I/O porter som Nano, men noen færre analoge, med bare 6.

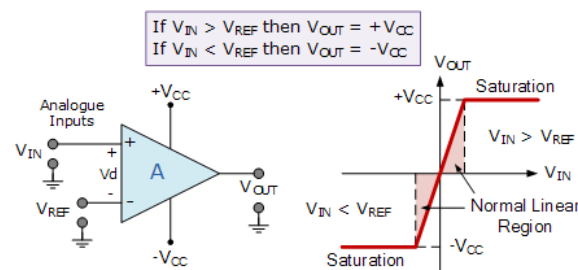
2.1.2 Komparator

En Komparator[6] er en krets bygget med hjelp av en Op-Amp. Kretsen benytter seg av Op-Ampens høye forsterkning til raskt å sette V_{out} til enten $+V_{cc}$, eller $-V_{cc}$.

Ved å sammenligne amplituden til V_{in} mot V_{ref} gjør kretsen en vurdering over hvilke av de to som er størst. Om V_{in} er større enn V_{ref} øker utgangen raskt helt til den går i metning på et nivå som teoretisk tilsvarer $+V_{cc}$. I

virkeligheten vil denne ligge litt lavere, men nært nok til at det ikke spiller noen stor rolle. Om det er V_{ref} som er størst, vil den raskt minke til ca. $-V_{cc}$.

Ved å variere på V_{ref} kan vi med dette bestemme på hvilket nivå av V_{in} kretsen skal skifte mellom $+V_{cc}$, eller $-V_{cc}$.

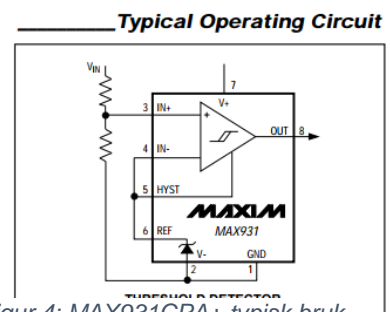


Figur 3: Op-Amp komparator

2.1.2.1 MAX931CPA+

MAX931CPA+[7] er et eksempel på en ferdig komparator IC. Istedenfor å måtte designe kretsen rundt en Op-Amp, kan man bruke denne.

Medfølgende datablad gir gode eksempler på kretsen i bruk og forslag til måter å bruke den. Figur 4 viser et eksempel på dette, det vi ser er en typisk komparator krets som bruker brikkens interne V_{ref} på 1.182V.



Figur 4: MAX931CPA+ typisk bruk

Brikken håndterer single supply, dvs. vi kan da bruke $V_+ = 5V$, $V_- = GND$.

2.2 Funksjonalitet

WSS vil være en komponent i et større system for automatisk testing, den vil være festet på WS, og koblet til connectoren der fyrings solenoiden tidligere sto. Hensikten med komponenten er å detektere når trigger signalet på WS går høyt, redusere dette til et signal en Arduino kan håndtere og logge dette på Arduino. Deretter vil den rapportere denne avfiringen trådløst tilbake til software trådløst via Blåtann. For å muliggjøre dette må enheten også kunne gi input til WS, nærmere bestemt en Våpen ID. Vi vil her ta for oss disse funksjonene.

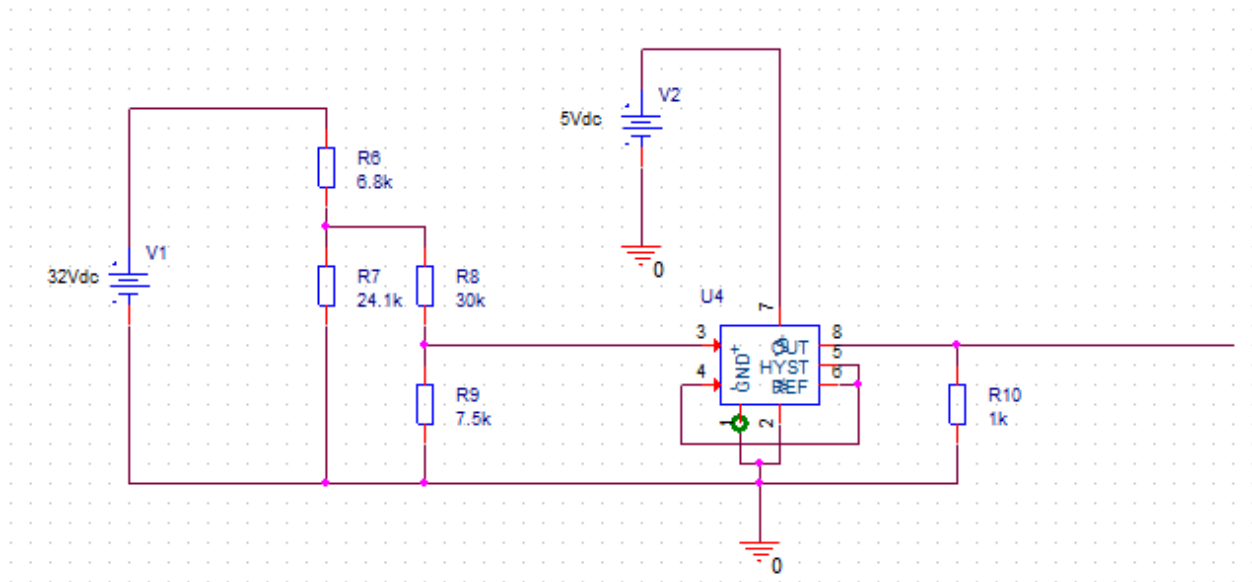
2.2.1 Fyringskretsen

Fyringssignalet fra WS kan variere over et nokså bredt spekter. Amplituden på signalet kan variere fra 18V til 32V avhengig av hvilken strømkilde den er koblet til. Dette ga oss en liten utfordring når det kom til design av fyringskretsen for tolking av dette signalet, da Arduino har en innebygget threshold på ca. 3V for å kunne trigge interrupt () funksjon.

Ved utregning og simulering i OrCad fant vi at en enkel spenningsdeler kunne dempe amplituden ned til 4.4-2,4V. 2.4V er som nevnt litt under Arduino sin threshold for deteksjon av et høyt signal, og dette alene vil derfor ikke være en god nok løsning.

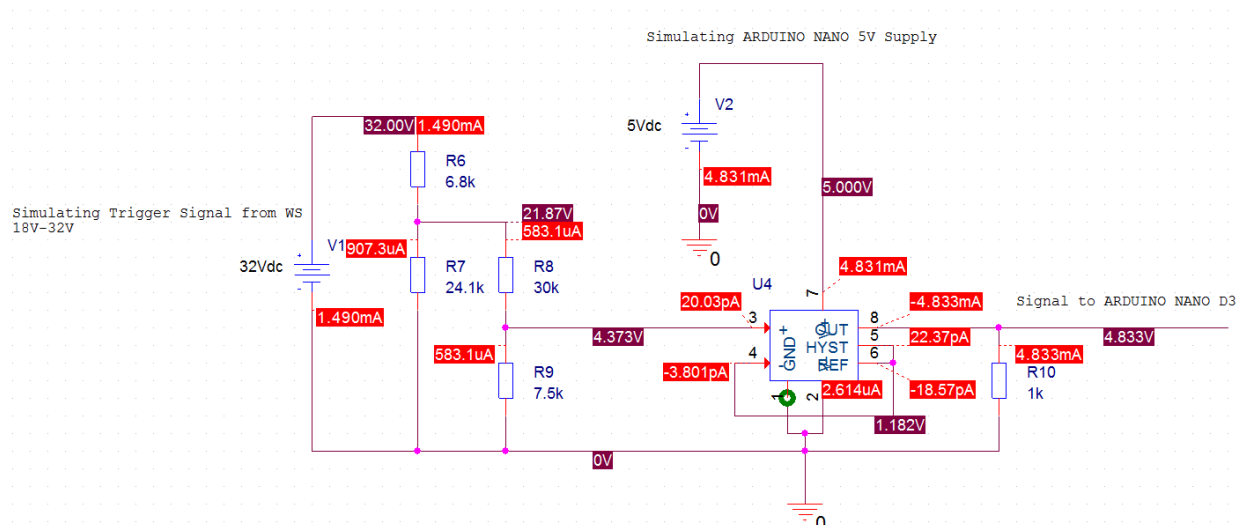
Komparatoren i kretsen ble da introdusert. MAX931CPA+ sine egenskaper med støtte for 5V single supply og innebygget Vref på 1.182V var ideell for våre behov. Dette gjøre at alle signaler større enn 1.182V vil resultere i et høyt signal på 4.833V inn på Arduino sin interrupt pin.

Figur 5 viser en oversikt over fyringskretsen sin oppbygging, her med faste DC supply for simulering av funksjonalitet.



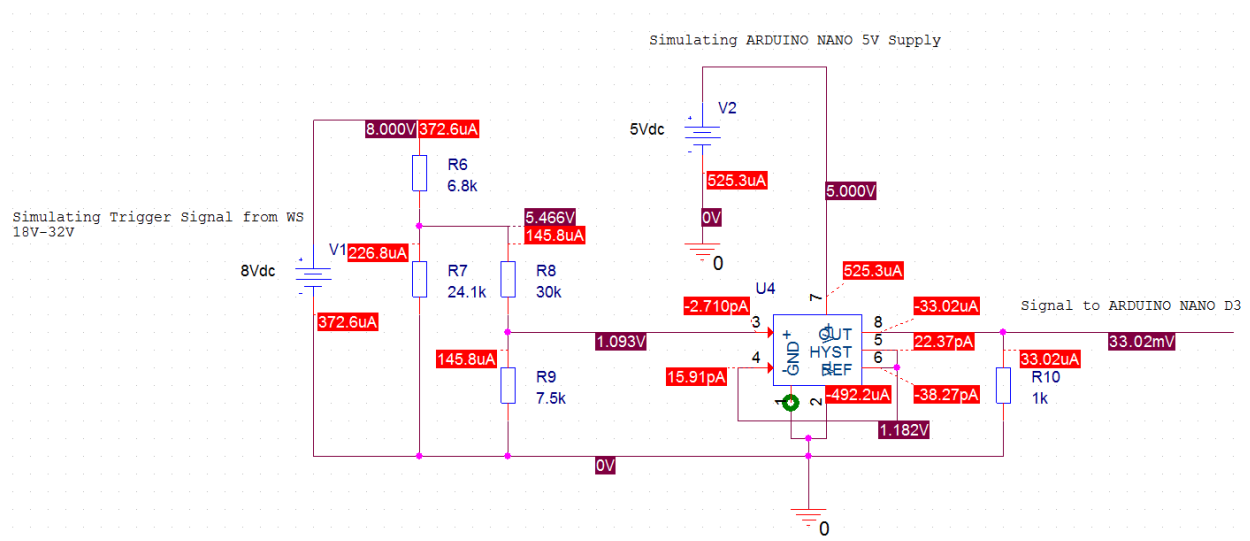
Figur 5: Wss fyringskrets

Simuleringer viser at signalet fra WS kan variere mellom 32V og 9V uten at det vil utgjøre noe forskjell for fyringskretsens funksjonalitet.



Figur 6: WSS fyringskrets simulering 32V

Ved 8V vil inngangen på komparatoren ligge så lavt at den kommer under thresholden, og komparatoren vil da gi et lavt signal ut. Dette kan tilpasses med å endre spenningen inn på pin 4 på komparatoren, men vi gjorde vurderingen at 1.182V var en helt akseptabel threshold å bruke. Figur 7 viser kretsens håndtering av trigger signal lavere enn 9V.



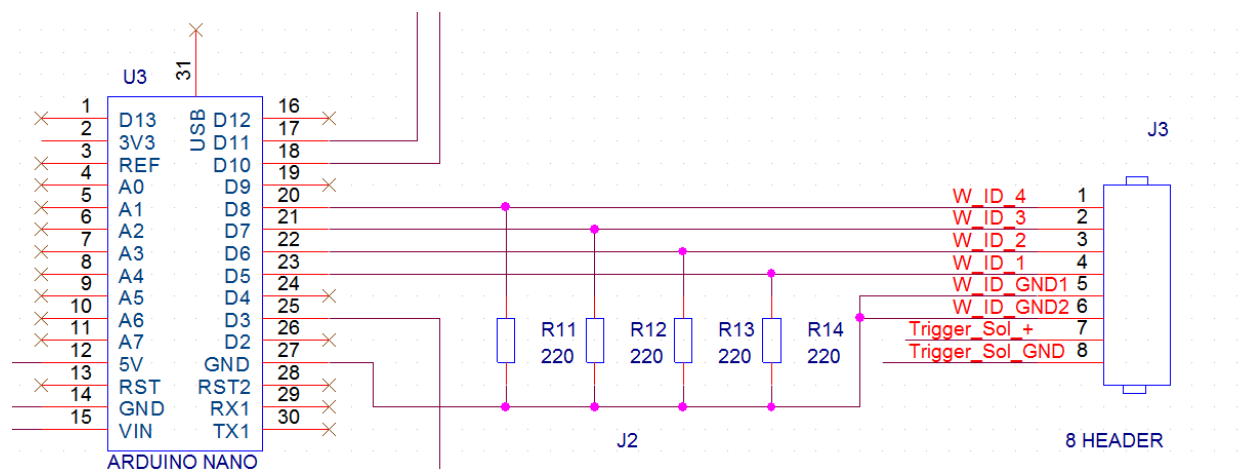
Figur 7: WSS fyringskrets simulering 8V

2.2.2 Våpen ID krets

Innebygget sikkerhet på WS forhindrer at avfyriings signal blir sendt uten at WS har fått en bekreftet Weapon_ID. Siden WS leser denne Weapon_ID' en fra den samme connectoren fyringssensor er koblet til for å detektere avfyringer må WSS også supplere WS med denne.

Software på WS leser Weapon_ID ut fra tilstandene på et sett med GPIO pinner på den interne FPGA'en. Det er 16 mulige tilstander som kan sendes inn på disse pinnene 0-15, eller 0-F. Software leser dette binært fra pinnene, der Weapon_ID_1 fungerer som LSB, og Weapon_ID_4 som MSB, kobling mot GND på disse pinnen vil detekteres av WS Software som en binær 1. Avlesning på disse vil dermed gi et binært tall mellom 0000-1111 som WS Software konverterer til en hexadesimal verdi mellom 0-F.

Arduino sine digitale utganger 5-8 kobles mot connectoren på pinnene til Weapon_Id_1 – 4 som vist i figur 8.



Figur 8: Kobling Arduino mot WS

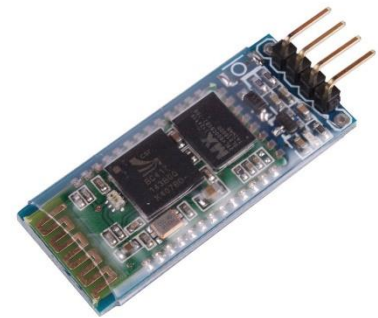
2.2.3 Blåtann modul

Da WSS vil være direkte tilkoblet WS, som har mulighet til å fri rotasjon, ønsker vi ikke i hindre denne på noen måte. Vi ønsker derfor at kommunikasjon som skal skje tilbake til test software foregår trådløst.

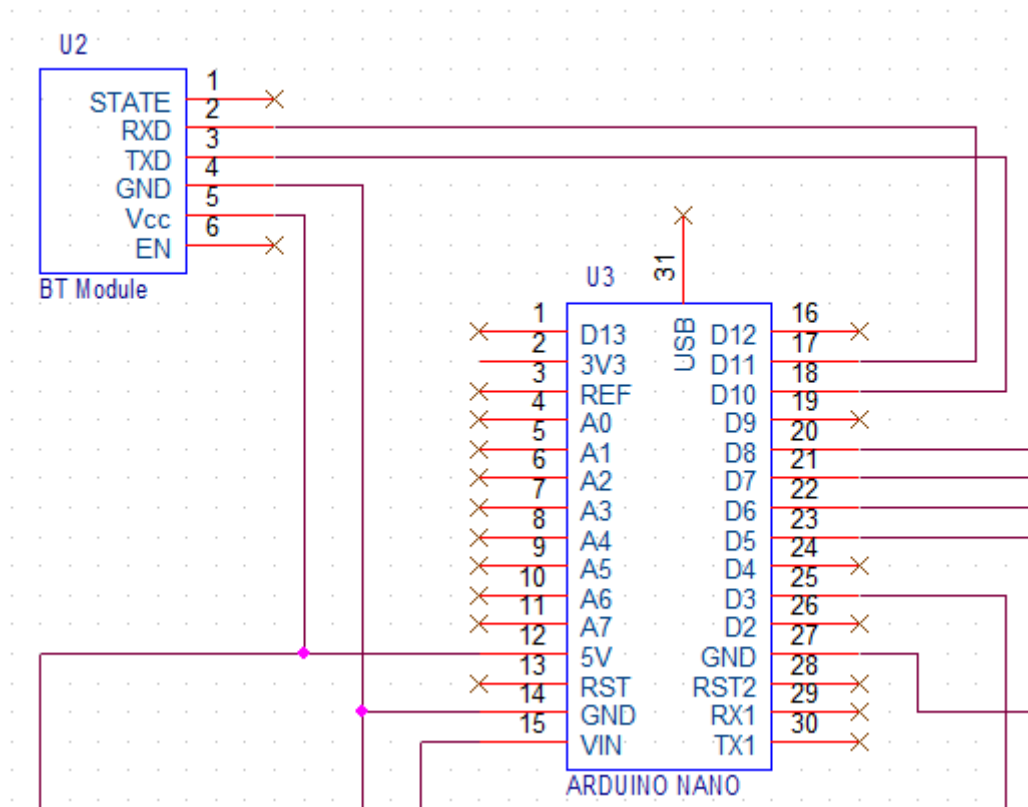
Flere løsninger er vurdert, deriblant Blåtann, WiFi og 433MHz radio. Valget falt på blåtann da kompleksiteten for bruken av dette er svært lav, og det er en velprøvd å testet løsning. Sikkerhet kom opp som et spørsmål, men vi vurderer verdien av kommunikasjonen som skal passere her som så lav at det ikke er et problem. Rekkevidden vil heller aldri bli noe problem, da test maskin aldri vil stå langt unna WS.

Vi vil derfor ta i bruk en Jbtek HC-06 blåtann modul for å håndtere kommunikasjonen mellom mikrokontrolleren og test software.

Tilkoblingene på modulen gjøres direkte mot mikrokontrollerens +5V, GND og RX/TX porter.



Figur 9: HC-06 Blåtann modul



Figur 10: Tilkobling Arduino mot Blåtann modul

2.2.4 Arduino

Vi besluttet tidlig i prosessen at vi ønsket en Arduino mikrokontroller til å ha rollen som «hjernen» i fyringssensoren vår, da vi så behovet for en litt smartere sensor. Siden vi har erfaring med bruk av denne fra tidligere er ikke terskelen så høy for å ta den i bruk også her.

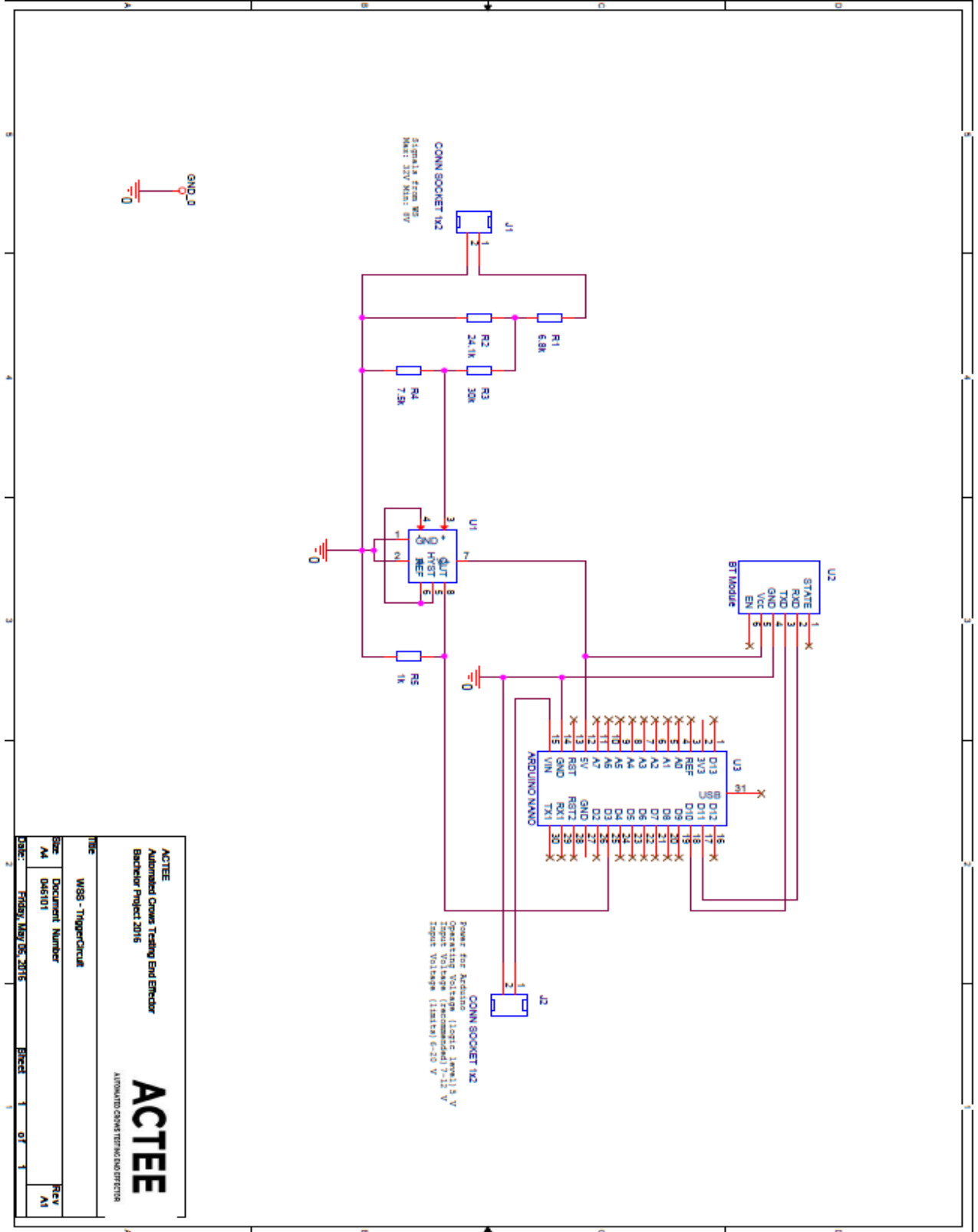
Hvilken Arduino vi vil bruke blir en avveining basert på funksjonalitet og utforming. Da vi ikke ønsker en veldig stor og klumpete sensor, da vi kan risikere at denne blir i veien for WS sine normale funksjoner, ble vurderingen gjort hovedsakelig mellom to av Arduino sine mindre produkter.

Arduino Nano, og Arduino Uno er da to naturlige valg å se på. Uno har fordelen med at den er svært robust, billigere og lettere å jobbe med, da den har connectorer på hver utgang og er ferdig tilpasset til bruk med Arduino sine prototype shields. Nano har fordelen av at den er en god del mindre enn Uno, og at den er ferdig tilpasset til bruk med breadboards, det vil si, den er lett å lage test-kretser med. Ellers er de nokså like når det gjelder antall I/O porter og annen funksjonalitet.

Valget falt på Nano, hovedsakelig på grunn av formfaktoren, og fordi den er godt egnet til å loddas på prototype breadboards for en mer permanent sammenstilling.

2.2.4.1 Arduino tilkoblinger

Arduino sine tilkoblinger i den komplette kretsen vises i figur 11.



Figur 11: Fyringskrets

Tabellen under viser en kort oversikt over de benyttede tilkoblingene og hvilke signaler som forventes å finne der.

Tabell 3: Tilkoblinger Arduino Nano

Pin Navn	Pin Nr	Koblet mot	Signal
5V	12	Koblet sammen med Vcc til BT modul og V+ på komparator	+5V
GND	15	Koblet sammen med GND fra WS, Strømkilde, BT Modul	0V /GND
D11	17	Koblet mot RXD på BT Modul	SerialCom
D10	18	Koblet mot TXD på BT Modul	SerialCom
D8	20	Koblet mot connector mot WS, pin A (Weapon_ID_4)	GND/5V
D7	21	Koblet mot connector mot WS, pin J (Weapon_ID_3)	GND/5V
D6	22	Koblet mot connector mot WS, pin E (Weapon_ID_2)	GND/5V
D5	23	Koblet mot connector mot WS, pin D (Weapon_ID_1)	GND/5V
D5	25	Koblet til pin 8 på komparator, og motstand mot GND	4.8V/0V
GND	27	Koblet mot connector mot WS, pin B og K (Weapon_ID_GND)	0V /GND

Tabell 4: Bill of materials WSS

Item Number	Quantity	Value	Description	Artikkelnr	Part Reference	Forhandler	Stk Pris
		CONN SOCKET	Screw Terminals 5mm	SPF-PRT-			
1	1	1x2	Pitch (2-Pin)	08432	J1	youblob	14 NOK
2	1	8 HEADER		-	J2	-	-
3	2	6.8k	Motstand 6.81 kOhm 0.11W +/- 0.1%	160-59- 627	R1	ELFA	4,64 NOK
4	2	24.1k	Motstand 24.1 kOhm 0.11W +/- 0.1%	160-59- 596	R2	ELFA	3,25 NOK
5	1	30k	Motstand 30 kOhm 0.11W +/- 0.1%	160-59- 604	R3	ELFA	4,64 NOK
6	1	7.5k	Motstand 7.5 kOhm 0.11W +/- 0.1%	160-59- 631	R4	ELFA	4,64 NOK
7	1	1k	Motstand 1 kOhm 0.11W +/- 0.1%	160-59- 570	R5	ELFA	4,64 NOK
8	4	220	Motstand 220 Ohm 0.11W +/- 0.1%	110-52- 165	R6, R7, R8, R9	ELFA	14,90 NOK
9	1	MAXIM931CPA+	Komparator Enkel DIL-8	330	U1	ELFA	39,20 NOK
10	1	BT Module	Bluetooth SMD Module - RN-41	SPF-WRL- 11786	U2	youblob	169 NOK
11	1	ARDUINO NANO	Mikrokontrollerkort/Nano Atmega328	110-96- 733	U3	ELFA	171 NOK

2.2.4.2 Arduino Scetch

Arduino koden vår har behov for å registrere et høyt signal på en av de digitale inngangen, samt kunne sende ut et binært 4-bits signal på 4 utganger, samt kunne kommunisere serielt vi en Blåtann modul. For å muliggjøre dette lager vi en scetch i Arduino IDE som vi kan laste opp til mikrokontrolleren via USB. Komplette scetch ligger tilgjengelig her

http://www.mediafire.com/download/nb8t77dd2pjycbb/WSS_NANO_3_0-2016-05-20.zip

2.2.4.2.1 Bibliotek og variabler

```
14 #include <Arduino.h>           //Adds the arduino.h library to the scetch
15 #include <SoftwareSerial.h>     //Adds the softwareSerial.h library to the scetch to enable communication with the BT modul
16
17
18 const byte rxPin = 10;          //Defines a constant, for SWSerial communication
19 const byte txPin = 11;          //Defines a constant, for SWSerial communication
20 SoftwareSerial mySerial (rxPin, txPin); //Defines the RX/TX pins to be used by mySerial
21
22 const int intPin = 3;           //Defines pin 3 to be called intPin, will be used in the interrupt()
23 const int wldPin[] = {5, 6, 7, 8}; //Defines an array of constant pins, will be used in the setWid()
24
25
26 volatile int counter = 0;       //Defines a counter to keep track of number of triggers detected, volatile to be able to use it in the interrupt()
27 volatile unsigned long last_micros; //Defines a placeholder for time, volatile to be able to use in interrupt()
28
```

Figur 12: Bibliotek og variabler

Første del av koden inneholder inkludering av bibliotek og definering av globale variabler.

Vi legger ved Arduino.h biblioteket da vi skal bruke flere av Arduino sine egne funksjoner som støtter seg på dette biblioteket, deriblant interrupt funksjonen.

2.2.4.2.2 SoftwareSerial

Kommunikasjonen mot blåtann modul ble satt opp ved å bruke SoftwareSerial biblioteket. Ved å benytte dette biblioteket isteden for den innebygde Serial kommunikasjonen, kan vi uten videre koble til USB kabel for å laste opp ny kode til enheten. Om man velger å benytte seg av standard Serial ville man måtte koble fra Blåtann modulen hver gang man kobler til USB kablen da to enheter ikke kan dele en SerialCom kanal.

Funksjonen benyttes ved å definere to digitale pinner som skal fungere som TX og RX kanaler, deretter definere hva man ønsker å navngi serial kanalen til, og supplerer de definerte portene som vist i figur 13.

```
15 #include <SoftwareSerial.h>           //Adds the softwareSerial.h library to the scetch to enable communication with the BT modul
16
17
18 const byte rxPin = 10;          //Defines a constant, for SWSerial communication
19 const byte txPin = 11;          //Defines a constant, for SWSerial communication
20 SoftwareSerial mySerial (rxPin, txPin); //Defines the RX/TX pins to be used by mySerial
```

Figur 13: SoftwareSerial

Når dette er gjort kan man bruke den som en hvilken som helst annen serial kommunikasjons kanal, men man benytter det valgte navnet istedenfor Serial, Serial1 osv.

2.2.4.2.3 void setup()

```

29 void setup()
30 {
31   mySerial.begin(9600);           //Starts Serial communication using Bluetooth (BT module, baud = 9600, MUST match the BT modules setup),
32   pinMode(intPin, INPUT);        //defines intPin as a input pin, placing it ready to receive
33   for (int i = 0; i < 4; i++)     //for loop, defining wldPins to be OUTPUT pins, and set them low, sending an F to the WS.
34   {
35     pinMode(wldPin[i], OUTPUT);   //defines as OUTPUT
36     digitalWrite(wldPin[i], LOW); //Sets pin low
37   }
38   attachInterrupt(digitalPinToInterrupt(intPin), fire, RISING); //Defines interrupt, triggers the fire() function when rising signal detected on intPin
39 }

```

Figur 14: void setup()

I void setup() definerer man vanligvis ting som man ikke planlegger å endre, eller start verdiene man ønsker å sette.

mySerial.begin(9600) setter kommunikasjons hastigheten til mySerial til å være 9600 baud. Dette er gjerne en standard verdi, og siden vi ikke har behov for noe raskere kommunikasjon enn dette beholder vi denne verdien. Om man ønsker å endre denne er det viktig å rekonfigurere Blåtann modulen slik at dens hastighet samsvarer med den satt her.

Deretter defineres intPin til å være en input pin, dvs. det forventes å lese et signal på denne pinnen, i motsetning til wldPin som defineres til en output. wldPin settes samtidig til å ligge lavt for å sende en F til WS som standard ved oppstart.

attachInterrupt () brukes til å legge til interrupt funksjonen vi ønsker. Den kobles mot intPin, vår eneste input pin. Vi definerer videre at vi ønsker at den skal kjøre funksjonen fire() når den trigges, og at den skal trigge på et signals rising edge. Det er flere muligheter å velge mellom når det gjelder å bestemme når interrupt skal trigge, men for at vi skal ha raskest mulig respons på fyringsdeteksjonen ønsker vi at det skjer med en gang signalet stiger.

2.2.4.2.4 void fire()

```

80 void fire()                               //Funksjon fire called by interrupt
81 {
82   if ((long)(micros() - last_micros) >= 1000000) //IF will only run if there is more than 100 millisecc since last time it ran.
83   {
84     //counter = counter + 1;                //Adds 1 to current count.
85     //String report = String(counter);      //converts the int counter to the string report
86     //mySerial.print(report);               //Sends the value of counter to the BT module
87     mySerial.print("F");                   //Sends an F as a string, signals that a high fire signal was detected
88     last_micros = micros();                 //notes the current time, will be used the next time the loop tries to run
89   }
90 }

```

Figur 15: void fire()

Funksjonen fire vil som definert i setup() kjøre når det detekteres et stigende signal på intPin. Det første funksjonen gjør er å sjekke om det er mer enn 100 millisekunder siden sist den ble kjørt. Om det er tilfellet, sender den en F som en string til test software vi blåttann modulen, som er det avtalte tegnet for at en registrert fyring skal legges i loggen.

Det ligger også kode kommentert ut som legger til 1 i counter variabelen, lagrer den nye verdien som en String i report variabelen og sender denne via Blåttann modulen til test Software for logging. Denne kan aktiveres igjen for å få med antall registrerte avfyringer i loggen.

Til slutt noterer den tiden i last_micros variabelen, slik at den kan bruke denne neste gang funksjonen kjører.

Hvis det er mindre enn 100 millisekunder siden sist, vil funksjonen la være å gjøre noe som helst. Tiden er satt til 100 millisekunder siden fyringssignalet fra WS er så full av transienter at om vi hadde satt funksjonen til å sende uavhengig av tid, ville vi registrert rundt 30 avfyringer hver gang.

2.2.4.2.5 void loop()

```

42 void loop()
43 {
44     for (int i = 0; i <= 10; i++)                // for loop, will run 10 times
45     {
46         if (mySerial.available())                //if loop, sends recived char from BT Serial to USB Serial
47         {
48
49
50             while (mySerial.available() > 0)      //While loop, runs as loong as data is detected in mySerial buffer.
51             {
52                 String input = "";                //string to save recived char from mySerial
53                 char nextChar;                    //Char to temp save recived char from mySerial
54                 nextChar = mySerial.read();        //collects the next char from buffer
55
56                 if (nextChar == 'R')              //if recived char is R, counter is reset
57                 {
58                     sendReset();
59                 }
60
61                 if (nextChar == 'W')              //if recived char is W
62                 {
63                     setWid();
64                 }
65
66                 else                              //if recived char is !R
67                 {
68                     input += nextChar;            //adds recived char to the end of string
69                 }
70             }
71         }
72
73         delay(200);                               //Pause, slik at den ikke sender informasjon kontinuerlig
74     }

```

Figur 16: void loop()

Hoveddelen i koden består av en for løkke som vil repetere ti ganger, før det sendes en '#' til test software. '#' ble avtalt med Software at skulle fungere som et keep alive signal, det vil si at om software ikke mottar denne på en stund skjønner den at noe er galt med tilkoblingen mot WSS. For løkken avsluttes med en delay () på 200 millisekunder, dette gjør at keep alive signalet sendes ca. hvert andre sekund.

I for løkken kjøres det først en sjekk for å se om det ligger noe i Serial kommunikasjon bufferen til Blåtann enheten. Om det gjør det leses det første tegnet inn i nextChar variabelen. Deretter kjøres det to tester, om det leste tegnet er enten 'R' eller 'W', så legges tegnet til i input variabelen. Siden koden i utgangspunktet ble skrevet for å kunne håndtere kommunikasjon både via Blåtann modul og via USB-kabel, ble reaksjonene på R og W lagt ut til egne funksjoner.

Ved mottatt 'R', som er avtalt som resett signal for counter variabelen som lagrer antall detekterte avfyringer kalles sendReset () funksjonen.

Ved mottatt 'W', kalles setWid () funksjonen som endrer våpen Id WSS sender til WS.

2.2.4.2.6 void setWid()

setWid () som kalles når WSS har mottatt 'W' fra test Software vil i første omgang opprette en variabel med navn temp, og lese inn neste tegn fra bufferen til denne. Dette tegnet er forventet å være fra 0 til F, de 16 mulige våpen id man kan velge mellom.

Det opprettes deretter en ny variabel av typen byte for å lagre verdien fra temp i binær form.

wID variabelen brukes videre i en for løkke som repeteres 4 ganger, en gang for hver bit i variabelen.

For løkken benytter seg der av funksjonen bitRead() som leser verdien på plass i den angitte variabelen. Om denne verdien er lik 1, setter pin lav, ellers settes den høy.

Siden vi valgte å lage våre wldPin som et array får vi her til å automatisere at koden leser verdien av LSB og sender den inverterte av denne verdien ut på vår LSB pin som er wldPin[0], eller D5. Den repeterer denne prosessen til den har gått igjennom alle 4 pinner, og avslutter med MSB.

```
99 void setWid()
100 {
101     char temp;
102     if (mySerial.available())
103     {
104         temp = mySerial.read();
105     }
106
107     byte wID = 0;
108     if (temp == '1') wID = 1;
109     if (temp == '2') wID = 2;
110     if (temp == '3') wID = 3;
111     if (temp == '4') wID = 4;
112     if (temp == '5') wID = 5;
113     if (temp == '6') wID = 6;
114     if (temp == '7') wID = 7;
115     if (temp == '8') wID = 8;
116     if (temp == '9') wID = 9;
117     if (temp == 'A') wID = 10;
118     if (temp == 'B') wID = 11;
119     if (temp == 'C') wID = 12;
120     if (temp == 'D') wID = 13;
121     if (temp == 'E') wID = 14;
122     if (temp == 'F') wID = 15;
123
124     for (int i = 0; i < 4; i++)
125     {
126         if (bitRead(wID, i) == 1)
127         {
128             digitalWrite(wIdPin[i], LOW);
129         }
130         else
131         {
132             digitalWrite(wIdPin[i], HIGH);
133         }
134     }
135 }
```

Figur 17: void setWid()

3 Konklusjon

WSS utfyller sin funksjon på en tilfredsstillende måte. Den vil melde fra umiddelbart om den detekterer et høyt signal fra WS og vil reagere på signaler innenfor hele den mulige rangen WS kan operere på, og litt til.

Enheten kan supplere WS med alle 16 mulige våpen Id'er. Det er ikke alle som er i bruk i dag, men siden WSS kan supplere alle vil det ikke trenge noen tilpasning ved eventuelle utvidelser av bruken fra i dag. Ved en eventuell endring til 5 eller flere bits, må det gjøres en endring i både kode og krets, men dette er relativt små endringer som må gjøres og vil være lett å gjennomføre for noen som har lest dokumentasjonen.

Kommunikasjonen mellom WSS og test software fungerer trådløst slik som planlagt uten at vi har oppdaget noen problemer med dette. Da dette ivaretar bevegelses friheten til WS må vi si oss svært fornøyde med dette.

Ved mer informasjon om trigger signalet kunne vi kanskje ha muliggjort en mer nøyaktig telling av antall fyrings signaler, også i burst og full-auto modus. Men dette lot seg dessverre ikke gjøre med informasjonen vi hadde tilgjengelig.

4 Referanser

[1] Universal Robots AS. (2015). *UR5 - roboter*.

Tilgjengelig fra:

<http://www.universal-robots.com/no/produkter/ur5-robot/>

Sist besøkt: 10.05.2016

[2] [A.K.](#) Svendsen ; E.L. [Roa](#) ; H.B. [Sørum](#) ; S. [Rudin](#). "Automated CROWS Testing" Bachelor Thesis, Fakultet for teknologi og maritime fag, Høgskolen i Buskerud og Vestfold, Kongsberg, 2014.

Tilgjengelig fra:

<http://hdl.handle.net/11250/216910>

[3] James Bruce, (2011, September 25), *What Is Arduino & What Can You Do With It?*

[Nettsted].

Tilgjengelig fra:

<https://www.arduino.cc/en/Guide/Introduction>

Sist besøkt: 10.05.2016

[4] Arduino. (2008). Arduino Nano (V2.3) User Manual.

[Nettsted].

Tilgjengelig fra:

<https://www.arduino.cc/en/uploads/Main/ArduinoNanoManual23.pdf>

Sist besøkt: 10.05.2016

[5] Arduino. (Ukjent). Arduino Uno and Genuino Uno.

[Nettsted].

Tilgjengelig fra:

<https://www.arduino.cc/en/main/arduinoBoardUno>

Sist besøkt: 10.05.2016

[6] Electronics-Tutorials. (Ukjent). Op-amp Comparator.

[Nettsted].

Tilgjengelig fra:

<http://www.electronics-tutorials.ws/opamp/op-amp-comparator.html>

Sist besøkt: 10.05.2016

[7] MAXIM. (Ukjent). Datasheet MAX931CPA+.

[Nettsted].

Tilgjengelig fra:

<https://www.elfadistelec.no/Web/Downloads/21/58/epT02158.pdf?mime=application%2Fpdf>

Sist besøkt: 10.05.2016

END EFFECTOR

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAKSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	End Effector
VERSJON / DATO	2.0 / 20. Mai 2016
FORFATTER	Monica Nadia Haugen
SIDEANTALL	6



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



Abstrakt

Formålet med denne oppgaven er å utvikle et gripeverktøy som fungerer som et fysisk grensesnitt mot M153 våpenstasjon. Det er utviklet et gripeverktøy med tre fingre. For å oppnå et ønsket resultat, må størrelsesforhold mellom håndflate og fingre, samt avstand og passering av fingre i håndflaten, være i en bestemt harmoni.

Forord

Kongsberg Protech Systems og Høgskolen i Sørøst-Norge, veileder oss gjennom dette prosjektet som er en videreføring av en tidligere bachelor oppgave ved navn ACT¹ (Automated Crows Testing). Hovedmålet med oppgaven er å effektivisere testkjøring av våpenstasjon M153 med Crows software. Vårt utgangspunkt var en ferdig installert robot arm fra Universal Robot av typen UR5². Oppgaven er spennende og svært aktuell. Automatisering ved hjelp av en robot med et egnet verktøy, tar stadig over monotone arbeidsoppgaver. Vårt produkt skal gjøre nettopp dette. Vårt mål er å presentere et gripeverktøy med evne til å foreta grunnleggende bevegelser, tilrettelegge og vurdere hvordan videre utvikling skal dekke flere funksjoner.

¹ Automated Crows Testing;

² UR5 er en seks-akset sylindrisk robot arm med 5kg løftekapasitet. <http://www.universal-robots.com/no/produkter/ur5-robot/>

Dokumentoversikt

DEL 1: Teknisk dokument for elektronikk

DEL 2: Designdokument for mekanikk

DEL 3: Teknologidokument for mekanikk

Tabell 1 Dokumenthistorie

v 0.1 19.02.16	Monica: oppbygging med innhold og struktur. Forslag og stikkord for innhold.
v 0.2 22.02.16	Monica: Forarbeid tekst
v.0.3 25.02.16	Monica: Tillegg tekst
v 0.4 28.02.16	Monica: Tillegg tekst
v 0.5 09.03.16	Monica: Restrukturering av oppsett
v 0.6 11.03.16	Monica: Sammenslåing el/mek, tillegg tekst
v 0.7 12.03.16	Monica: Tillegg tekst, Carl Martin: Elektronikkdokumentet, Tekniskdokument for elektrisk styring av EE (v0.2) implementeres.
v 0.8 13.03.16	Monica: Redigering, tillegg Carl Martin: tilleggstekst servoer.
v. 2.0 20.05.16*	Monica: oppretter ny versjon, revidert utgave.

* Tidligere versjoner splittet. Ny versjon, likt formål.

1 Innledning

1.1 Oppgaven

Vår oppgave er å utvikle et gripeverktøy som skal koples på en seks-akset robot arm. Robot armen er laget av Universal Robots, og heter UR5. Gripeverktøyet skal kunne betjene grensesnitt for testing av ulike funksjoner på en våpenstasjon. Dette omfatter operasjoner som i utgangspunktet er tilrettelagt for en menneskehånd. Sammen med elektronisk styring og programmering, skal gripeverktøyet automatisere monotone arbeidsoppgaver for en mer effektiv ressursbruk. Hovedprosjektet er gjennomført våren 2016 ved Høgskolen i Buskerud og Vestfold. Oppdragsgiver er Kongsberg Protech Systems.

1.2 Rapportens innhold

Denne tekniske rapporten beskriver hvordan vi har vurdert valg av design, styringsmekanismer, kraftoverføring og programmering for EE. Vi begrunner hvorfor vi har valgt løsningene ved å belyse alternativer for konsept, og forklare hvordan vi har tenkt.

Vi forklarer hvordan vi forsøker å forutse hindringer og problemstillinger, og belyser de ulike grensesnitt som et naturlig fokus opp mot en videre oppfølging av risikovurdering.

1.3 Oppbygging

Rapporten er ordnet i tre deler.

Del 1 omhandler vårt utgangspunkt, robot armen fra Universal Robots, og elektronikk for EE.

Del 2 omhandler designet, og beskriver fasene fra idé til planlegging og konseptvalg.

Del 3 omhandler teknisk informasjon knyttet til den mekaniske delen av EE.

Teknisk Dokument Elektronikk

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAKSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Teknisk Dokument Elektronikk
VERSJON / DATO	2.0 / 20.05.2016
FORFATTER	Carl Martin Mausest og Bård Simen Hamborg Enget
SIDEANTALL	70



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



i. Abstrakt

Dette er ett av to dokumenter tilhørende teknologi dokument EE, dette dokumentet vil gi den som leser en forståelse for hva slags elektronikk som vil bli brukt for å kunne gi konseptet for End Effector bevegelse. Det vil også vises grunnlagene for hvorfor vi har gått for det designet og type elektronikk som er blitt valgt med tanke på implementering av styreenheter.

ii. Innholdsfortegnelse

i. Abstrakt.....	1
ii. Innholdsfortegnelse	2
iii. Oversikt tabeller.....	4
iv. Oversikt figurer	5
v. Dokumenthistorie	6
vi. Forkortelser	7
1 Innledning	8
2 Utvalgelse av type motor.....	9
2.1 Styrings konsepter.	9
2.2 Krav til motorer.	9
2.2.1 Vektgrunnlag for pugh matrisene.	9
2.2.2 Masse og størrelse.	9
2.2.3 Kraft (kg/cm)	9
2.2.4 Effektforkbruk.	10
3 Pugh Matrise for motorer	11
3.1 Vektgrunnlag	11
1.3 Oppsummering av pugh matrise for motorer.....	21
4. Utvalgelse av Servomotorer	22
4.1 Vektings grunnlag av krav til servoer.....	22
4.1.1 Pris	22
4.1.2 Spenning.	22
4.1.3 Elektrisk strøm.	22
4.1.4 Kraft moment.	22
4.1.5 Masse.....	23
4.1.6 Størrelse.	23
4.1.7 Hastighet	23
5. Pugh Matrise for servomotorer.....	24
5.1 Vekting av pugh matrisene.	24

5.2 Oppsummering av pugh matrise for servoer.	40
7 Servomotor	42
8 Kontrollere	44
8.1 Arduino tilkobling	46
8.2 Arduino «Scetch»	49
8.2.1 Bibliotek og variabler	49
8.2.2 «void setup()»	51
8.2.3 «Void loop()».....	52
8.2.4 «Void inputAction()»	53
8.2.5 «void printStatusAndReset()».....	54
8.2.6 «void stateTest()»	55
8.2.7 «int changePos()»	55
8.3 Signal ledere	57
9. Strømkilde.....	58
9.1 CPX200	58
9.2 Bruk av Strømforsyning	59
9.3 Elektrisk ledere	60
9.4 Koblingsskjema for End Effector	61
10. Universal robot 5. UR5.	62
10.1 UR5 «software program.».....	64
10.2 UR5-Arduino «interface»	64
10.2.1 UR5 «tool I/O»	64
10.3 UR5-Arduino krets	65
10.4 Kommunikasjon	67
11 Konklusjon	68
11.1 Konklusjon servomotorer	68
11.2 Konklusjon kontrollere	68
11.3 Konklusjon strømkilde.....	68
Kilder	69

iii. Oversikt tabeller

Tabell 1: Dokumenthistorie.....	6
Tabell 2: Forkortelser	7
Tabell 3: Vektgrunnlag til pugh matrise for motorer	11
Tabell 4: Audi 100 visker motor.....	12
Tabell 5: Sanyo Denki 103H5208.....	13
Tabell 6: 238 Series Wiper Motor.....	14
Tabell 7: Mini Electric Metal Gear Motor.....	15
Tabell 8 Kinmore KM-32A500	16
Tabell 9: 50G EXI Digital Servo.....	17
Tabell 10: EMAX Mini Servo	18
Tabell 11: Micro stepper motor.....	19
Tabell 12: Micro geared reducer stepper motor.....	20
Tabell 13: TT MOTOR GM12-15BY micro.....	21
Tabell 14: Vektgrunnlag for Servomotorer.....	24
Tabell 15: Hitec HS-5665MH.....	25
Tabell 16: Hitec HS-645MG	26
Tabell 17: Hitec HS-635HB	27
Tabell 18: Hitec HS-5646WP	28
Tabell 19: Hitec HS-5495BH	29
Tabell 20: Hitec HS-485HB	30
Tabell 21: Futaba S3010.....	31
Tabell 22: Hitec HS-75BB	32
Tabell 23: JR NX588.....	33
Tabell 24: Power HD-1214TH HV	34
Tabell 25: Power HD-1235MG	35
Tabell 26: Power HD-1501MG	36
Tabell 27: Savöx Servo SC-0251MG	37
Tabell 28: Savöx Servo SC-1256TG	38
Tabell 29: Hitec HS-5485HB	39
Tabell 30. Power HS-1207 TG	41
Tabell 31. Elektrisk spesifikasjoner HS-1207 TG.....	43
Tabell 32. Mekaniske spesifikasjoner HS-1207 TG	43
Tabell 33.Kontroll spesifikasjoner HS-1207 TG.....	43
Tabell 34 Tekniske spesifikasjoner for Arduino ATmega 2560 [1]	45
Tabell 35. Tilkoblings oversikt for Arduino Atmega 2560	46
Tabell 36.Teknisk spesifikasjoner	57
Tabell 37. Teknisk spesifikasjoner CPX 200 power supply [3]	58
Tabell 38. Teknisk Spesifikasjoner AVG 16 kabel.	60
Tabell 39. Tekniske spesifikasjoner til UR5 [4]	63

Tabell 40. «Tool I/O» signal oversikt	64
---	----

iv. Oversikt figurer

Figur 1. HS-1207 TG servomotor.	42
Figur 2. 2D tegning av HD-1207 TG[x]	42
Figur 3: Arduino ATmega 2560	44
Figur 4. Arduino Mega Pin oversikt [5]	47
Figur 5. ATmega 2560 IC PIN diagram. [2]	48
Figur 6. Utdrag Arduino kode, variabler og bibliotek	49
Figur 7 : Utdrag Arduino kode, «void setup()»	51
Figur 8 : Utdrag Arduino kode, «void loop()»	52
Figur 9 : Utdrag Arduino kode, Skifte av servo posisjon	52
Figur 10 : Utdrag Arduino kode, «default action»	53
Figur 11 : Utdrag Arduino kode, «void inputAction()»	53
Figur 12 : Utdrag Arduino kode, void printStatusAndReset()	54
Figur 13 : Utdrag Arduino kode, «void stateTest()»	55
Figur 14. Utdrag Arduino kode, «int changePos()»	55
Figur 15. Signal ledere	57
Figur 16. CPX 200 power supply	58
Figur 17. Effekt fleksibilitet [3]	59
Figur 18. AVG 16 kabler for strømforsyning	60
Figur 19. Koblingsskjema for EE.	61
Figur 20 UR5. Unversal robot 5.	62
Figur 21. Illustrasjon av plassering av «tool I/O connector»	64
Figur 22 : Eksempel på Digital «Output» bruk	65
Figur 23 : Krets UR5-Arduino «interface»	66

v. Dokumenthistorie

Tabell 1: Dokumenthistorie

Versjon 0.1	15.03.2016	Mal opprettet av Carl Martin
Versjon 0.2	15.03.2016	Lagt inn teknisk dokument fra End Effector V0.9 for Elektronikk. Tillegg i tekst og pugh matrix. Bård Simen: Korrektur og rettskriving.
Versjon 0.3	16.03.2016	Carl Martin: Tabeller, innholdsfortegnelse og figurer.
Versjon 1.0	16.03.2016	Espen: siste sjekk og ferdigstilling Thomas Wegener: Opprettet versjon 1.0
Versjon 1.1	19.04.2016	Carl Martin: Opprettet versjon 1.1: Retting av innhold. Tilleggstekst.
Versjon 1.2	10.05.2016	Carl Martin: Opprettet versjon 1.2. Tilleggstekst.
Versjon 1.3	11.05.2016	Carl Martin: opprettet versjon 1.3. Tilleggstekst.
Versjon 1.4	18.05.2016	Carl Martin: opprettet versjon 1.4. Tilleggstekst. Bård Simen: tilleggstekst Arduino kode.
Versjon 1.5	19.05.2016	Carl M: opprettet versjon 1.5. korrektur. Bård Simen: Tilleggstekst UR5 interface.
Versjon 2.0	20.05.2016	Bård Simen: Oppdatert Arduino kode, gjort noen rettinger.

vi. Forkortelser

Tabell 2: Forkortelser

Forkortelser	Forklaring
KPS	Kongsberg Protech Systems
HSN	Høgskolen i Sørøst-Norge
ACTEE	Automated CROWS Testing End Effector
EE	End Effector, gripeverktøy
CG	Control Grip, styringsjoystick for våpenstasjon
DCP	Display Control Panel
UR5	Universal Robots 5 robotarm
US	User Stories
AC	Acceptance criteria

1 Innledning

For å kunne gi bevegelse til EE har vi ta noen valg når det kommer til hvordan vi ønsker å løse dette. Siden Universal Robot 5 (UR5) allerede tilrettelegger for styring med elektriske signaler, blir det naturlig at vi ser på forskjellige muligheter for bruk av elektriske motorer.

Vi har brukt mye tid å undersøkt forskjellige type motorer for å få et overblikk av hva som finnes på markedet, og hvordan de er bygget opp. Det finnes utrolig mange forskjellige typer elektriske motorer, de vi ser blir brukt mest er servomotorer, sterkt giret DC motorer og stepper motorer. Med tanke på at vi ikke trenger motorer som roterer flere runder for å bevege "fingrene" så vil en servomotor som holder seg innen 180 graders rotasjon egne seg godt til vårt konsept, de er ikke så store og er lette å programmere og gir god kraftmoment. Siden de også kommer i en egen pakke, er de også svært lette å bytte ut ved behov, som tilrettelegging for drift og styring av påmonterte verktøy, har UR5 roboten et eget uttak for signaler og strøm.

Siden uttaket har et begrenset mengde leveranse av strøm 12V/24V og 600mA vil en erstatnings mulighet være å ha en ekstern strømkilde. Den begrensede mengden strøm vi kan hente fra UR5 gir oss noen utfordringer med tanke på servomotor valg. Vi har fått lånt en lab strømforsyning av Kongsberg Protech systems (KPS) som vi planlegger å kartlegg strømbehovet vårt med. Resultatene av disse testene vil gi oss et bedre bilde av mulighetene våre. Styringssignalene vi kan sende via UR5 er også noe begrenset, vi ser derfor at vi vil ha behov for en egen kontroller for vår løsning, vi har fra tidligere erfaring med Arduino og ønsker å benytte denne typen kontroller til vårt prosjekt, disse er ferdig moduler og enkle å jobbe med.

2 Utvelgelse av type motor.

2.1 Styrings konsepter.

Etter en del undersøkelse på nettet for motorer som brukes i liknende applikasjoner som vi skal utvikle i vårt prosjekt, samt forslag til andre muligheter, har vi plukket ut diverse typer motorer. For å få et mest mulig helhetlig og gjennomførbart design, gikk vi gjennom en del forskjellige konsepter for mekanisk overføringer for kraft ut i ledd til EE. Dette vil bli vist i **[Design dokument for grep.]**

2.2 Krav til motorer.

Ved User story (US-050101, US-051001, US-051101, US-044101) fra **[Prosjektplan]**. Stilles det forskjellige krav.

Dette er det vi har satt i fokus for at EE skal kunne betjene Display Control Panel (DCP) og Control Grip (CG):

2.2.1 Vektgrunnlag for pugh matrisene.

Som vist i **[Tabell 4 til 13.]** Har vi hatt en del undersøkelse for type motor vi vil bruke. Ved vekting av poeng for pugh matrisen har vi fulgt US kravene slik at vi setter en poengsum fra 1-5 poeng ettersom hva som er viktig, grunnlag for positiv verdi eller negativ verdi hvis de oppfyller kriteriene i **[Tabell 3.]**

2.2.2 Masse og størrelse.

Ved pugh matrisene **[Tabell 4 til 13.]** er det først og fremst viktig at vi ikke velger en motor som er tung og stor, slik at konseptet blir klønete og for tungt for at UR5 kan ha den festet på seg og gjøre oppgaver, dermed har masse og størrelse fått en poengsum på 5, og vektgrunnlaget for poeng blir satt til positiv ved mindre enn 50g i masse, og negativ ved en masse på mer enn 50g som vist i **[Tabell 3.]**

2.2.3 Kraft (kg/cm)

Vi har tenkt at kraftmomentet er noe som kan videre undersøke etter valg av motorer og har derfor fått en poengsum på 3, men satt vektgrunnlaget til positiv verdi ved kraft på mer enn 4kg/cm og negativ poeng med kraft mindre enn 4kg/cm.

2.2.4 Effektforbruk.

Forbruk av spenning/elektrisk strøm er det ikke satt så høyt med tanke på vektings-poeng i starten av prosjektet. Ideelt ønsker vi at UR5 kan betjene motorene med forsyningen den leverer, men dette kan løses med en ekstern strømkilde, da vi er klar over UR5 sin begrensning på 12V/24V og 600mA. Vekting for dette punktet er blitt satt til 2 poeng med vektgrunnlag på positiv ved mindre enn 12V og negativ ved mer enn 12V. Og vi har unnlatt å legge vektgrunnlag på oppgitt elektrisk strøm som vist i **[tabell 3]**.

3 Pugh Matrise for motorer

Ved bruk av pugh matrisen for utvalgelse av motorer har vi gitt en poengsum for hva vi ønsker at motorene har som egenskap, her er det gitt 0-5 poeng. Vi setter et vektgrunnlag slik at poengene enten trekker negativt eller positivt for å kunne gi en helhetlig total sum.

3.1 Vektgrunnlag

[Tabell 3] viser vekting av egenskaper til motorene som er viktig for konseptet i første utvalgelse.

Tabell 3: Vektgrunnlag til pugh matrise for motorer

Vektgrunnlag: Positiv (1). Negativ (-1). Eller nøytral verd(0)	Spennings:	Kraft:	Masse:	Størrelse:
1	<=12v	>4kg/cm	<50g	<40x20x40mm
-1	>12v	<4kg/cm	>50g	>40x20x40mm
0	-	-	-	-

Tabell 4: Audi 100 visker motor

Navn	Audi 100 visker motor Art.nr: 63191 Originalnr. 443 955 113 C	Kriterier:	Poeng:	Vekting:
Pris	NOK 472,-	Spenning	2	1
Spenning:	N/A (12 v bil batteri)	Kraft	3	1
Elektrisk strøm:	N/A	Masse	5	-1
Masse:	N/A disse typene er generelt tung.	Størrelse	5	-1
Størrelse:	17x12cm, Sylinder formet	Sum av positive		5
Kraft:	3 kg-cm	Sum av negative		10
		Sum av nøytrale		0
		Total:		-5
Referanse:	https://www.skruvat.no/Viskermotor-P63191.aspx			
Kommentar:	Disse Motorene blir brukt til vindusviskere på biler, De er store så de vil nok ikke være et godt utfall og ende med.			

Tabell 5: Sanyo Denki 103H5208

Navn	Sanyo Denki 103H5208-10U41	Kriterier:	Poeng:	Vekting:
Pris	NOK 137,-	Spenning	2	-1
Spenning:	24V DC	Kraft	3	1
Elektrisk strøm:	1.2 A /phase	Masse	5	-1
Masse:	290g	Størrelse	5	1
Størrelse:	42 MM firkantet	Sum av positive		8
Kraft:	3 kg-cm	Sum av negative		7
		Sum av nøytrale		0
		Total:		1
Referanse:	http://www.ebay.com/itm/Sanyo-2-Ph-Step-Stepper-Motors-CNC-Router-103H5208-10U41-Nema-17-Frame-Hobby-NEW-/291338494060?hash=item43d520a06c			
Kommentar:	Steppermotorer er en god motor som kan holde god kraftmoment. Men denne er typen som er valgt her er ganske tung.			

Tabell 6: 238 Series Wiper Motor

Navn	238 Series Wiper Motor. Product ID #238-1002	Kriterier:	Poeng:	Vekting:
Pris	NOK 900,-	Spenning	2	1
Spenning:	12V	Kraft	3	1
Elektrisk strøm:	N/A	Masse	5	-1
Masse:	N/A disse typene er generelt tung.	Størrelse	5	-1
Størrelse:	178x63 mm +++	Sum av positive		5
Kraft:	3.9 kg-cm	Sum av negative		10
		Sum av nøytrale		0
		Total:		-5
Referanse:	http://store.amequipment.com/238-series-wiper-motor-12v-left-hand-dynamic-park-p-39.html			
Kommentar:	Her gjelder det samme som Audi 100 visker motoren.			

Tabell 7: Mini Electric Metal Gear Motor

Navn	DC 12V 300RPM Mini Electric Metal Gear Motor Gearwheel N20 3mm Shaft Box Motor	Kriterier:	Poeng:	Vekting:
Pris	NOK 30	Spenning	2	1
Spenning:	12V	Kraft	3	-1
Elektrisk strøm:	N/A	Masse	5	1
Masse:	Lett(20-40g)	Størrelse	5	1
Størrelse:	Diameter: 12 mm length: 26 mm	Sum av positive		12
Kraft:	4 kg-cm	Sum av negative		3
		Sum av nøytrale		0
		Total:		9
Referanse:	http://www.ebay.com/itm/DC-12V-300RPM-Mini-Electric-Metal-Gear-Motor-Gearwheel-N20-3mm-Shaft-Box-Motor/321773207015?_trksid=p2047675.c100005.m1851&_trkparms=aid%3D222007%26algo%3DSIC.MBE%26ao%3D1%26asc%3D20131003132420%26meid%3D0ee2a6e54725409dad9cf88783b76db7%26pid%3D100005%26rk%3D1%26rkt%3D6%26sd%3D181904655801			
Kommentar:	Denne typen DC-motor med sterk giring blir ofte brukt i små modeller av RC kjøretøy, de har en god kraft til å være så små.			

Tabell 8 Kinmore KM-32A500

Navn	Kinmore KM-32A500-466- 0610	Kriterier:	Poeng:	Vekting:
Pris:	NOK 10,-/27,-	Spenning	2	1
Spenning:	6V	Kraft	3	-1
Elektrisk strøm:	On load 0.18A No load 0.04A	Masse	5	0
Masse:	N/A	Størrelse	5	1
Størrelse:	32mm diameter x 19.3mm	Sum av positive		7
Kraft:	4 kg-cm	Sum av negative		3
		Sum av nøytrale		0
		Total:		4
Referanse:	http://www.alibaba.com/product-detail/Low-rpm-dc-gear-motor-6v_60031939098.html?spm=a2700.7724838.30.19.U2Z198			
Kommentar:	Denne typen er ganske liten og har en god kraft. Vi vet ikke så mye om vekten til denne motoren.			

Tabell 9: 50G EXI Digital Servo

Navn	50G EXI Digital Servo D124F w/ Metal Gear	Kriterier:	Poeng:	Vekting:
Pris:	NOK 80,-	Spenning	2	1
Spenning:	6V	Kraft	3	1
Elektrisk strøm:	N/A	Masse	5	1
Masse:	50 g	Størrelse	5	1
Størrelse:	39.3x19.4x38.7	Sum av positive		15
Kraft:	6.5 kg-cm	Sum av negative		0
		Sum av nøytrale		0
		Total:		15
Referanse:	http://www.hobbypartz.com/50gexidised11.html			
Kommentar:	Servomotorer er gode potensielle motorer for applikasjoner som er innenfor 180 graders rotasjon, noe som vil gi godt grunnlag med tanke på at vi skal bevege en finger trenger vi ikke mange rotasjoner, denne typen har også god kraft og er heller ikke så store.			

Tabell 10: EMAX Mini Servo

Navn	EMAX Mini Servo ES08MA II 12g	Kriterier:	Poeng:	Vekting:
Pris:	NOK 40,-	Spenning	2	1
Spenning:	4.8~6V	Kraft	3	-1
Elektrisk strøm:	N/A	Masse	5	0
Masse:	12g	Størrelse	5	1
Størrelse:	23x11.5x24mm	Sum av positive		7
Kraft:	(4.8V) 1.6kg-cm (6V) 2.0Kg-cm	Sum av negative		3
		Sum av nøytrale		0
		Total:		4
Referanse:	http://www.hobbypartz.com/66p-205-es08ma-ii.html			
Kommentar:	Som servomotoren 50G EXI. Denne typen er enda mindre og lettere, men yter lite kraft noe som også er avgjørende for at det skal kunne håndteres ved bruk av trykking på knapper.			

Tabell 11: Micro stepper motor

Navn	Micro stepper motor 4 phase 35mm NEMA14 stepper motor with high torque	Kriterier:	Poeng:	Vekting:
Pris:	NOK 80,- /120,-	Spenning	2	1
Spenning:	12v	Kraft	3	-1
Elektrisk strøm:	0.3A	Masse	5	0
Masse:	N/A	Størrelse	5	1
Størrelse:	35.5x35.5x31mm	Sum av positive		7
Kraft:	1.4Kg-cm	Sum av negative		3
		Sum av nøytrale		0
		Total:		4
Referanse:	http://www.alibaba.com/product-detail/micro-stepper-motor-4-phase-35mm_60289177061.html?spm=a2700.7724838.30.25.u0PORK&s=p			
Kommentar:	Som Sanyo Denki modellen, denne modellen er litt mindre, og yter en del mindre kraft.			

Tabell 12: Micro geared reducer stepper motor

	20mm micro geared reducer stepper motor Brand Name: TT MOTOR Model Number:			
Navn	GM20-20BY	Kriterier:	Poeng:	Vekting:
Pris:	N/A	Spenning	2	1
Spenning:	12V	Kraft	3	1
Elektrisk strøm:	0.1A	Masse	5	0
Masse:	N/A	Størrelse	5	1
Størrelse:	Diameter: 20mm.length: 36mm	Sum av positive		10
Kraft:	Stall Torq 10kg-cm	Sum av negative		0
		Sum av nøytrale		0
		Total:		10
Referanse:	http://www.alibaba.com/product-detail/20mm-micro-geared-reducer-stepper-motor_60257703026.html?spm=a2700.7724838.30.33.u0PORK&s=p			
Kommentar:	Denne motoren er ganske liten, og sterk, drar hele 10kg/cm. Dette kan være en potensiell motor for styring.			

Tabell 13: TT MOTOR GM12-15BY micro

Navn	Brand Name: TT MOTOR GM12-15BY micro 15mm stepper motor with gear box	Kriterier:	Poeng:	Vekting:
Pris:	N/A	Spenning	2	1
Spenning:	5V	Kraft	3	-1
Elektrisk strøm:	0.5A	Masse	5	0
Masse:	N/A	Størrelse	5	1
Størrelse:	15.0*21.5mm	Sum av positive		7
Kraft:	(0.3-1) kg-cm (Variert fra modell)	Sum av negative		3
		Sum av nøytrale		0
		Total:		4
Referanse:	http://www.alibaba.com/product-detail/GM12-15BY-micro-15mm-stepper-motor_359761240.html?spm=a2700.7724838.30.41.u0PORK&s=p			
Kommentar:	Denne motoren er ganske liten, men yter for lite kraft.			

1.3 Oppsummering av pugh matrise for motorer.

Vi ser at gjennom en runde med pugh matrisen over kriteriene vi har satt for motorene, fikk typen servomotorer høyest poeng fra [tabell 9] 50G EXI Digital servo. Vi vil derfor gå videre med design og undersøke typer av servomotorer som kan passe konseptet vårt best mulig videre.

4. Utvelgelse av Servomotorer

Fra utvelgelse med pugh matrisen av type motorer gikk vi for typen servomotorer. Vi vil derfor bruke disse som til styring av EE. For at vi skal gjøre et best mulig valg, må vi sette noen videre krav basert på kravene fra Kapittel [2.2 Krav til motorer] og utføre en pugh matrise for flere typer servomotorer, som vist i [Tabell 15. Til 29.]

4.1 Vektings grunnlag av krav til servoer.

For utvelgelse for servomotorer har vi satt en ny pugh matrise for utvalgte servomotorer. Pugh matrisen får en vektings-score fra 1-5 poeng i [Tabell 14] der har vi satt følgende krav.

4.1.1 Pris

For å følge vårt budsjett setter vi en øvre og nedre grense på pris. Positivt grunnlag mindre enn 350kr, og negativt grunnlag mer enn 350kr. Vektingen er satt til en poengsum på 4 i pugh matrisen da dette er viktig for oss å kunne overholde en lav kostnad for budsjettet.

4.1.2 Spenning.

Spenning blir satt til en poengsum på 2, siden vi kan løse dette med en eventuell ekstern strømkilde. Positiv grunnlag er mindre enn 12v, og negativ mer enn 12v.

4.1.3 Elektrisk strøm.

Effektforbruk er blitt satt til en poengsum på 2, siden vi kan løse dette med en ekstern strømkilde om vi trenger mer enn UR5 kan gi oss. Positiv grunnlag er mindre enn 200mA, og negativ mer enn 200mA.

4.1.4 Kraft moment.

Vi vil ha en kraft i motorene som kan tilfredsstille US for bruk av EE. Vi ser for oss at mer enn 10kg-cm vil gi nok kraft ut i EE og setter poengsum til 5 om dette tilfredsstilles. Positiv grunnlag er mer enn 10kg-cm, og negativ mindre enn 8kg/cm, nøytralt om kraftmomentet er mellom 8-10kg-cm.

Utrekning bli vist bedre i Teknologi for Mekanikk 2, Gripeverktøyets fingre.

4.1.5 Masse.

For å tilfredsstille [US-050102] er det viktig at vi holder oss til lette komponenter for å ikke gjøre EE for tung. Løftekapasiteten til UR5 kan håndtere 5kg. Poengsummen er satt til 3.

Servomotorene som er blitt valgt ut har en masse mellom 35g-65g, noe som ikke utgjør en stor mengde masse, men jo lettere jo bedre. Grunnlaget blir satt til positiv hvis massen er under 40g og negativ med masse over 45g, nøytralt dersom massen er mellom 40-45g.

4.1.6 Størrelse.

Størrelsene på servomotorene blir satt til en poengsum på 2, siden EE må bli tilpasset til motorens størrelse, men mindre servomotorer er bedre da det har et relativt stor innvirkning på utsende til EE og dens fleksibilitet. Her setter vi grunnlag for større eller mindre enn 30.4cm³ i total volum, (40x20x38mm). Positiv dersom mindre enn 30.4cm³, negativ dersom større 30.4cm³. Nøytralt om 30.4cm³ i volum.

4.1.7 Hastighet

Hastighet har ingen betydning for EE i første omgang, de fleste servomotorene beveger seg 60 grader på under 0.5 sekunder som er mer enn raskt nok. Her setter vi ingen grunnlag, å vokter dem som nøytralt uansett hastighet siden vi har ikke behov for en rask bevegelse, dette kan programmeres med forsinkelser i «script» dersom bevegelsen skjer for raskt.

5. Pugh Matrise for servomotorer.

Ved bruk av pugh matrisen for utvelgelse av servomotorer har vi gitt en poengsum for hva vi mener er en viktig egenskap som servomotorene har, her er det gitt 0-5 poeng. Vi setter et vektgrunnlag slik at poengene enten trekker negativt eller positivt for å kunne gi en helhetlig total sum.

5.1 Vekting av pugh matrisene.

[Tabell 14] viser vekting av hva som er viktig for konseptet ved bruk av servomotorer.

Tabell 14: Vektgrunnlag for Servomotorer.

Vektgrunnlag: positiv(1), negativ(-1) eller nøytral verd(0) i:	Pris:	Spennings:	Elektrisk strøm:	Kraft-moment:	Masse:	Størrelse:	Hastighet:
1	<350kr	<12v	<200mA	>10kg-cm	<40g	<40x20x38 mm(30.4cm ³)	-
-1	>350kr	>12v	>200mA	<8kg-cm	>45g	>40x20x38 mm(30.4cm ³)	-
0	-	-	-	8-10kg-cm	40-45g	40x20x38 mm	-

Tabell 15: Hitec HS-5665MH

Navn	Hitec HS-5665MH HV 2S LiPo	Kriterier:	Poeng:	Vekting:
Pris:	NOK 460,-	Pris	4	-1
Kraftmoment:	10kg-cm	Spenning	2	1
Hastighet:	0.16/0.14s	Elektrisk strøm	2	-1
Spenning:	6.0/7.4V	Kraftmoment	5	0
Elektrisk strøm:	200mA idle 2.0A stall. 240mA idle, 3.0A stall	Masse	3	-1
Størrelse mm	41 x 20 x 38	Størrelse	2	0
Masse	60g	Hastighet.	1	0
		Sum av positive		2
		Sum av negative		9
		Sum av nøytrale		0
		Total:		-7
Referanse	https://www.servocity.com/html/hs-5665mh_servo.html#.VtgLHJzhBI9			
Kommentar	Denne servoen er middel sterk med tanke på at den kan yte 10kg-cm i kraftmoment, men Prisen er dyr i forhold til ytelse. Normal størrelse på kabinett, noe "tung"			

Tabell 16: Hitec HS-645MG

Navn	Hitec HS-645MG	Kriterier:	Poeng:	Vekting:
Pris:	NOK 345,-	Pris	4	1
Kraftmoment:	7.7/9.6kg-cm	Spenning	2	1
Hastighet:	0.24/0.20s	Elektrisk strøm:	2	-1
Spenning:	4.8/6V	Kraftmoment	5	0
Elektrisk strøm:	8.8mA/idle 350mA no load 9.1mA/idle 450mA no load	Masse	3	-1
Størrelse mm:	41 x 20 x 38	Størrelse	2	0
Masse	55.2g	Hastighet.	1	0
		Sum av positive		6
		Sum av negative		5
		Sum av nøytrale		0
		Total:		1
Referanse	https://www.servocity.com/html/hs-645mg_ultra_torque.html#.VtgLLZzhBI9			
Kommentar	Denne servoen er middels dyr, 9.6kg-cm og normal størrelse, kan være en potensiell komponent for bruk i gripeverktøy			

Tabell 17: Hitec HS-635HB

Navn	Hitec HS-635HB	Kriterier:	Poeng:	Vekting:
Pris:	NOK 250,-	Pris	4	1
Kraftmoment:	5.5 / 6 kg-cm	Spenning	2	1
Hastighet:	0.18 / 0.15 s	Elektrisk strøm	2	0
Spenning:	4.8/6.0v	Kraftmoment	5	-1
Elektrisk strøm:	N/A	Masse	3	-1
Størrelse mm:	41 x 20 x 39	Størrelse	2	-1
Masse:	50g	Hastighet	1	0
		Sum av positive		7
		Sum av negative		11
		Sum av nøytrale		0
		Total:		-4
Referanse	http://elefun.no/p/prod.aspx?v=1017			
Kommentar	Denne servoen er billig og grenser til mulig for å yte for svakt kraftmoment, størrelsen er normal.			

Tabell 18: Hitec HS-5646WP

Navn	Hitec HS-5646WP WaterProof HV	Kriterier:	Poeng:	Vekting:
Pris:	NOK 432,-	Pris	4	-1
Kraftmoment:	11.3 / 12.9kg-cm	Spenning	2	1
Hastighet:	0.20 / 0.18	Elektrisk strøm	2	0
Spenning:	6.0V/7.4V	Kraftmoment:	5	1
Elektrisk strøm:	N/A	Masse	3	-1
Størrelse mm:	41.8 x 21.0 x 40.0	Størrelse	2	-1
Masse:	61g	Hastighet	1	0
		Sum av positive		7
		Sum av negative		9
		Sum av nøytrale		0
		Total:		-2
Referanse	https://www.servocity.com/html/hs-5646wp_servo.html#.VtgLbJzhBI8			
Kommentar	Denne servoen er sterk, men på grensen for dyr, den er noe større i kabinett. Og "tung"			

Tabell 19: Hitec HS-5495BH

Navn	Hitec HS-5495BH HV 2S LiPo	Kriterier:	Poeng:	Vekting:
Pris:	NOK 265,-	Pris	4	1
Kraftmoment:	6.4 / 7.5kg-cm	Spenning	2	1
Hastighet:	0.17 / 0.15	Elektrisk strøm	2	0
Spenning	6.0V/7.4V	Kraftmoment:	5	-1
Elektrisk strøm:	N/A	Masse	3	1
Størrelse mm:	40 x 20 x 38	Størrelse	2	1
Masse:	44.5g	Hastighet.	1	0
		Sum av positive		11
		Sum av negative		5
		Sum av nøytrale		0
		Total:		6
Referanse	https://www.servocity.com/html/hs-5495bh_servo.html#.VtgLv5zhBI8			
Kommentar	Denne servoen er billig og lett, kraftmomentet er under grensen for hva vi ønsker til gripeverktøyet.			

Tabell 20: Hitec HS-485HB

Navn	Hitec HS-485HB	Kriterier:	Poeng:	Vekting:
Pris:	NOK 178,-	Pris	4	1
Kraftmoment:	4.8 / 6.0 kg-cm	Spenning	2	1
Hastighet:	0.22 / 0.18	Elektrisk strøm	2	1
Spenning:	4.8/6.0v	Kraftmoment	5	-1
Elektrisk strøm:	8mA idle, 150mA no load. 8.8mA idle, 180mA no load.	Masse	3	1
Størrelse mm:	39.8 x 19.8 x 38	Størrelse	2	1
Masse:	45g	Hastighet	1	0
		Sum av positive		13
		Sum av negative		5
		Sum av nøytrale		0
		Total:		8
Referanse	https://www.servocity.com/html/hs-485hb_servo.html#.VtgMA5zhBI8			
Kommentar	Denne servoen er ganske billig men grenser til for svakt kraftmoment, den er liten og lett.			

Tabell 21: Futaba S3010

Navn	Futaba S3010 1bb std	Kriterier:	Poeng:	Vekting:
Pris:	NOK 299,-	Pris	4	1
Kraftmoment:	5.2/6.5kg-cm	Spenning	2	1
Hastighet:	0.20/0.16s	Elektrisk strøm	2	0
Spenning	4.8/6.0v	Kraftmoment	5	-1
Elektrisk strøm:	N/A	Masse:	3	1
Størrelse mm:	40x10x38x	Størrelse	2	1
Masse:	41g	Hastighet	1	0
		Sum av positive		11
		Sum av negative		5
		Sum av nøytrale		0
		Total:		6
Referanse	https://www.servocity.com/html/s3010_std_ht_ball_bearing.html#.VtgMP5zhBI8			
Kommentar	Servomotoren har mange gode egenskaper, men har litt for lavt av ønsket kraftmoment.			

Tabell 22: Hitec HS-75BB

Navn	Hitec HS-75BB Retract servo	Kriterier:	Poeng:	Vekting:
Pris:	NOK 290,-	Pris	4	1
Kraftmoment:	6.6 / 8.2 kg-cm	Spenning	2	1
Hastighet:	0.45 / 0.34	Elektrisk strøm	2	1
Spenning:	4.8/6.0v	Kraftmoment	5	0
Elektrisk strøm:	1mA idle, 150mA no load. 1.6mA idle, 180mA no load.	Masse	3	1
Størrelse mm:	44 x 23 x 25	Størrelse	2	1
Masse:	35g	Hastighet	1	0
		Sum av positive		13
		Sum av negative		0
		Sum av nøytrale		0
		Total:		13
Referanse	https://www.servocity.com/html/hs-75bb_retract.html#.VtgMZJzhBl8			
Kommentar	Denne servoen er noe mindre enn de vanlige, den er lett og har en grei kraftmoment, prisen er også overkommelig.			

Tabell 23: JR NX588

Navn	JR NX588 Digital WV	Kriterier:	Poeng:	Vekting:
Pris:	NOK 278,-	Pris	4	1
Kraftmoment:	6,5/7,7kg-cm	Spenning	2	1
Hastighet:	0.20/0,19 s	Elektrisk krets	2	0
Spenning:	4.5V-8.5V	Kraftmoment	5	-1
Elektrisk strøm:	N/A	Masse	3	1
Størrelse mm:	33.5×19×38.5	Størrelse	2	1
Masse:	42g	Hastighet	1	0
		Sum av positive		11
		Sum av negative		5
		Sum av nøytrale		0
		Total:		6
Referanse	http://elefun.no/p/prod.aspx?v=29986			
Kommentar	Rimelig, normal størrelse, ønsker høyere kraftmoment.			

Tabell 24: Power HD-1214TH HV

Navn	Power HD-1214TH HV	Kriterier:	Poeng:	Vekting:
Pris:	NOK 559,-	Pris	4	-1
Kraftmoment:	16kg-cm	Spenning	2	1
Hastighet:	0.12s	Elektrisk strøm	2	0
Spenning:	7.4V	Kraftmoment	5	1
Elektrisk strøm:	N/A	Masse	3	-1
Størrelse mm:	40.3x20.2x37.2	Størrelse	2	1
Masse:	64g	Hastighet	1	0
		Sum av positive		9
		Sum av negative		7
		Sum av nøytrale		0
		Total:		2
Referanse	http://elefun.no/p/prod.aspx?v=25425			
Kommentar	Denne servoen er ganske kraftig, men noe dyr, noe "tung" størrelsen er litt mindre enn normalt.			

Tabell 25: Power HD-1235MG

Navn	Power HD-1235MG Large Scale HV	Kriterier:	Poeng:	Vekting:
Pris:	NOK 555,-	Pris	4	-1
Kraftmoment:	40kg-cm	Spenning	2	1
Hastighet:	0.18s	Elektrisk strøm	2	0
Spenning:	7.4V	Kraftmoment	5	1
Elektrisk strøm:	N/A	Masse	3	0
Størrelse mm:	59.5x29.5x55	Størrelse	2	-1
Masse:	N/A	Hastighet	1	0
		Sum av positive		7
		Sum av negative		6
		Sum av nøytrale		0
		Total:		1
Referanse	http://elefun.no/p/prod.aspx?v=25426			
Kommentar	Denne servoen er den sterkeste i sin pris, samtidig prisen kan være noe dyr for oss, kraftmomentet yter hele 40kg-cm noe som er ganske bra, den er ganske stor i forhold til de andre som blir nevnt.			

Tabell 26: Power HD-1501MG

Navn	Power HD-1501MG Analog	Kriterier:	Poeng:	Vekting:
Pris:	NOK 225,-	Pris	4	1
Kraftmoment:	17kg-cm	Spenning	2	1
Hastighet:	0.14s	Elektrisk strøm	2	0
Spenning:	4.5/6.0v	Kraftmoment	5	1
Elektrisk strøm:	N/A	Masse	3	0
Størrelse mm:	40.7x20.5x39.5	Størrelse	2	-1
Masse:	N/A	Hastighet	1	0
		Sum av positive		11
		Sum av negative		2
		Sum av nøytrale		0
		Total:		9
Referanse	http://elefun.no/p/prod.aspx?v=14492			
Kommentar	Denne servoen er billig og sterk, størrelsen er heller ikke så verst, skulle gjerne hatt tyngde på motor, denne er en sterk kandidat for bruk av motor i "finger"			

Tabell 27: Savöx Servo SC-0251MG

Navn	Savöx Servo SC-0251MG	Kriterier:	Poeng:	Vekting:
Pris:	NOK 329,-	Pris	4	1
Kraftmoment:	13/16kg-cm	Spenning	2	1
Hastighet:	0.20/0.18s	Elektrisk strøm	2	0
Spenning:	4.8/6.0v	Kraftmoment	5	1
Elektrisk strøm:	N/A	Masse	3	-1
Størrelse mm:	40.7 x 19.6 x 42.4	Størrelse	2	-1
Masse:	61g	Hastighet	1	0
		Sum av positive		11
		Sum av negative		5
		Sum av nøytrale		0
		Total:		6
Referanse	http://elefun.no/p/prod.aspx?v=20847			
Kommentar	Denne servoen er sterk og billig, men den er noe større og noe tyngre, denne kan være en mulig kandidat.			

Tabell 28: Savöx Servo SC-1256TG

Navn	Savöx Servo SC-1256TG	Kriterier:	Poeng:	Vekting:
Pris:	NOK 630,-	Pris	4	-1
Kraftmoment:	16/20kg-cm	Spenning	2	1
Hastighet:	0.18/0.15s	Elektrisk strøm	2	0
Spenning	4.8v/6.0v	Kraftmoment	5	1
Elektrisk strøm:	N/A	Masse	3	-1
Størrelse mm:	40.3 x 20.2 x 37.2	Størrelse	2	1
Masse:	52.4g	Hastighet	1	0
		Sum av positive		9
		Sum av negative		7
		Sum av nøytrale		0
		Total:		2
Referanse	http://elefun.no/p/prod.aspx?v=17552			
Kommentar	Denne servoen er ganske dyr, gir en god ytelse, samt størrelsen er noe mindre enn normalt.			

Tabell 29: Hitec HS-5485HB

Navn	Hitec HS-5485HB Digital	Kriterier:	Poeng:	Vekting:
Pris:	NOK 222,-	Pris	4	1
Kraftmoment:	5.2/6.4kg-cm	Spenning	2	1
Hastighet:	0.20/0.17s	Elektrisk strøm	2	-1
Spenning:	4.8/6.0V	Kraftmoment	5	-1
Elektrisk strøm:	8.8mA idle 400mA no load. 9.1mA idle, 500mA no load.	Masse	3	1
Størrelse mm:	39.88 x 19.81 x 37.85	Størrelse	2	1
Masse:	45.08g	Hastighet	1	0
		Sum av positive		11
		Sum av negative		7
		Sum av nøytrale		0
		Total:		4
Referanse	http://elefun.no/p/prod.aspx?v=16281			
Kommentar	Billig, noe svak, og noe mindre størrelse.			

5.2 Oppsummering av pugh matrise for servoer.

Fra pugh matrisen fra **[2.1 Pugh matrise for servomotorer]**. Ble servomotoren (Hitec HS-75BB) utvalgt som beste servomotor for bevegelse til EE tidlig i prosjektet. Det ble senere oppdaget en utfordring ved denne type servomotor modellen, vi hadde oversett en vesentlig egenskap denne typen servomotor hadde, dette var en type retract servomotor, det vil si at den er bygget slik at den beveger seg hele 180 grader ved ett pulssignal, slik at den ikke kan roteres til ønsket posisjon slik vi ønsker at det er på vanlig type servomotorer, en slik type servomotor blir gjerne brukt til landingsunderstell for radiostyrte fly eller helikoptre, siden den har den funksjonen at den bare trenger ett signal for å kunne rotere ut landingsutstyret eller rotere inn dette, er den genial for dette bruket, men dessverre ikke brukelig for vårt prosjekt.

Siden vi hadde startet å lage design passet for denne typen servomotor trengte vi en erstatting slik at vi har posisjoneringen vi så for oss at vi skulle ha.

Vi har derfor laget en ny pugh matrise på følgende kriterier som ble brukt for å velge servomotorer fra **[2.1 Pugh matrise for servomotorer]** der samme vektgrunnlag og poeng blir gjenopptatt fra **[Tabell 14]** i **[tabell 30]** ser dere tekniske spesifikasjonene og poengene servomotoren fikk.

Tabell 30. Power HS-1207 TG

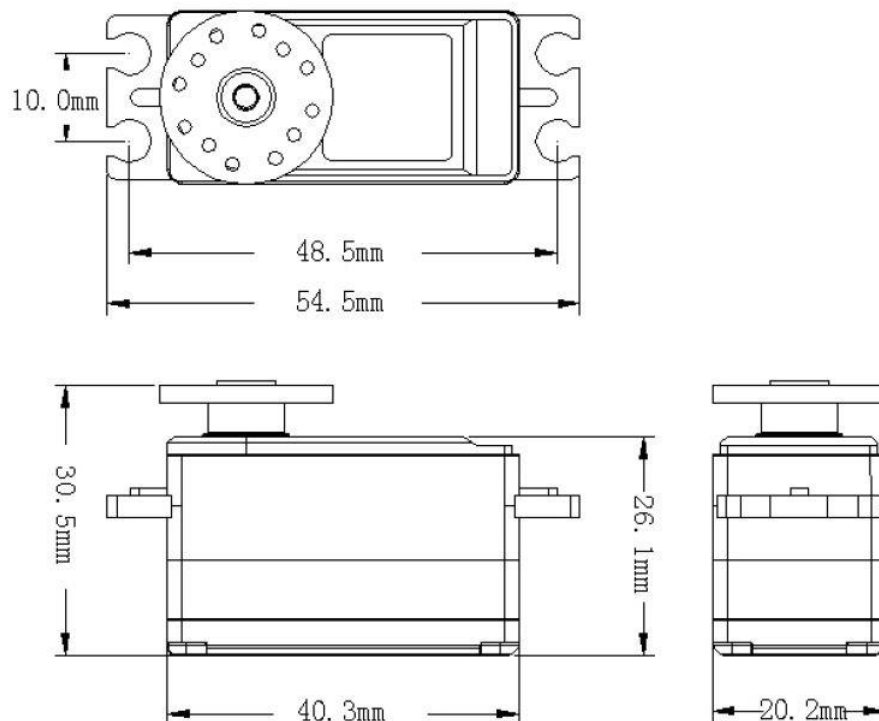
Navn	Power HD-1207TG Low Profile	Kriterier:	Poeng:	Vekting:
Pris:	NOK 495,-	Pris	4	-1
Kraftmoment:	8kg-cm	Spenning	2	1
Hastighet:	0.1s/0.09s	Elektrisk strøm	2	-1
Spenning:	4.8V/6.0V	Kraftmoment	5	0
Elektrisk strøm:	400mA no load, 2.3mA stall, 4mA idle. / 500mA no load, 2.5mA stall, 5mA idle.	Masse	3	-1
Størrelse mm:	40.3x20.2x26.1	Størrelse	2	1
Masse:	48g	Hastighet	1	0
		Sum av positive		4
		Sum av negative		9
		Sum av nøytrale		0
		Total:		-5
Referanse	http://elefun.no/p/prod.aspx?v=16281			
Kommentar	Denne servomotoren er nesten identisk med servomotoren fra [tabell 22] Hitec HS-75BB servoen, Prisen for servomotoren er høyere, og den har et høyere effektforbruk. Størrelsen er noe mindre, noe som gjør designet litt mindre enn ved bruk av HS-75BB. På grunn av høyt effektforbruk, høy pris, og massen er mer på grunn av denne har metaldrev noe som er positivt med tanke på lenger levetid. Alt i alt får servoen - 5 poeng i total poeng sum, noe som gjør at den faktisk ikke ville blitt med videre om den hadde fått være en kandidat i den tidligere pugh matrisen. (Denne servomotoren vil også gjøre det avgjørende at vi bruker en ekstern strømkilde for å kunne drive dem)			

7 Servomotor

Servomotoren som vil bli brukt til bevegelse i End Effector er en Servomotor fra Power, HS-1207 TG. Dette er en servomotor som har en lavere profil enn normale servomotorer, den yter god kraft som tilfredsstiller kravene fra [US-050101, US-051001, US-051101, US-044101] i [Prosjektplan]



Figur 1. HS-1207 TG servomotor.



Figur 2. 2D tegning av HS-1207 TG[x]

Tabell 31. Elektrisk spesifikasjoner HS-1207 TG

Spesifikasjoner:	4.8V	6.0V
Hastighet (uten last)	0.1 sek/60 grader	0.09 sek/60 grader
Drifts strøm (uten last)	400 mA	500 mA
Kraftmoment (låst posisjon)	7.0 kg-cm	8.0kg-cm
Drifts strøm (låst posisjon)	2.3 mA	2.5mA
Inaktiv strøm (ved stans)	4 mA	5 mA

Tabell 32. Mekaniske spesifikasjoner HS-1207 TG

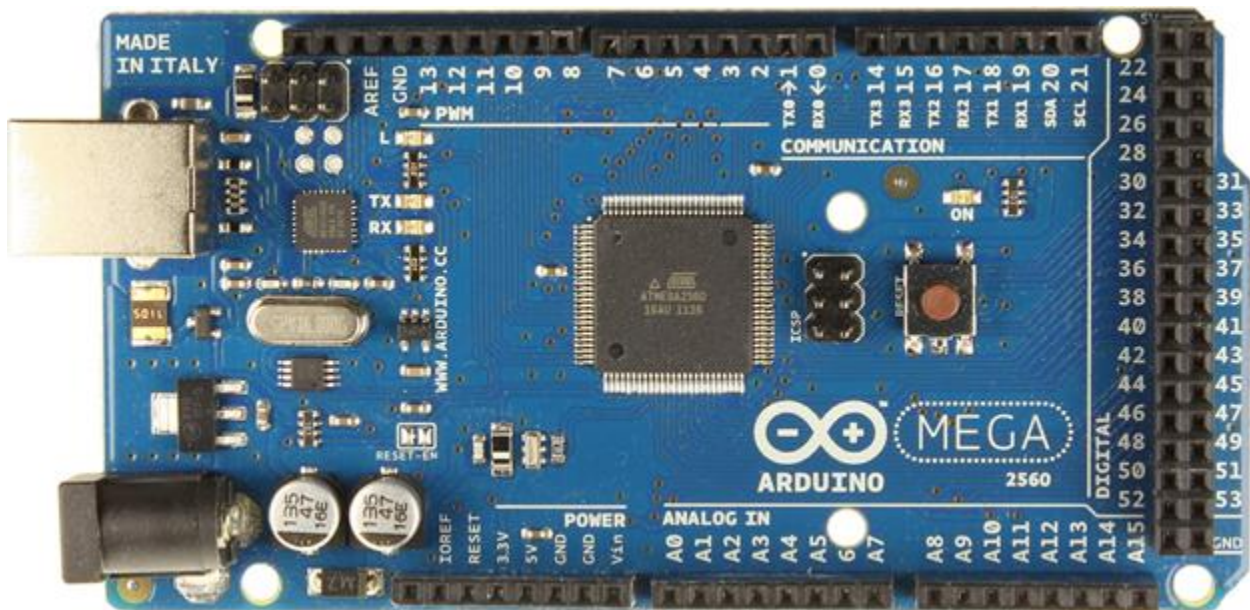
Drev materiale	Titanium shielded 7075 Alu
Maksimum vinkel	180°± 10°
Masse	48 ± 1 g
Lengde på tilkoblingsledning	300 ± 5 mm
Utgående aksel	25T 5.80 Diameter
Nedakselerasjon på drev	1/210

Tabell 33. Kontroll spesifikasjoner HS-1207 TG

Operasjons frekvens:	250HZ
Drift posisjon:	90° (1000 → 2000 ms)
Nøytral posisjon:	1500 ms
Død båndbredde:	4 ms
Roterende retning:	1500 → 2 000 ms
Pulsbredde rekkevidde:	800 → 2200 ms
Maksimum rotering:	165° 800 → 2200 ms

8 Kontrollere

For å kunne styre motorene til EE og motta signaler fra UR5, trenger vi en kontroller, det finnes mange type kontrollkort som kan styre forskjellige funksjoner, men siden vi er kjent med Arduino sine kontrollkort, og disse holder til applikasjonene vi skal utføre, ønsker vi bruke disse. Vi har tidligere brukt en Arduino ATmega 2560 kontrollkort. Denne har 54 digitale input/output pins hvor 15 kan brukes som PWM outputs. 16 analog inputs, den har en lagring på 256 KB. Arduino Mega 2560 kan bli programmert med Arduino software (IDE). [1]



Figur 3: Arduino ATmega 2560

Tekniske spesifikasjoner for Arduino ATmega 2560

Tabell 34 Tekniske spesifikasjoner for Arduino ATmega 2560 [1]

Microcontroller	ATmega2560
Driftsspenning	5V
Inngangsspenning (Anbefales)	7-12V
Inngangsspenning (Grense)	6-20V
Digitale I/O pins	54 (15 støtter PWM utganger)
Analoge input pins	16
DC strøm per I/O Pin	20mA
DC Strøm for 3.3 V Pin	50mA
Flashminne	256KB (8KB bootloader)
SRAM	8KB
EEPROM	4KB
Klokkefrekvens	16MHz
Lengde	101.52 mm
Bredde	53.3 mm
Masse	37 g

8.1 Arduino tilkobling

Arduino sine tilkoblinger i den komplette kretsen ligger i **[Brukermanual, side 26]**

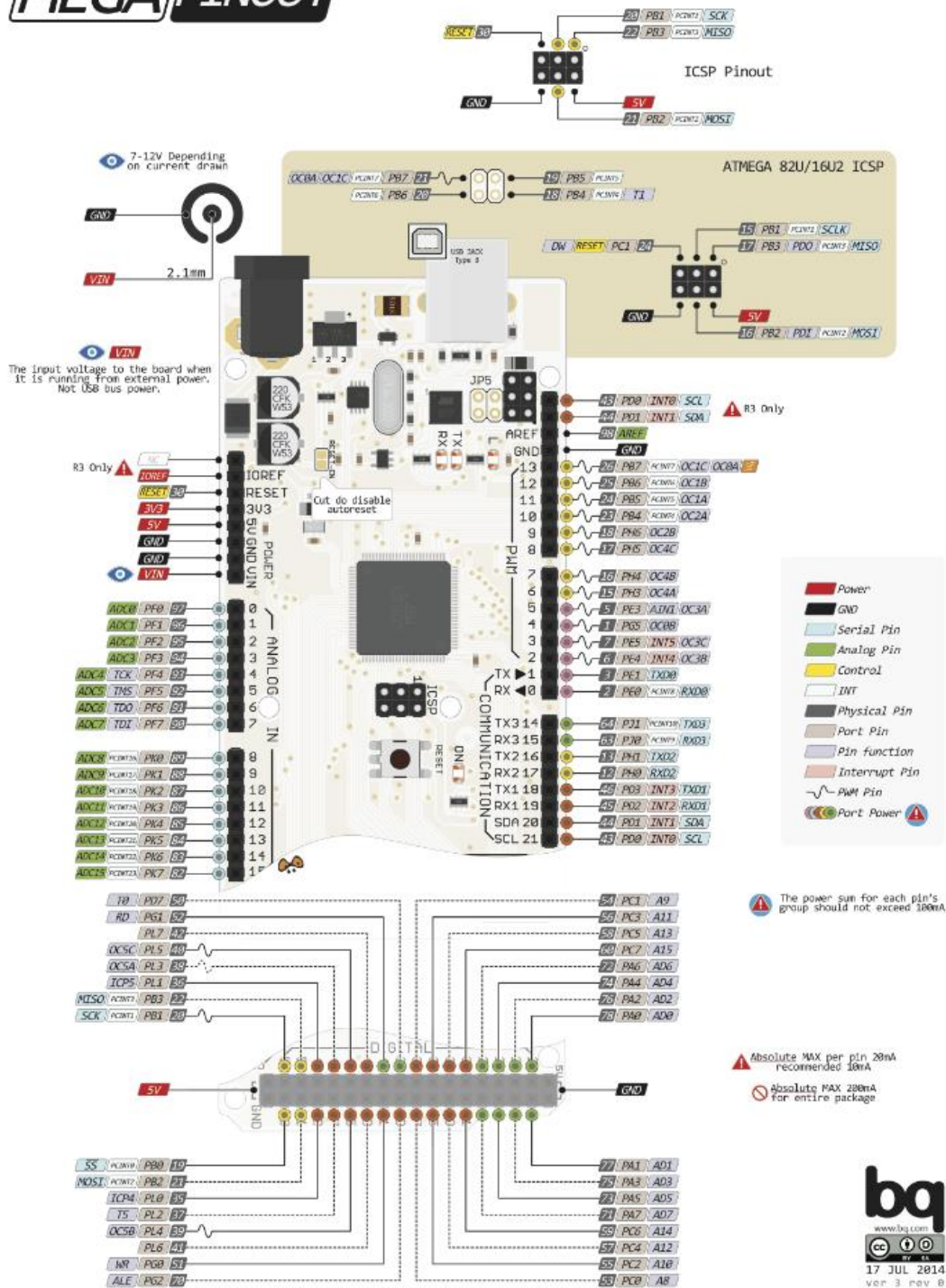
[Tabell 35] Viser en kort oversikt over de benyttede tilkoblingene og hvilke signaler som forventes å finne der. Komplette oversikt over Arduino Mega «pinout» kan leses av **[figur 4]**.

Tilkoblings oversikt for Arduino ATmega 2560

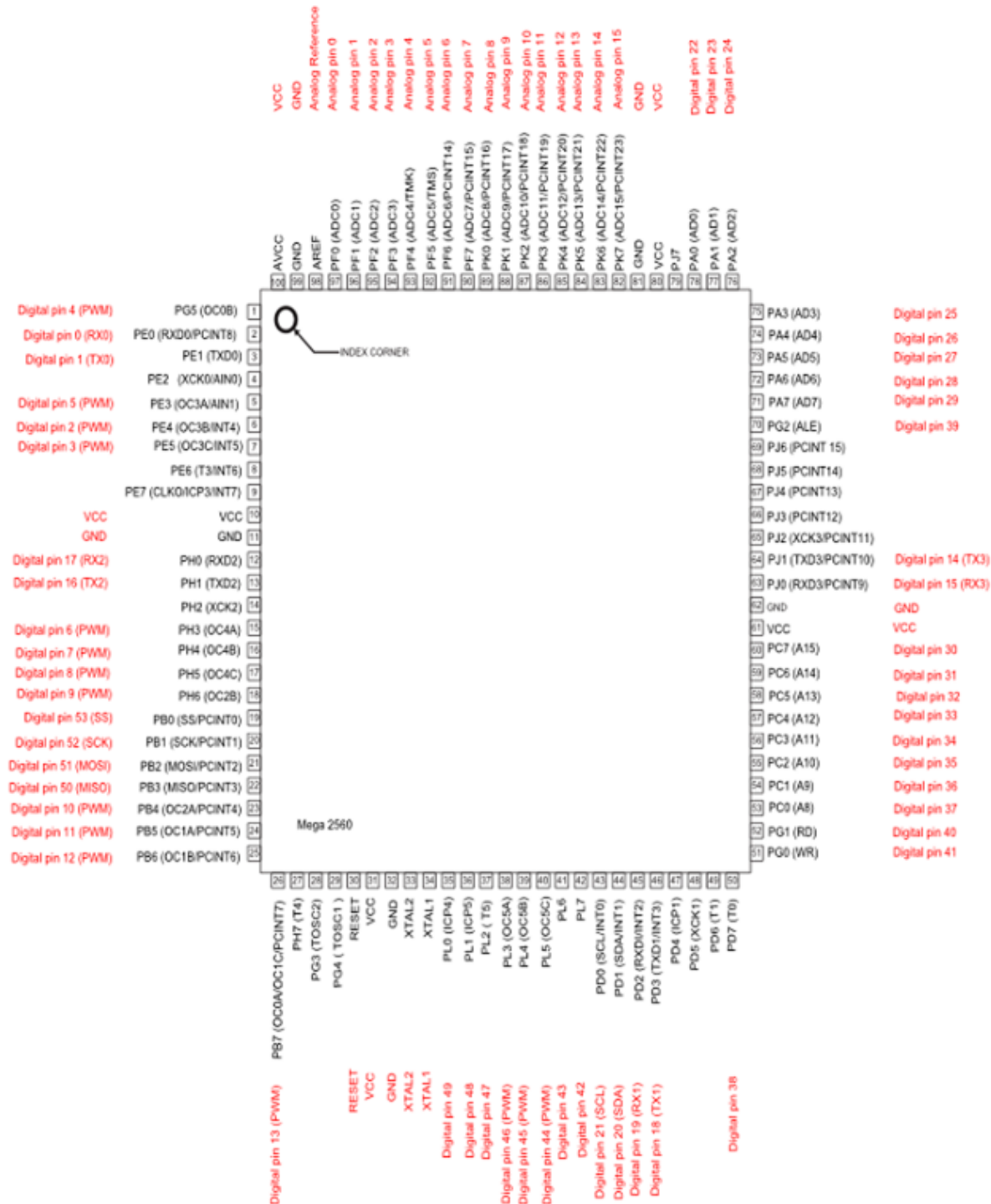
Tabell 35. Tilkoblings oversikt for Arduino ATmega 2560

Pin Navn	Pin Nr	Koblet mot	Signal
Vin	Vin	Koblet mot V+ på PSU	+6V
GND	GND	Koblet sammen med GND fra PSU og UR5 GND	0V /GND
D2	6	Koblet mot signal pin på ServoA	5V PWM SIGNAL
D3	7	Koblet mot signal pin på ServoB	5V PWM SIGNAL
D4	1	Koblet mot signal pin på ServoC	5V PWM SIGNAL
D5	5	Koblet mot signal pin på ServoD	5V PWM SIGNAL
D6	15	Koblet mot signal pin på ServoE	5V PWM SIGNAL
D7	16	Koblet mot signal pin på ServoF	5V PWM SIGNAL
D8	17	Koblet mot signal pin på ServoG	5V PWM SIGNAL
D9	18	Koblet mot signal pin på ServoH	5V PWM SIGNAL
D11	24	Koblet mot UR5 krets, (Digital out 8)	5V/GND
D18	46	Koblet mot UR5 krets, (Digital out 9)	5V/GND

MEGA PINOUT



Figur 4. Arduino Mega Pin oversikt [5]



Figur 5. ATmega 2560 IC PIN diagram. [2]

8.2 Arduino «Scetch»

Koden er skrevet i Arduino IDE, Arduino sitt eget IDE. Da vi ser at det vil være behov for tilpasninger i kode, blant annet for å legge til nye bevegelser, beholder vi en del test kode i kretsen. Alle linjer som det er kommentert med //UFT kan uten problemer kommenteres ut når kode skal legges på mikrokontrolleren, men da det som sendes ikke vises noe sted om man ikke er koblet til mikrokontrolleren med en USB kabel, er det ikke noe problem å la den ligge heller.

8.2.1 Bibliotek og variabler

```

9 #include <Servo.h>           //includes the servo library
10
11 Servo myservoA;             //defines my servos, A-H, used from the servo library
12 Servo myservoB;
13 Servo myservoC;
14 Servo myservoD;
15 Servo myservoE;
16 Servo myservoF;
17 Servo myservoG;
18 Servo myservoH;
19
20 int posA = 90;               // variables to store the servo positions
21 int posB = 90;               // Set to 90 as that is the starting positions for the servoes
22 int posC = 90;
23 int posD = 90;
24 int posE = 90;
25 int posF = 90;
26 int posG = 90;
27 int posH = 90;
28
29 int targetPosA = 0;          // variables to store the deseired servo positions
30 int targetPosB = 0;          // set to 0 as we havent set any desired position yet
31 int targetPosC = 0;
32 int targetPosD = 0;
33 int targetPosE = 0;
34 int targetPosF = 0;
35 int targetPosG = 0;
36 int targetPosH = 0;
37
38 int timer = 0;               //Variabel to store time, used for testing to keep track of numbers of loops -UFT
39 char servoSelect = 'X';      //char variabel to help the changePos() function to select servo
40
41 const int setPin = 18;        //defines constant variabel to be used in communication with URS, connect to Dig9 on Ur5, Pink cable IMPORTANT! - MUST BE PIN THAT SUPPORTS INTERRUPTS
42 const int varPin = 11;       //defines constant variabel to be used in communication with URS, connect to Dig8 on UR5, Blue cable
43
44 //defines volatile variables, can be used in interrupts
45 volatile int cnt = 0;         //counter to be used in inputAction()
46 volatile int bitZero = 0;     //Storage for read value of varPin
47 volatile int bitOne = 0;      //Storage for read value of varPin
48 volatile int bitTwo = 0;      //Storage for read value of varPin
49 volatile int bitThree = 0;    //Storage for read value of varPin
50 volatile int input = 0;       //Stores the combined value of the bit variabels, used in the switch case in loop()
51

```

Figur 6. Utdrag Arduino kode, variabler og bibliotek

«#include <Servo.h>» legger til servomotor biblioteket som vi skal benytte til å kontrollere de 8 servomotorene i EE. Biblioteket muliggjør at vi med en linje kode kan sette posisjonen til servoen til en bestemt grad fra 0-180.

For at servomotor-biblioteket skal fungere må det defineres noen servomotorene den kan jobbe mot, «Servo myservoA;» legger til en ny servomotor, som heretter kan refereres til som «myservoA.» Vi definerer da våre 8 servomotorer.

«posA-H» variablene legges globale da de skal kunne endres globalt, disse skal brukes til å holde styr på hver servomotor sin nåværende posisjon. De blir satt til 90 umiddelbart da dette er servomotorenes utgangsposisjon.

«targetPosA-H» variabler lages for å kunne legge ønsket posisjon for de enkelte servomotoren. Disse vil brukes senere i koden for å sette ønsket posisjon.

Variabelen «timer» lages for bruk under test, denne vil holde «track» på hvor mange ganger koden har kjørt. Hovedsakelig for å kunne visuelt sjekke at koden kjører, og at den ikke har hengt seg etter eventuelle endringer.

Vi definerer også en «servoSelect variabel», som vil brukes ved posisjons-endringer for å supplere koden med hvilken servomotor som skal jobbes mot.

Variablene «setPin» og «varPin» lages konstante, da disse ikke skal endres. Variablene bestemmes av hvilke «pins» som ønskes brukt til kommunikasjon mellom UR5 og Arduinoen. «setPin» settes til 18, da dette er en «interrupt pin» på Arduino Mega, om denne skal endres er det viktig at den endres til en annen «interrupt pin».

«Volatile» variabler benyttes da disse skal brukes i «interrupt» funksjonen, disse variablene skrives i minnet istedenfor i registeret. Variabelen «cnt» skal brukes til å telle antall «interrupt» som er «trigget.» «bitZero» – «bitThree» benyttes til å lagre de 4 avlesningene fra «varPin». «input» vil inneholde verdien som er mottatt fra UR5 sine digitale utganger.

8.2.2 «void setup()»

```
52 void setup()
53 {
54   Serial.begin(9600);           //starts Serial communication -UFT
55   pinMode(setPin, INPUT);       //defiens pinMode input/output for setPin
56   pinMode(varPin, INPUT);       //defiens pinMode input/output for varPin
57   myservoA.attach(2);           // attaches the servos on pin 2-9 to the servo object
58   myservoB.attach(3);           // values can be swapped in case of mixups in connecting the servos
59   myservoC.attach(4);           //IMPORTANT! - MUST BE PINS THAT SUPPORT PWM
60   myservoD.attach(5);
61   myservoE.attach(6);
62   myservoF.attach(7);
63   myservoG.attach(9);
64   myservoH.attach(8);
65   attachInterrupt(digitalPinToInterrupt(setPin), inputAction, CHANGE);
66   //attaches the interrupt to setPin, on change in signal will run the inputAction() function
67 }
```

Figur 7: Utdrag Arduino kode, «void setup()»

I «void setup()» definerer man vanligvis ting som man ikke planlegger å endre, eller start verdiene man ønsker å sette.

«mySerial.begin(9600)» setter kommunikasjons hastigheten til «mySerial» til å være «9600 baud». Dette er gjerne en standard verdi, og siden vi ikke har behov for noe raskere kommunikasjon enn dette beholder vi denne verdien.

Deretter defineres «setPin» og «varPin» til å være «input pins», dvs. Det forventes å motta signaler på disse pinnene.

Som en del av «setup» krever «sero.h» biblioteket at de definerte servomotorene kobles mot «PWM pinner». «myservoA.attach(2);» kobler servomotoren vi har kalt «myservoA» til «pwm» utgang på «pin 2». Dette gjentas så for alle servomotorer, med en unik «pin» til hver.

«attachInterrupt ()» brukes til å legge til «interrupt» funksjonen vi ønsker. Den kobles mot «setPin». Vi definerer videre at vi ønsker at den skal kjøre funksjonen «inputAction()» når den «trigges», og at den skal «trigge» ved et skift i signalet. Det er flere muligheter å velge mellom når det gjelder å bestemme når «interrupt» skal «trigges», men vi ønsker å benytte oss av denne da vi ser for oss at dette vil skape den raskete og mest stabile kommunikasjonen.

8.2.3 «Void loop()»

```

69 void loop()
70 {
71 /*
72 Serial.println("-_____"); //UFT
73 Serial.print("Timer = "); //UFT
74 ++timer; //adds 1 to the timer counter. //UFT
75 Serial.println(timer); //UFT
76 stateTest(); //writes test data to the screen. //UFT
77
78
79 */
80 long randomNumber = random(0, 5); //For "easy" testing.. MUST BE COMMENTED OUT WHEN IN PROPPER USE!!
81 Serial.print("long = ");
82 Serial.println(randomNumber);
83 input = (int)randomNumber;
84 Serial.print("input = ");
85 Serial.println(input);
86 */
87
88
89
90
91 switch (input) //Switch case, to determind action based on the input variabel.
92 {
93     case 0: //case for input = 0
94         Serial.println("Case0"); //UFT
95         break;
96     case 1: //case for input = 1
97         Serial.println("Case1"); //UFT
98
99         targetPosA = 35;
100         servoSelect = 'A';
101         posA = changePos(posA, targetPosA, servoSelect);
102
103         targetPosB = 35;
104         servoSelect = 'B';
105         posB = changePos(posB, targetPosB, servoSelect);
106
107

```

Figur 8: Utdrag Arduino kode, «void loop()»

«void loop()» domineres av en «switch/case» funksjon som benytter seg av innholdet i «input» variabelen til å velge hvilken «case» som skal gjennomføres. Når «input» er lik 0 vil EE forholde seg rolig, når den er 1, vil EE gjøre nødvendige bevegelser for å åpne hånden.

For å endre posisjonen på en servomotor benytter man seg av «changePos» funksjonen som forklares i kapittel [8.2.7] Man må her passe på at man først setter «targetPosX», og «servoSelect» til ønskede verdier. «targetPosX» skal inneholde ny ønsket posisjon. «servoSelect» skal settes til ønsket servomotor bokstav som vist i eksempelet i figur 9

```

102     targetPosA = 35;
103     servoSelect = 'A';
104     posA = changePos(posA, targetPosA, servoSelect);

```

Figur 9: Utdrag Arduino kode, Skifte av servo posisjon

«changePos()» vil da ta meg seg nåværende posisjon, ønsket ny posisjon og servomotor bokstaven inn i funksjonen, og returnere ny nåværende posisjon når funksjonen har gjort sin jobb.

Man kan endre rekkefølge på servomotorenes handlinger ved å endre rekkefølge på handlingene i koden. Om man trenger at en servomotor beveger seg flere ganger er det bare å legge den til flere ganger i koden.

```

213 default:                                     //default action, if input does not match any of the predetermined cases.
214     Serial.print(input);                       //UFT
215     Serial.println(" : Er ukjent input, interrupt tæmp disabled"); //UFT
216     detachInterrupt(digitalPinToInterrupt(setPin)); //disables the interrupt.
217     delay(2000);                               //delay for "timeOut"
218     printStatusAndReset();                     //Prints the status unless that is commentet out, reattach the interrupt and resets input and cnt.
219     Serial.println("Klar for nye oppdrag");      //UFT
220
221     break;

```

Figur 10: Utdrag Arduino kode, «default action»

Om «input» variabelen ikke «matcher» noen av «casene» definert i «switch/case» løkken, vil den gjøre en standard handling. Dette innebærer å «detach» «interrupt» funksjonen, ta en pause på 2000 millisekunder og kjøre «printStatusAndReset()» funksjonen. Dette gir EE en liten «timeout» og resetter den.

Hver «case», med unntak av «case 0» må avsluttes med «printStatusAndReset()» funksjonen da denne tilbakestiller «input» variabelen, og reaktiverer «interrupt» funksjonen.

8.2.4 «Void inputAction()»

```

232 void inputAction()                          //inputAction that trigges on changing signal on the sigPin
233 {
234     Serial.println("--Interrupt Triggered"); //UFT
235
236     if (cnt == 0) bitZero = digitalRead(varPin); //Reads the current value of varPin to the variabel to be used later
237     if (cnt == 1) bitOne = digitalRead(varPin); //Reads the current value of varPin to the variabel to be used later
238     if (cnt == 2) bitTwo = digitalRead(varPin); //Reads the current value of varPin to the variabel to be used later
239     if (cnt == 3) bitThree = digitalRead(varPin); //Reads the current value of varPin to the variabel to be used later
240
241     Serial.print("--Counter : ");             //UFT
242     Serial.println(cnt);                     //UFT
243     ++cnt;                                   //adds one to the cnt counter
244
245     if (cnt >= 4)                             //if the count of cnt reached 4, resets the cnt and creates the input value
246     {
247         cnt = 0;                               //RESTS the cnt variabel
248         Serial.println("--COUNTERRESET");      //UFT
249         Serial.println(bitZero);               //UFT
250         Serial.println(bitOne);               //UFT
251         Serial.println(bitTwo);               //UFT
252         Serial.println(bitThree);             //UFT
253         input = 1 * bitZero + 2 * bitOne + 4 * bitTwo + 8 * bitThree; //creates the value for the int variabel to use in loop.
254         Serial.print("INPUT = ");             //UFT
255         Serial.println(input);                 //UFT
256     }
257     Serial.println("-----");                //UFT
258 }

```

Figur 11: Utdrag Arduino kode, «void inputAction()»

«inputAction()» funksjonen kjøres hver gang Arduino detekterer et skifte på «setPin» inngangen. Avhengig av hvilken verdi «cnt» har vil den lese av verdien på «varPin» og lagre denne i en av de fire bit variablene, før den øker «cnt» verdien med 1.

Når den har gjort dette 4 ganger, vil «cnt» variabelen ha økt nok til å «trigge» nest steg av «interrupt» koden. Denne delen av koden resetter «cnt» variabelen, og bruker de 4 bit variablene til å generere et tall mellom 0 og 15 i «input» variabelen. «input» variabelen får sin verdi ved å regne ut tall verdien til den binære koden bit variablene gir, der «bitZero» fungerer som «(least significant bit)» «LSB», og «bitThree» fungerer som «(Most significant bit)» «MSB».

8.2.5 «void printStatusAndReset()»

```
260 void printStatusAndReset()
261 {
262     /*
263     Serial.println("-----");           //UFT
264     Serial.print(" posA : ");           //UFT
265     Serial.println(posA);               //UFT
266     Serial.print(" posB : ");           //UFT
267     Serial.println(posB);               //UFT
268     Serial.print(" posC : ");           //UFT
269     Serial.println(posC);               //UFT
270     Serial.print(" posD : ");           //UFT
271     Serial.println(posD);               //UFT
272     Serial.print(" posE : ");           //UFT
273     Serial.println(posE);               //UFT
274     Serial.print(" posF : ");           //UFT
275     Serial.println(posF);               //UFT
276     Serial.print(" posG : ");           //UFT
277     Serial.println(posG);               //UFT
278     Serial.print(" posH : ");           //UFT
279     Serial.println(posH);               //UFT
280     */
281     delay(100);                          //pause to give the code time to work
282     cnt = 0;                             //resets the cnt var
283     input = 0;                           //resets the input var
284 }
```

Figur 12: Utdrag Arduino kode, void printStatusAndReset()

Utskrift av status delen av funksjonen kan skrive ut en liste med nåværende posisjon for alle servomotorer, men dette vil bare være synlig når man har koblet USB kabel til mikrokontrolleren.

«cnt» og «input» variablene settes begge til 0 slik at mikrokontrolleren skal være klar til ny input.

8.2.6 «void stateTest()»

```

285 void stateTest()                                // Function for testing, monitors the two input pins setPin, and varPin. //UFT
286 { //UFT
287   int testSetPin = digitalRead(setPin);           //UFT
288   Serial.print("Set pin is : ");                 //UFT
289   Serial.println(testSetPin);                     //UFT
290   int testVarPin = digitalRead(varPin);           //UFT
291   Serial.print("Var pin is : ");                  //UFT
292   Serial.println(testVarPin);                     //UFT
293 }

```

Figur 13: Utdrag Arduino kode, «void stateTest()»

Funksjonen benyttes kun til testing, når den kalles vil den gi en utskrift til skjerm med status detektert for «setPin» og «varPin». Kan kommenteres ut under normal drift, eventuelt fjernes helt, men er beholdt i koden fordi den har verdi for forståelse for de som eventuelt skal jobbe videre med systemet.

8.2.7 «int changePos()»

```

295 int changePos(int pos, int targetPos, char x)      //function to set servo pos equal to desired targetPos inputs(current pos ex: posA, desired pos ex: targetPosA,
296 {                                                  //while loop, will run untill pos value matches targetPos value
297   while (pos != targetPos)
298   {
299     if (pos < targetPos)                            //if, runs if pos is less than targetPos
300     {
301       ++pos;                                        //add 1 to the value of pos, moving it one step closer to targetPos
302     }
303
304     if (pos > targetPos)                            //if, runs if pos is bigger than targetPos
305     {
306       --pos;                                        //subtracts 1 to the value of pos, moving it one step closer to targetPos
307     }
308
309     if (x == 'A') myservoA.write(pos);              //Sets servo to pos measured in deg, depending on value of x.
310     if (x == 'B') myservoB.write(pos);
311     if (x == 'C') myservoC.write(pos);
312     if (x == 'D') myservoD.write(pos);
313     if (x == 'E') myservoE.write(pos);
314     if (x == 'F') myservoF.write(pos);
315     if (x == 'G') myservoG.write(pos);
316     if (x == 'H') myservoH.write(pos);
317
318     delay(15);
319
320     if (x == 'A') Serial.print(" TEMPosA : ");     //UFT
321     if (x == 'B') Serial.print(" TEMPosB : ");     //UFT
322     if (x == 'C') Serial.print(" TEMPosC : ");     //UFT
323     if (x == 'D') Serial.print(" TEMPosD : ");     //UFT
324     if (x == 'E') Serial.print(" TEMPosE : ");     //UFT
325     if (x == 'F') Serial.print(" TEMPosF : ");     //UFT
326     if (x == 'G') Serial.print(" TEMPosG : ");     //UFT
327     if (x == 'H') Serial.print(" TEMPosH : ");     //UFT
328     Serial.println(pos);                           //UFT
329   }
330
331   return pos;
332 }

```

Figur 7. Utdrag Arduino kode, «int changePos ()»

Funksjonen som brukes til å endre posisjonen til en servomotor. Denne funksjonen er avhengig av tre «input» variabler.

1. «pos» variabelen må inneholde servomotorens nåværende posisjon.
2. «targetPos» variabelen må inneholde ønsket ny posisjon.
3. x variabelen må inneholde bokstaven knyttet til servomotoren som skal endre posisjon.

Funksjonen kjører en «while» løkke, som vil kjøre så lenge «pos» og «targetPos» variablene er ulike. Det første som skjer i «while» løkken er at «pos» variabelen endres 1 steg opp, eller ned avhengig av om verdien i «targetPos» er større eller mindre enn verdien i «pos».

Deretter endres posisjonen til servomotor ved å bruke «myservoX.write(pos);» kommandoen. Hvilken motor som endrer posisjon er avhengig av verdien av x variabelen. Det er derfor viktig at denne samsvarer med hvilken servomotor man ønsker å styre.

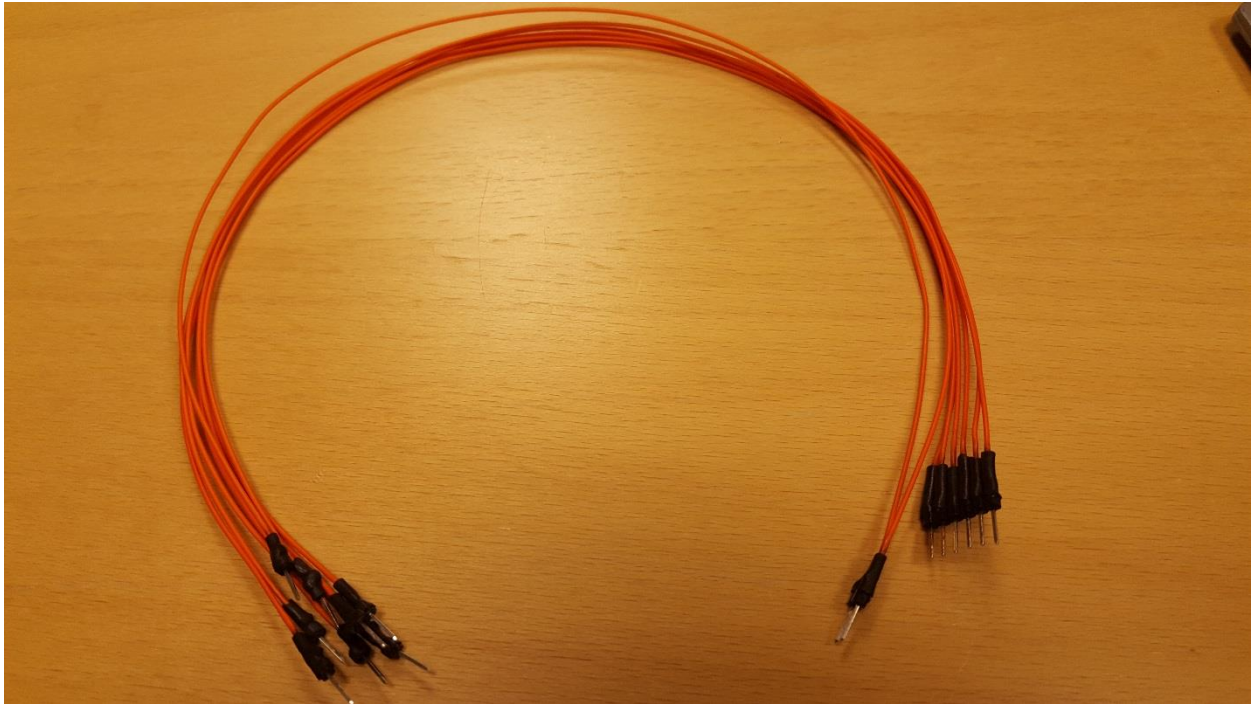
Pausen som følger ved bruk av «delay()» funksjonen kan varieres for å endre hastigheten på bevegelsene på servomotorene.

Den nye posisjonen skrives til skjerm og kan leses av i Arduino IDE «Serial monitor» vinduet om USB kablet er koblet til. Dette er ikke nødvendig for funksjonaliteten, men veldig greit å bruke under test.

«While» løkken vil repetere frem til «pos» variabelen har samme verdi som «targetpos» variabelen. Når løkken avsluttes returnerer funksjonen den nye verdien av «pos» tilbake til «void loop()».

8.3 Signal ledere

For at vi skal kunne gi signaler til servomotorene må vi ha ledere, vi har derfor laget 8 signal ledere, av AVG 25 kabler som skal kobles på Arduino fra «PWM output pin» 2 til 9 til Servomotorene.



Figur 8. Signal ledere

Tabell 36. Teknisk spesifikasjoner

Kabel type:	Lengde:	Farge:	Antall:
AVG 24	0.4 meter	Oransje	8

9. Strømkilde

Ved bruk av komponenter som forbruker mer effekt enn hva UR5 roboten kan forsyne blir en ekstern strømkilde tatt i bruk.

9.1 CPX200

Det blir brukt en CPX200 DUAL 35V 10A power supply fra Thurlby Thandar Instruments som er i KPS sitt eie, for å kunne forsyne Servomotorene og Arduinoen med strøm.



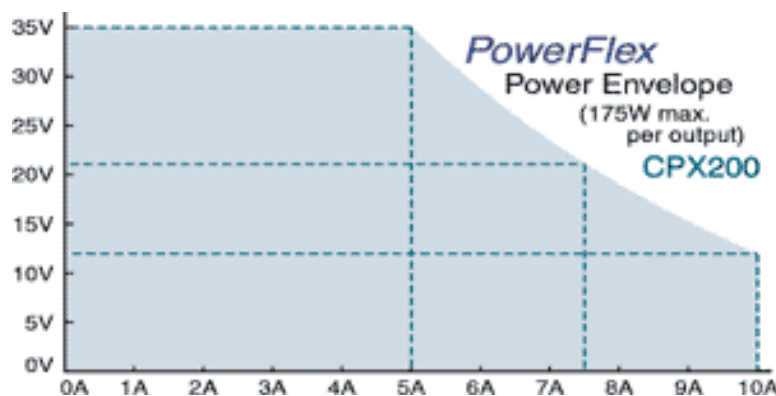
Figur 9. CPX 200 power supply

Tabell 37. Teknisk spesifikasjoner CPX 200 power supply [3]

Spenningsområdet	Strømområdet	Effektområdet
0V-35V	0A-10A	Opptil 175W

9.2 Bruk av Strømforsyning

Ved bruk av CPX200 må man følge en graf for kombinasjon av strøm og spenning for effekten den leverer, ved vårt bruk trenger vi 6V spenning og 1A per servomotor, ved bruk av alle 8 servomotorene, vil strømforsyningen gi ca. 50W forsyningen til End effector.



Figur 10. Effekt fleksibilitet [3]

9.3 Elektrisk ledere

For å forsyne End effector med strøm har vi laget kabler tilpasset bruket vi skal ha dem til. Disse kablene blir brukt til tidlig testing, og er ikke endelig produkt for et ferdigstilt system, ønsket er at vi bruker en egen ekstern strømkilde utenom en lab strømkilde.

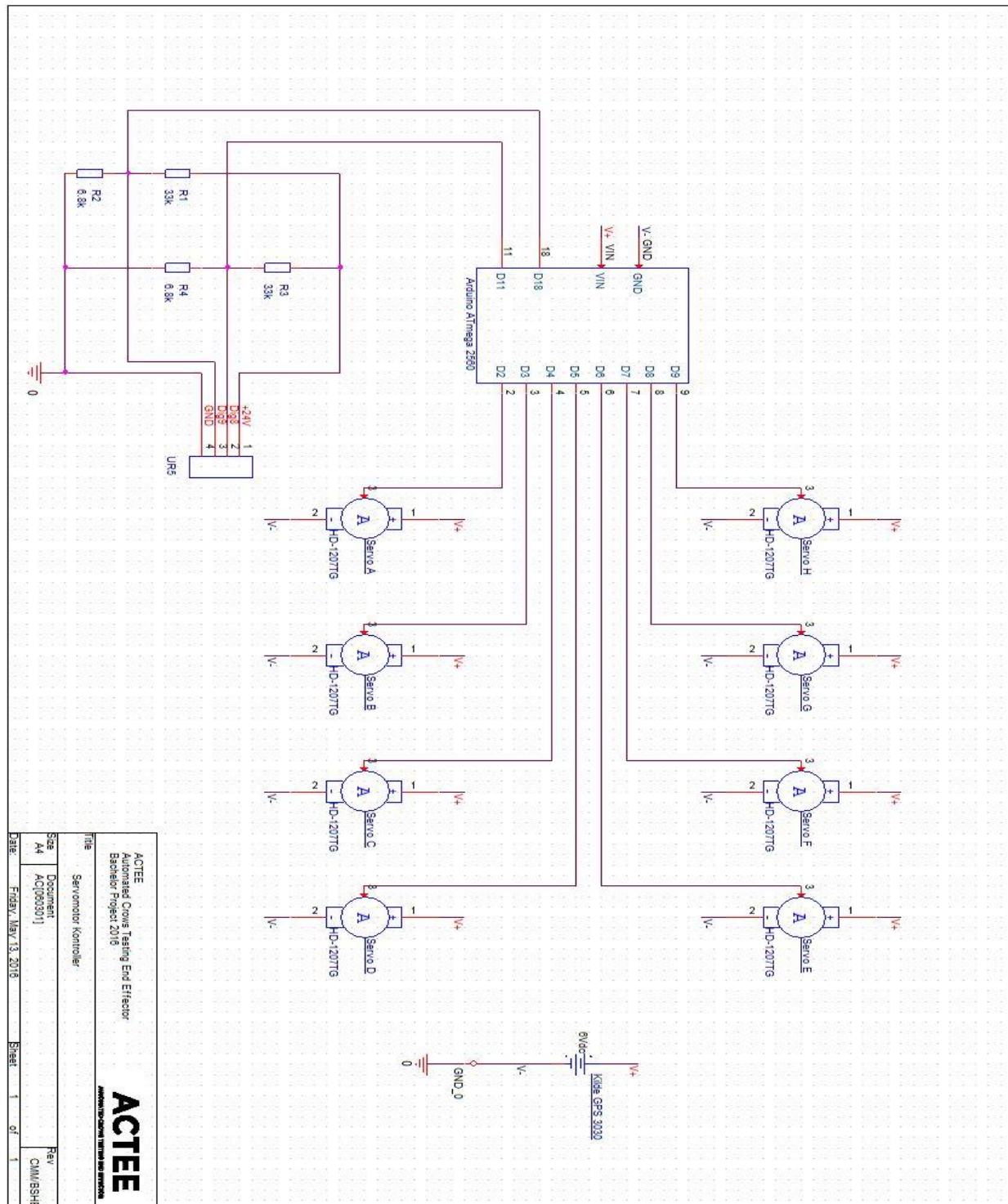


Figur 11. AVG 16 kabler for strømforsyning

Tabell 38. Teknisk Spesifikasjoner AVG 16 kabel.

Kabel type	Lengde:	Farge:	Antall:
AVG 16	3 meter	Rød	9
AVG 16	3meter	Sort	9

9.4 Koblingsskjema for End Effector



Figur 12. Koblingsskjema for EE.

10. Universal robot 5. UR5.

UR5 er en 6-akset sylindrisk robotarm. Hvert ledd kan rotere 360° ved hjelp av innebygde servomotorer hvor kraften overføres direkte ved rotasjon. Den har en løftekapasitet på 5 kg totalt med verktøyets egenvekt og objekt som skal løftes. For vår oppgave er det ikke relevant å beregne at roboten skal kunne løfte noe, men vi har likevel fokusert på at systemets totale masse skal holdes så lav som mulig.

Vår oppgave er derfor å lage et verktøy som kan gjøre robot armen fra Universal Robots På grunn av den store variasjonen i oppgaver for en hånd, blir ofte «End Effectors» eller «End-of-Arm-Tools» spesialtilpasset sitt formål.



Figur 13 UR5. Universal robot 5.

Teknisk spesifikasjoner til UR5

Tabell 39. Tekniske spesifikasjoner til UR5 [4]

6 – Axis robot arm with a working radius of 850 mm / 33.5 in	
Weight:	18.4 kg / 40.6 lbs
Payload:	5 kg / 11 lbs
Reach:	850 mm / 33.5 in
Joint ranges:	+/- 360° on all joints
Speed:	Joint: Max 180°/sec. Tool: Approx. 1 m/sec. / Approx 39.4 in/sec.
Repeatability:	+/- 0.1 mm / +/- 0.0039 in (4mil)
Footprint:	Ø149 mm / 5.9 in
Degrees of freedom:	6 rotating joints
Control box size (WxHxD):	475 mm x 423 mm c 268 mm / 18.7 x 16.7 x 10.6 in
I/O ports:	10 digital in, 10 digital out, 4 analogue in, 2 analogue out
I/O power supply:	24 V 1200 mA in control box and 12 V/24 V 600 mA in tool
Communication:	TCP/IP 100Mbit: IEEE 802.3u, 100BASE-TX Ethernet socket & Modbus TCP
Programming:	Polyscope graphical user interface on 12 inch touchscreen with mounting
Noise:	Comparatively noiseless
IP classification:	IP54
Power consumption:	Apporx: 200 watts using a typical program
Collaboration operation:	Tested in accordance with sections 5.10.1 and 5.10.5 of EN ISO 10218-1:2006
Materials:	Aluminium, ABS plastic
Temperature:	The robot can work in a temperature range of 0-50°C
Power supply:	100-240 VAC, 50-60 Hz
Calculated Operating Life:	35,000 Hours

10.1 UR5 «software program.»

Språket som blir brukt til å programmere roboten er noe som heter «URScript Programming Language». Det er tre måter å kontrollere roboten: «The Graphical User-Interface Level», «the Script Level» og «the C-API Level». «URScript» er robotens programmerings språk som blir brukt til å styre roboten på «Script Level». Vi bruker forskjellige «scripts» for å kontrollere bevegelsene til roboten. Når roboten skal programmeres brukes «Polyscope» (det grafiske brukergrensesnittet). Dette er en berøringsskjerm som er koblet til robotens datamaskin. [4]

10.2 UR5-Arduino «interface»

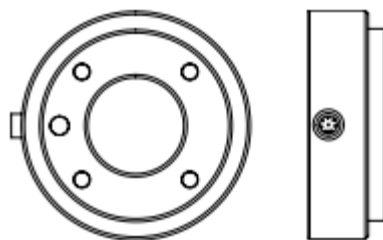
UR5 roboten har en egen «tool I/O» kontakt bestemte vi oss for å benytte oss av denne for kommunikasjon mellom UR5 og EE. Dette kapittelet vil forklare litt prosessen rundt dette og beslutningene som ble tatt i forbindelse med dette.

10.2.1 UR5 «tool I/O»

UR5 har 8 ledere som man kan koble seg til på verktøy enden av roboten. Oversikt over disse kan man se i [tabell 40]

Tabell 40. «Tool I/O» signal oversikt

Farge	Signal
RØD	0V/(GND)
GRÅ	0V/12V24V(POWER)
BLÅ	Digital Output 8 (DO8)
ROSA	Digital Output 9 (DO9)
GUL	Digital Input 8 (DI8)
GRØNN	Digital Input 9 (DI9)
HVIT	Analog Input 2 (AI2)
BRUN	Analog Input 3 (AI3)



Figur 14. Illustrasjon av plassering av «tool I/O connector»

Den grå lederen kan man velge verdi på med å bruke UR5 sin medfølgende programvare, da dette gjøres med et tastetrykk på en berøringsskjerm velger vi å tilrettelegge kretsen for bruk av 24V, slik at det ikke vil oppstå noen problemer om noen ved et uhell skifter utgangen til 24V.

De digitale «output» pinnene vi planlegger å benytte til kommunikasjonen har to tilstander, «TRUE» og «FALSE». Når en digital utgang er satt til «TRUE», vil den kobles mot GND eller 0V. Om den settes til «FALSE» blir den satt i en åpen tilstand. Dette så vi at vi måtte ta hensyn til når kretsen skulle designes.

De digitale «input» portene har innebygde «pull-down» motstander, dette betyr at de vil registreres som «LOW» når det ikke er noe input koblet til. Signaler under 2V vil registreres som «LOW» og signaler over 5.5V vil registreres som et høyt signal.

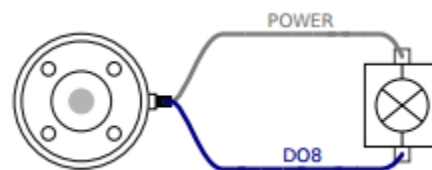
10.3 UR5-Arduino krets

For å muliggjøre kommunikasjonen mellom disse to enhetene må vi ha en krets som senker 24V signalene ut fra UR5 ned til 5V som kan tolkes av Arduino sine digitale «I/O» pinner.

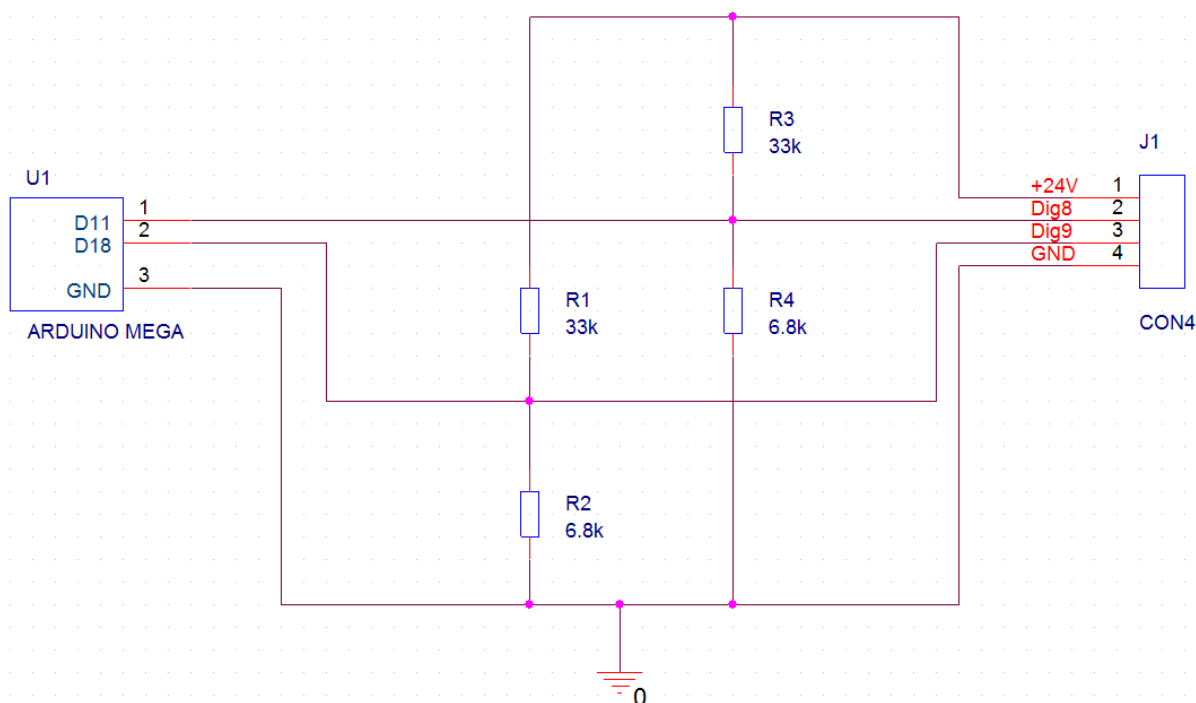
Vi forsøkte først med en enkel spenningsdeler koblet slik som Brukermanualen [6] til UR5 roboten anbefalte å benytte de digitale utgangene. Eksempelet deres er vist i [figur 22.]

Vi var skeptiske til denne løsningen i utgangspunktet, da dette ville bety at inngangene på Arduino mikrokontrolleren ble liggende flytende og dermed være sårbar for falske signaler.

Dessverre fikk vi rett i dette, Arduino sine utganger tolket et høyt signal uten problemer, men når vi forsøkte å sette de lavt, med å sette UR5-Digital «Output» til «FALSE», var det helt tilfeldig hvilke status vi kunne lese fra Arduino sine innganger. Vi ble tvunget til å tenke nytt.



Figur 22: Eksempel på Digital «Output» bruk



Figur 23: Krets UR5-Arduino «interface»

Løsningen vi fant kan man se i [figur 23]. Kretsen sikrer at Arduino sine innganger alltid har en referanse mot GND, og eliminerer derfor den flytende egenskapen som skapes av mangel på dette.

Når Digital «Output» fra UR5 settes til «FALSE», vil kretsen behandle det som at den er koblet fra, 24V vil legge seg over spennings-deleren vår og verdiene på motstandene sikrer at det vil ligge ca. 5V mellom inngangene på Arduino, og GND. Dette resulterer i at Arduino vil lese et høyt signal, eller en logisk 1 på denne porten.

Når Digital «Output» settes til «TRUE», vil det legge seg 24V over den øverste motstanden i spennings-deleren, og inngangene på Arduino vil detektere 0V eller en logisk 0.

Dette betyr at kretsen nå kan benyttes mot Arduino, men man må reversere logikken i hodet, en «FALSE» på UR5, vil resultere i en logisk 1 på Arduino, mens en «TRUE» på UR5, resulterer i en logisk 0 på Arduino.

10.4 Kommunikasjon

De digitale utgangene på UR5 står normalt på «FALSE», med dette som utgangspunkt bestemte vi oss for å bruke disse utgangene sammen med Arduino til å serielt overføre et 4-bits signal. Dette muliggjør at vi kan sende 16 «kommandoer» til Arduino for styring av EE. Fordelen med dette er at det ganske enkelt kan utvides ved å legge til flere bit i kommunikasjonen.

Arduino sin kode vil reagere på endringer i signalet på Digital Output 8, når denne skifter verdi, vil Arduino lese nåværende verdi fra Digital «Output» 9 og lagre denne i en buffer.

For å kommunisere til Arduino må man derfor lage en sekvens på UR5 som endrer «output» etter et satt mønster.

Set verdien på Digital «Output» 9 til ønsket verdi av «LSB». «FALSE» for 1, «TRUE» for 0.

Digital «Output» 8 endres til «TRUE».

Endre, eller behold verdi på Digital «Output» 9. «FALSE» for 1, «TRUE» for 0.

Digital «Output» 8 endres til «FALSE».

Endre, eller behold verdi på Digital «Output» 9. «FALSE» for 1, «TRUE» for 0.

Digital «Output» 8 endres til «TRUE».

Endre, eller behold verdi på Digital «Output» 9, setter verdien av «MSB». «FALSE» for 1, «TRUE» for 0.

Digital «Output» 8 endres til «FALSE».

Arduino vil oversette innputten til en tallverdi og benytte denne til å gjøre en bestemt handling. Dette kan leses mer om i kapittel **[8.2.4]**

11 Konklusjon

Dette kapittelet gir et sammendrag av erfaringer ved utvalgt elektronikk til End Effector.

11.1 Konklusjon servomotorer

Vi ser at servomotorer fungerer veldig godt til funksjonaliteten vi trenger, noe av ulempen med at vi trenger et høyt kraftmoment, er at man gjerne ikke får mindre servomotorer som yter vårt ønske. Størrelsen til servomotoren utgjør mye på størrelse til design på EE, det kunne vært et ønske om å ha mindre størrelse på design. Det kan være en løsning med å endre bruksområdet til servomotorene og utnytte kraftmomentet på en annen mekanisk måte enn det som er blitt testet for å slanke designet.

11.2 Konklusjon kontrollere

Mikrokontrolleren Arduino ATmega 2560 fungerer godt til det bruket vi trengte den til, koding og brukervennlighet til dette systemet er veldig snilt og er lett å lære, det er mange «script» tilgjengelig på «forums» og på nettet som kunne gi oss en pekepinn hvordan vi skulle løse oppgavene vi trengte å løse. Det ble lagd signalkabler i AVG 24 slik at vi kunne sende signaler videre til servomotorene fra mikrokontrolleren.

11.3 Konklusjon strømkilde

Vi hadde et ønske om å kunne bruke forsyningen UR5 roboten kunne levere 12/24V, 600mA, men siden servomotorene trengte mye mer effekt for å kunne utføre oppgavene de skal utføre, måtte vi gå for å bruke en ekstern strømkilde, vi lagde strømkabler i AVG 16 for å kunne lede strøm fra en ekstern strømkilde som står under bordet der UR5 roboten står og gjennom et åpent hull i bordet, og stripset kablet fast med mulighet for at UR5 kan bevege seg uten at kablene skulle hindre bevegelser.

Kilder

[1] Arduino. (2016)

Arduino ATmega 2560.

Tilgjengelig fra:

<https://www.arduino.cc/en/Main/ArduinoBoardMega2560>

Sist besøkt: 18.05.2016

[2] Arduino (2016)

ATmega2560-Arduino Pin Mapping.

Tilgjengelig fra:

<https://www.arduino.cc/en/Hacking/PinMapping2560>

Sist besøkt: 18.05.2016

[3] Tti-test.com (2016)

CPX200 – Description and Specifications.

Tilgjengelig fra:

<http://www.tti-test.com/products-tti/text-pages/psu-cpx200.htm>

Sist besøkt: 18.05.2016.

[4] A.K. Svendsen; E.L. Roa; H.B. Sørsum; S. Rudin. Automated CROWS Testing Bachelor Thesis, Fakultet for teknologi og maritime fag, Høgskolen i Buskerud og Vestfold, Kongsberg, 2014.

Tilgjengelig fra:

<http://hdl.handle.net/11250/216910>

Sist besøkt: 18.05.2016

[5] Alberto Piganti, (Ukjent), *Arduino Mega Pinout*, [ONLINE].

Tilgjengelig fra:

<http://yourduino.com/docs/MegaPinOut.png>

Sist besøkt: 18.05.2016

[6] Universal Robots, (Ukjent), *UR5 User Manual*, [ONLINE].

Tilgjengelig fra:

http://www.universal-robots.com/media/8704/ur5_user_manual_gb.pdf

Sist besøkt: 18.05.2016

DESIGNDOKUMENT FOR MEKANIKK

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAKSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Designdokument for mekanikk
VERSJON / DATO	2.0 / 19 Mai 2016
FORFATTER	Monica Nadia Haugen
SIDEANTALL	18



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



Abstrakt

Det utvikles et gripeverktøy til montering på en robot arm, og vår oppgave er å integrere verktøyet og dets funksjoner inn i et eksisterende system. Hensikten er at robotarm og gripeverktøy skal kunne betjene knapper og paneler for kommandoer som er nødvendig for testing av systemets software.

i. Innholdsfortegnelse

i. Innholdsfortegnelse	3
ii. Oversikt Tabeller.....	4
iii. Oversikt Figurer	4
iv. Dokumenthistorie	5
v. Forkortelser	6
1 Innledning.....	7
1.1 Kartlegging av kriterier og kategoriseringer	7
1.2 Vårt valg av et konsept.....	8
1.3 Prosjektmodellen	8
2 Kartlegging	9
2.1 Hvilken type gripeverktøy passer?	9
2.2 Viktige fysiske egenskaper.....	9
Overflate på objekt	9
2.3 Nødvendige funksjoner	10
3 Valg av mekanisk design.....	11
3.1 Idéfasen	11
3.2 Inspirasjonskilder	12
3.3 Første prototype.....	13
3.4 Etter første iterasjon.....	14
4 Konseptvalg	15
4.1 Hensikten.....	15
4.2 Valg av antall fingre	15
4.3 Valg av løsning for kraft og bevegelse	16
5 Vurderinger	17
5.1 Verdi	17
5.2 utfordringer.....	17
5.3 Metodikk og prosjektmodell.....	17
Referanser	18

ii. Oversikt Tabeller

Tabell 1: Dokumenthistorie	5
Tabell 2: Forkortelser.....	6
Tabell 3: Kriterier og kategorisering	7
Tabell 4: Matrise for konseptvalg for antall fingre.....	15
Tabell 5: Pugh matrix for konseptvalg for kraft og bevegelse.....	16

iii. Oversikt Figurer

Figur 1: Illustrasjon på prosjektet og «Scrum» med stadige iterasjoner	8
Figur 2: Illustrasjon på robot med gripeverktøy.....	9
Figur 3 "Control Grip"	10
Figur 4: Skisser fra idè myldring	11
Figur 5: «The Barretthand» [2].....	12
Figur 6 «Robotiq» three finger adaptive gripper [3].....	12
Figur 7 «Schunk»[4]	12
Figur 8: Første prototype	13
Figur 9: Første prototype med tre fingre	14
Figur 10: Samarbeid.....	14

iv. Dokumenthistorie

Tabell 1: Dokumenthistorie

v 0.1 19.02.16	Monica: oppbygging med innhold og struktur. Forslag og stikkord for innhold.
v 0.2 22.02.16	Monica: Forarbeid tekst
v.0.3 25.02.16	Monica: Tillegg tekst
v 0.4 28.02.16	Monica: Bilde grensesnitt UR5 og EE denne går i til opprettet separat Testverifikasjonsdokument
v 0.5 09.03.1	Monica: Restrukturering av oppsett
v 0.6 11.03.16	Monica: Sammenslåing el/mek, tillegg tekst
v 07. 12.03.16	Monica: Tillegg tekst, Carl Martin: Elektronikkdokumentet, Tekniskdokument for elektrisk styring av EE (v0.2) implementeres.
v 0.8 13.03.16	Monica: Redigering, tillegg Carl Martin: tilleggstekst servoer.
V 1.0 15.03.16	Monica: Splittet til fire underdokumenter for EE
V 1.1 16.04.16	Monica: Oppretter revidert utgave
V 1.2 04.05.16	Monica: Bilder og tekst,
V 1.3 10.05.16	Monica: Mer tekst og begrunnelser for konseptvalg
V 1.4 12.	
v. 2.0 19.05.16	Monica ferdigstiller og legges i EE-dokumentet

v. Forkortelser

Tabell 2: Forkortelser

Forkortelser	Forklaring
KPS	Kongsberg Protech Systems
HSN	Høgskolen i Sørøst-Norge
ACT [2]	Automated CROWS Testing (foregående bachelorgruppe)
ACTEE	Automated CROWS Testing End Effector
EE	End Effector, gripeverktøy
CG	Control Grip, styringsjoystick for våpenstasjon
DCP	Display Control Panel
UR5 [1]	Universal Robots 5 robotarm

1 Innledning

Dette dokumentet gir en beskrivelse av utviklingen for den mekaniske delen av gripevektøyet, heretter omtalt som EE («End Effector»). Her beskrives grunnlaget for valg av design og konsept for funksjonalitet, som omhandler fasene fra idé til planlegging og konseptvalg, og videre til designet, basert på de valgte konsepter.

Det finnes mange ulike gripeverktøy, og teknologier rundt automatisering utvikler seg raskt. Gjennom grundig research fikk vi en del kunnskap om hvor det er lønnsomt å starte, hva som bør kartlegges og ligge til rette for å oppnå et bra resultat med våre forutsetninger.

For et slikt verktøy er vekt viktig – massesenteret bør ligge lavest mulig, dvs. i håndflata. Generelt bør det etterstrebes lavest mulig tyngde. Og de geometriske forhold, størrelse på hvert ledd i forhold til hverandre samt riktig størrelse på fingre i forhold til håndflaten.

1.1 Kartlegging av kriterier og kategoriseringer

Tabell 3: Kriterier og kategorisering

FOKUS	VURDERING OG HENSYN
Objekt som skal betjenes	Vekt og størrelse form bevegelighet overflate
Metode for aktivering	Mekanisk grep vakuum kopp magnet friksjon
Kraft og overføring	Pneumatikk elektronikk hydraulikk mekanikk
Gripekraft	vekt på objekt type grep friksjons koeffisient
Posisjonering	lengde på fingre instabilitet etter bevegelser for store toleranser
Vedlikehold	levetid på aktuatorer utmatting og slitasje tilgang på deler brukermanual
Materialer	styrke og stivhet holdbarhet produksjonskostnad friksjon på overflate
Andre hensyn	utbyttbare fingre standard på design grensesnitt følger av endring på design

1.2 Vårt valg av et konsept

Vi valgte en tre finger gripper fordi det er det færreste antall fingre man må ha for å kunne utføre bestemte handlinger, men har samtidig et bredt bruksområde på grunn av mange kombinasjoner på orientering av fingre. Det finnes mange varianter av et verktøy. Et enkelt to-finger gripeverktøy vil kunne utføre “pick-and-place”, men er ikke egnet til å utføre betjening av en «control grip».

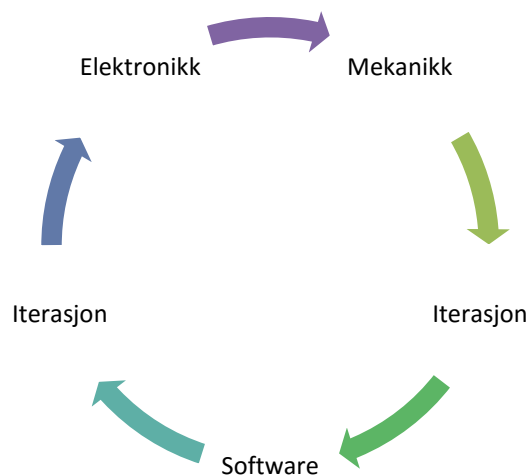
Et verktøy med tre fingre gir et langt bredere bruksområde. Med en «three-finger-gripper», kan man ha mulighet til å utføre en rekke ulike typer grep. Samtidig er denne løsningen mindre kompleks og mer robust enn en med fem fingre. Selv om enklere gripeverktøy med to fingre innehar frihetsgrader, er det ikke mulig å rotere fingrenes plassering i forhold til håndflaten, noe som er nødvendig ved samvirkende og / eller motstående orientering av fingrene.

Plasseringene av fingrene i forhold til en håndflate kan også spille en rolle i manipulering av objekter. Håndflaten bør ha en geometri som gjør det mulig for fingrene å bli rotert om en akse som er sammenfallende med bunnen og koblingen av fingre. Dette forklarer vårt valg et konsept med tre fingre med evne til å rotere rundt håndflaten.

Valget vårt er basert på grundige undersøkelser – vi ønsket å satse på en enkel løsning, men samtidig en løsning som kan videreutvikles.

1.3 Prosjektmodellen

Prosjektet er basert på «Scrum»-modellen, en agil prosjektmodell med intervaller med gjennomkjøring av alle faser, og produktet blir mer «utstyrt» for hver runde. Det ble i henhold til denne strategien produsert en tidlig prototype. Denne har vi brukt som utgangspunkt i den videre utviklingen av gripeverktøyet. Vi benytter SolidWorks som verktøy for designutviklingen.



Figur 1: Illustrasjon på prosjektet og «Scrum» med stadige iterasjoner

2 Kartlegging

2.1 Hvilken type gripeverktøy passer?

Å velge rett type verktøy er essensielt for at prosjektet skal lykkes. Viktige momenter ved valg av «end effector», er geometri på objektet som skal gripes og grad av kompleksitet på gripebevegelser.

Eksisterende «end effectors» kan deles i to brede kategorier. Den ene er karakterisert ved evnen til å utføre presise fingerbevegelser, og har lyktes i å oppnå en evne til å håndtere små objekter med presisjon, smidighet, og styrt gripekraft. Denne kategorien omhandler verktøy med design mest mulig lik en menneskehånd. For disse kan utfordringer være begrenset styrke og/eller skjør konstruksjon.

Den andre kategorien av gripeverktøy omfatter enkle gripeinnretninger som finnes i de fleste industrielle robotsystemer og er ofte styrt av pneumatikk eller hydraulikk. Disse enhetene er sterke og robuste, og består som regel av to «fingre» som utgjør en klype eller klo.

2.2 Viktige fysiske egenskaper

Overflate på objekt

Overflateegenskapene på objektet EE skal ta i, er, i tillegg til geometri, en viktig faktor å ta hensyn til. Høy friksjon mellom EE og objektet kan gi stor hjelp for å få fingrene til å feste seg. Dette er da forutsatt at det er små krefter fra EE i en slik situasjon.

Størrelse tilpasset CG

For å oppnå et vellykket resultat, må både størrelse på håndflate og avstand mellom fingre være planlagt. I tillegg er størrelsesforholdet mellom lengde på hver del av fingeren mellom ledd(ene) og tykkelse på fingrene i forhold til lengde en viktig faktor å være bevisst på.

Utforming av fingertupp

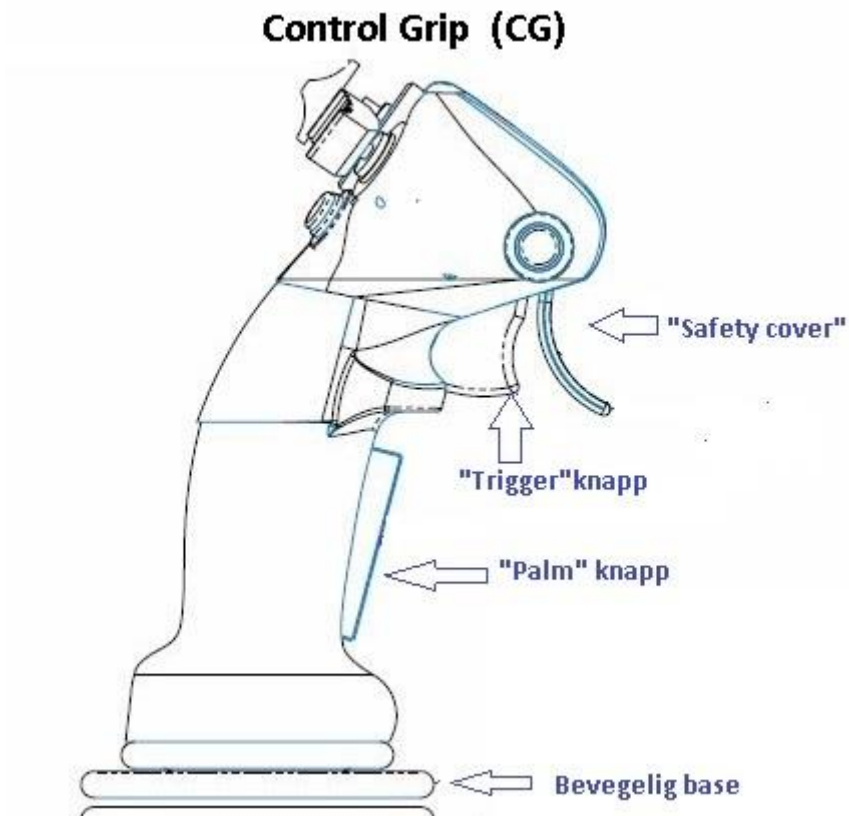
Fingertuppene er en viktig del av designet. Skal de kunne plukke ut og sette på plass, bør de ha utforming som gjør det mulig å klemme på små objekter. For vår oppgave vil dette være svært gjeldende da det er flere typer knapper som skal betjenes.



Figur 2: Illustrasjon på robot med gripeverktøy.

2.3 Nødvendige funksjoner

EEⁱ skal kunne gripe CG og betjene både «palm» og «trigger». Gripeverktøyet skal også kunne betjene DCP i form av vippebryter og knapper.



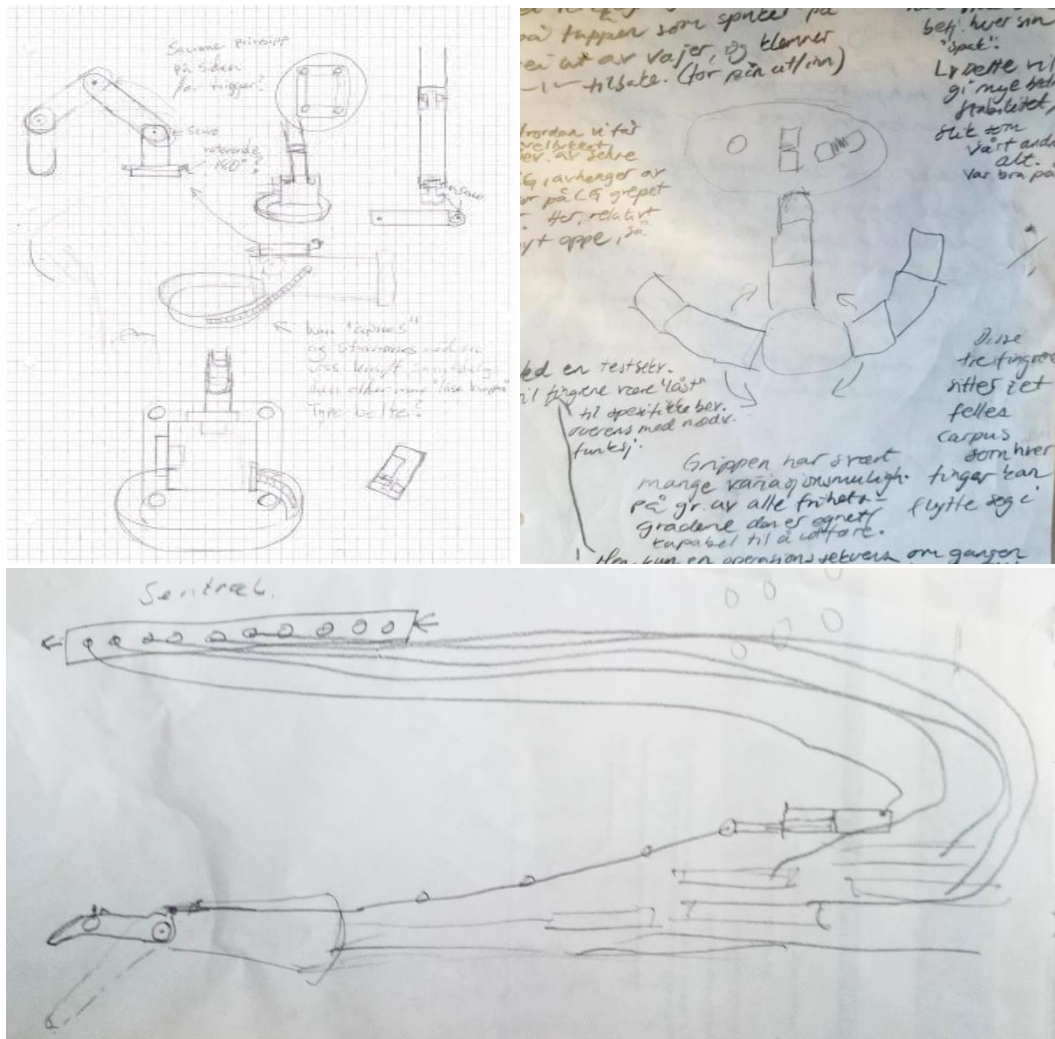
Figur 3 "Control Grip"

Sekundært er det ønsket at gripeverktøyet også skal kunne betjene de øvrige knappene på CG. Det er også et ønske om at gripeverktøyet skal kunne dra ut og putte tilbake pins fra break-outbox. Dette er tatt hensyn til under utviklingen av verktøyet, slik at det er mulighet til å tilføye verktøyet flere funksjoner. For at funksjoner på Control Grip skal aktiveres, må palm-knappen nederst på forsiden holdes inne. Dette er derfor et naturlig sted å begynne med funksjonalitet på verktøyet.

3 Valg av mekanisk design

3.1 Idéfasen

Det ble tidlig presentert flere idéer. Etter noen runder rundt dette, ble vi enige om at det beste er å lage noe ut ifra noe som er prøvd ut, og finnes fra før av. Først og fremst på grunnlag av en liten tidsramme, men også fordi det er slik ting stort sett fungerer i vår tid. Ting forbedres, og man får raskere resultater i en utviklingsprosess ved å benytte seg av andres erfaringer. Utviklingen går raskt, og det gjør også endringen for behov.



Figur 4: Skisser fra idémyldring

3.2 Inspirasjonskilder

En tidsramme på fire måneder er lite for et prosjekt som dette. For å sikre at vi kommer i mål med oppgaven, er det tatt utgangspunkt i eksisterende modeller. Her er de tre mest aktuelle.

The Barrett hand [1]

Dette er en av våre inspirasjonskilder til design og utviklingen av denne. Det er tre fingre som minner om en tommel, hvor to av dem roterer 180 til hver sin side rundt en håndflate som er festet i forbindelse med den tredje fingeren som er rigid.



Figur 5: «The Barretthand» [2]

Robotiq «Three finger adaptive gripper» [3]

Denne tre-grippen er designet i en mer firkantet håndflate, hvor alle tre fingre har lik funksjonalitet i form av bevegelighet. Denne har tre ledd, og en funksjonalitet i fingertupper med god evne til "pick and place".



Figur 6 «Robotiq» three finger adaptive gripper [3]

Schunk [4]

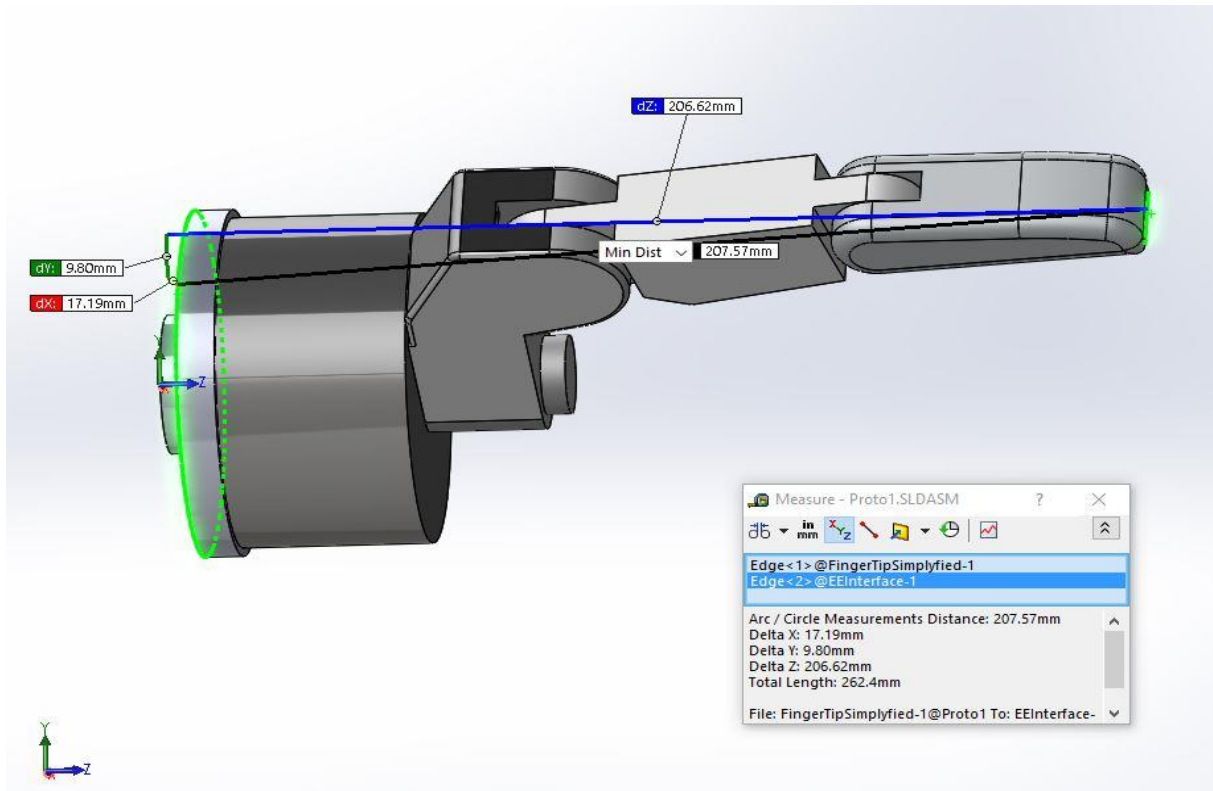
Schunk har også tre fingre plassert i en likevinklet trekant, og har pivot-ledd i innfestningen til «håndflaten». Denne er fullt aktivert, styrt med servoer i alle ledd.



Figur 7 «Schunk» [4]

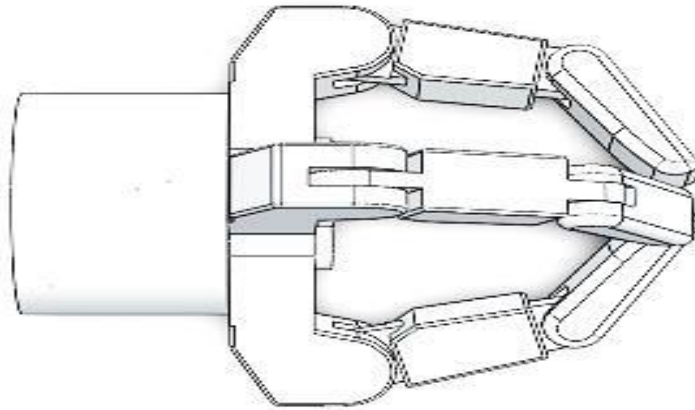
3.3 Første prototype

Vår første prototype ble laget i full skala med inspirasjon fra «The Barretthand». Dette var et utgangspunkt hvor vi viste til en modell med en av de tre fingrene. Prototypen har ingen teknisk funksjonalitet, men den er et visuelt hjelpemiddel under faglige dialoger for videre utvikling i ettertid. Modellen ble laget i Høgskolens 3D-printer som printer ut parter i et plast materiale. Dette gir et godt resultat, og vi bruker denne også til videre prototyp av vårt planlagte design.



Figur 8: Første prototype

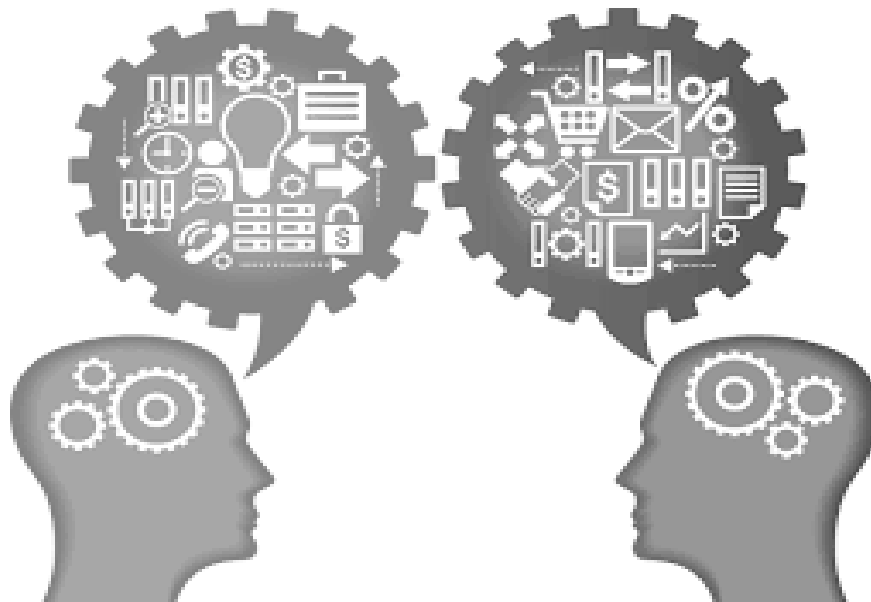
En prototype setter i gang en idéprosess. Vi gjør oss automatisk tanker rundt fremtidig funksjonalitet. Sammen med utvikling av kravspesifikasjon og testspesifikasjon, vil prosessen føre til en gradvis modning for evaluering av aktuelle løsninger. Den gir også en indikasjon på hva vi tenker, slik at oppdragsgiver får et vurderingsgrunnlag for validering tidligst mulig i prosessen.



Figur 9: Første prototype med tre fingre

3.4 Etter første iterasjon

Første prototype er presentert, og det rettes mer fokus mot designfasen. Utviklerteamet er enige om at utgangspunkt for design er «en god kandidat», på bakgrunn av at designet har røtter i noe vi kan stole på at virker. Valg av konsept beskrives i neste kapittel. Det fokuseres på kritiske områder som er definert til ytre fysiske grensesnitt, som innebærer nøyaktig treff på ulike gripefaser, og videre treff på knapper og brytere.



Figur 10: Samarbeid

4 Konseptvalg

4.1 Hensikten

Valg av konsept dreier seg om å vurdere ulike løsninger for en eller flere sammensatte funksjoner for produktet. Ved å rangere sentrale egenskaper, og deretter plassere disse opp mot konseptene i en matrise, er det lettere å få oversikt og vurdere helheten av hver løsning.

Denne fasen i prosjektet er med på å øke bevisstheten på hvilke komponenter med hva slags egenskaper som er nødvendig og best egnet til å møte produktets funksjonelle krav.

Når matrisen er fylt inn, skal man vekte de ulike konseptene ved å gi poeng ut ifra de ulike egenskapene. Til slutt summeres poengene, og «kandidaten» med høyest poengsum «vinner». Dette resulterer nok en underliggende god effekt:

Man må automatisk gjøre vurderinger opp mot krav for å kunne vekte ulike løsninger for oppfyllelse av dem.

4.2 Valg av antall fingre

Tabell 4: Matrise for konseptvalg for antall fingre

KRITERIER		FEM FINGRE	TO FINGRE	TRE FINGRE
Vekt	20%	0.25	0.75	0.50
Enkelhet	20%	0.30	1.20	1.00
Bruksbredde	25%	1.75	0.50	1.30
Kostnad	25%	0.20	0.80	0.70
SUM		2.50	3.25	3.60

Under evalueringen av konseptvalg for mekanisk design er det fokusert på avstander på knappene verktøyet skal betjene på CG, da denne presenterer det mest kompliserte grepet. Det er essensielt å ha støtte fra begge sider siden CG har en bevegelig sokkel som utgjør en av dens funksjoner.

Betjeningen av CG kan være mulig med kun to fingre. De fleste slike gripeverktøy er formet som en slags saks/klo som står parallelt overfor hverandre. For de fleste oppgaver er dette en logisk oppbygging, men for å gjennomføre grepet rundt CG er det nødvendig at fingrene treffer med en avstand for å treffe hver sin knapp (se figur 3, side 4).

Til de andre oppgavene, som er å trykke på DCP knappepanel og ta ut og sette på plass sikringer i «break-out-boksen», ville et verktøy med to fingre være godt egnet. Disse vurderingene er foretatt på bakgrunn av rent tekniske egenskaper for gripeevnen, sett bort ifra posisjonering.

KPS ønsket et verktøy med mulighet for andre bruksområder. Med disse tingene lagt til grunn, ble valget et verktøy med tre fingre. Utgangspunktet for designet er «The Barrethand» som er presentert tidligere i dokumentet.

4.3 Valg av løsning for kraft og bevegelse

Tabell 5: Pugh matrix for konseptvalg for kraft og bevegelse

KRITERIER		Tannhjulsoverføring fra motor/er nederst	Hydraulikk	Kun elektrisk motor
Kostnad	30%	0.55	0.60	0.20
Vedlikehold	20%	0.40	0.50	0.70
Mobilitet	15%	0.65	0.20	0.65
Utviklingsvennlig	25%	0.65	0.30	0.70
SUM		2.25	1.6	2.25

I samarbeid med elektro valgte vi å bruke servomotorer for kraft. Det resulterer i en litt større EE, men trenger ikke å ha noen betydning. Med servomotorer får man muligheten til å kontrollere hvert ledd uavhengig av hverandre. Dette har en klar fordel i vår oppgave, hvor ulike knapper skal betjenes hver for seg eller samtidig. Servomotorer er sterke og gir en hensiktsmessig stivhet i fingre og ledd, noe som også er et sentralt begrep for dette temaet.

Videre i den tekniske delen av dokumentet, belyses dette mer grundig.

5 Vurderinger

5.1 Verdi

Hovedfaktor for at problemstillingen i dette prosjektet skal få status som “løst”, er at UR5 og EE skal kunne implementeres i det eksisterende systemet, og kunne gjennomføre testsekvenser i form av programmerte bevegelser.

I tillegg er en av kriteriene for en verdifull oppgave, at produktet skal kunne oppdateres og videreutvikles i takt med tid og framtidige behov. Med ordet “verdifull” mener vi at det skal ligge framtidig verdi i produktet. Både i målbare og ikke-målbare hensyn.

En effektiviseringsprosess for KPS vil innebære høyere kvalitet pr investerte arbeidstime og mindre belastning på ansatte ressurspersoner. Dette er verdier som vokser over tid.

Ved at designet ikke er låst kun til de bruksområder som nå er definert, vil verdien på sikt være langt større for oppdragsgiver. I 2016 utvikles ulike teknologier raskere enn noen gang. For at vårt system skal være bærekraftig, må det legges til rette for rask integrering, ukompliserte grensesnitt og ha et oppsett og en brukermanual som tiltenkte brukere kan finne lett ut av.

5.2 utfordringer

På bakgrunn av grundig research, er det stadig mer innlysende at dette er en kompleks oppgave. Vi vil bruke vår utdanning og kunnskap, og må anvende den innen en kort tidsramme.

5.3 Metodikk og prosjektmodell

Prosjektgruppa bruker «Scrum» som prosjektmodell. Dette er en såkalt «agile», eller fleksibel modell, hvor det er lagt opp til inkrementell utvikling av et produkt. Med korte sprinter på to-tre uker, blir det svært lite hensiktsmessig for utviklingsprosessen for den mekaniske delen, å skulle produsere en fysisk prototype etter hver sprint. Vi har, på bakgrunn av egen erfaring og vurderinger underveis, og en studie[7], funnet ut at en modifisering i form av visuell prototype midtveis, vil være en god løsning for vår praktisering av «Scrum».

Referanser

[1]Guiseppe Carbon,"Grasping in robotics", Springer (2013)

[2]The Barrett Hand, "TheBarrethand.com."

[3]Robotiq`s Blog, [www."robotiq.com"](http://www.robotiq.com) (sist besøkt 11.04.16)

[4]"Schunk.com", www.schunk.com

[5] Boraas_Dollar_IJRR_ design paper,24.02.2016, Research Gate

[6] http://wiki.ros.org/barrett_hand#Overview (sist besøkt 19.02.16)

[7]Pórdís Reynisdóttir, « Scrum in Mechanical Product Development», Chalmers University of Technology

TEKNOLOGIER FOR MEKANIKK

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAGSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	TEKNOLOGIER FOR MEKANIKK
VERSJON / DATO	2.0 / 22 Mai 2016
FORFATTER	Monica Nadia Haugen



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



i. Abstrakt

Oppgaven går ut på å utvikle et gripeverktøy som kan erstatte en hånd til å betjene ulike grensesnitt mot en våpenstasjon. Det er spesielt tre parametre som gjelder. Geometri mellom de ulike deler av «hånden» og forholdet mellom to eller tre ledd.

ii. Innholdsfortegnelse

PERM 1

i. Abstrakt.....	2
ii. Innholdsfortegnelse	3
iii. Oversikt tabeller.....	3
iv. Oversikt figurer	4
vi. Forkortelser	4
iii. Dokumenthistorie.....	5
1 Innledning	6
1.1 Avgrensninger i oppgaven	7
1.2 Grepet	7
1.3 Angående sensorer	8
1.4 Friksjon.....	8
2 End Effektor	9
2.1 Grensesnitt mellom UR5 og EE	9
2.2 Kabinett	10
2.3 Plassering av fingrene	10
3 Materialteknisk	11
Referanser	13

iii. Oversikt tabeller

Tabell 1 : Forkortelser	4
Tabell 2 : Dokumenthistorie.....	5
Tabell 3 : Materialelegenskaper	11

iv. Oversikt figurer

Figur 1 : Blokkdiagram over fysiske egenskaper	6
Figur 2 : Overgang UR5/EE	9
Figur 3 : UR5 grensesnittet og kabinettet.	10

vi. Forkortelser

Tabell 1 : Forkortelser

Forkortelser	Forklaring
KPS	Kongsberg Protech Systems
HSN	Høgskolen i Sørøst-Norge
ACT	Automated CROWS Testing (foregående bachelorgruppe)
ACTEE	Automated CROWS Testing End Effector
EE	End Effector, gripeverktøy
CG	Control Grip, styringsjoystick for våpenstasjon
DCP	Display Control Panel
UR5	Universal Robots 5 robotarm
ABS	Akrylnitril-butandien-styren terpolymer, plastmateriale brukt i 3D-print.

iii. Dokumenthistorie

Tabell 2 : Dokumenthistorie

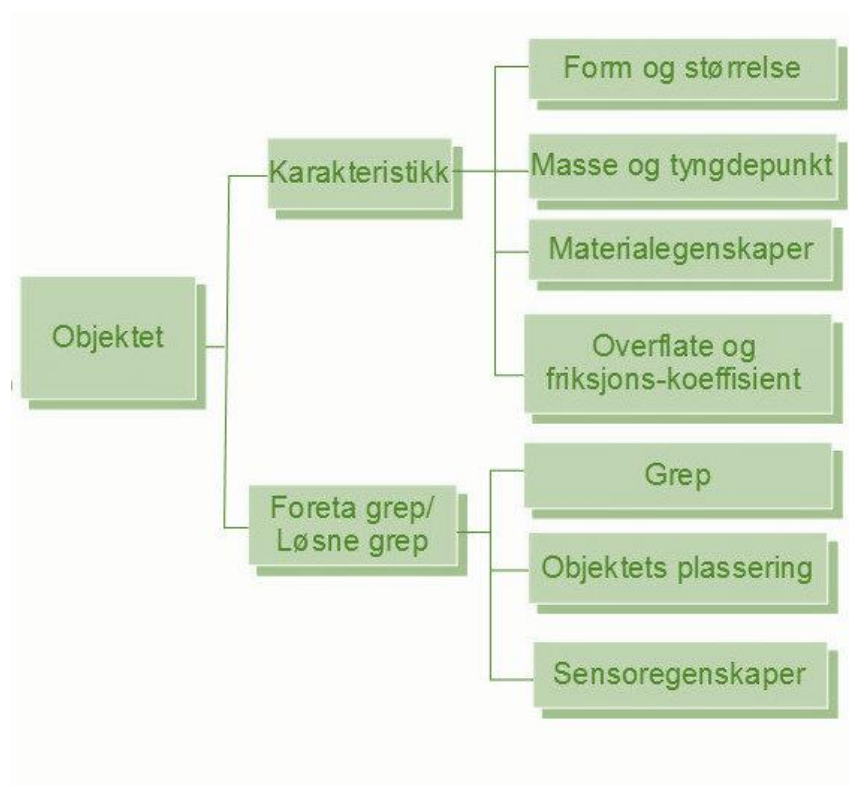
Versjon 0.1	20.02.16	Mal av Thomas.
Versjon 1.0	22.05.16	Sammendrag, innledning og konklusjon til hele rapportsamlingen for utviklingen av ACTEE

1 Innledning

Dette er et teknisk dokument for utviklingen av gripeverktøyet.

Utformingen av gripeverktøyet, som heretter kalles EE («End Effector»), er inspirert av en eksisterende modell, for å dra nytte av eksisterende teknologier vi vet fungerer.

Ut ifra en rekke forutsetninger, omtalt i design dokumentet, kan man danne seg et grunnlag for å utvikle et gripeverktøy som passer til tenkt formål. Dette dokumentet beskriver de fysiske og mekaniske parametre som påvirker egenskapene til et gripeverktøy. Blokkdiagrammet nedenfor gir en oversikt over hvilke fysiske egenskaper som påvirker et system med et gripeverktøy. Det forklares nærmere senere i dokumentet.



Figur 1 : Blokkdiagram over fysiske egenskaper

1.1 Avgrensninger i oppgaven

Kompleksiteten i å lage et verktøy som skal erstatte et menneskes hånd, varierer ut ifra om robot og objekt har en fast referanse til hverandre, eller om roboten må lokalisere objektet selv. Denne oppgaven er løst med hensikt i å utvikle et gripeverktøy som er fleksibelt i form av evne til å utføre flere typer grep, men avgrenset til å bli ført til rett posisjon av robotarmen. Dette innebærer at robot arm og de grensesnitt gripeverktøyet skal betjene, står fast i forhold til hverandre. En slik avgrensning gjør at vi kan forenkle kinematiske beregninger til kun gripeverktøyet uten å planlegge «banen» for hele systemet.

Etter å ha lest forskningsrapporter, journaler og relevant faglitteratur som omhandler utvikling av gripeverktøy og planlegging av bevegelser, er det en utfordring å vurdere hvilke områder som kan forenkles. Forenklinger og avgrensninger foretas på bakgrunn av en analyse av hva vi har som utgangspunkt og hva vårt produkt skal kunne gjøre sammen med systemet.

- Hvilken parameter er best i vårt tilfelle?
- Hvilke funksjoner fra robot armen kan vi dra nytte av?

1.2 Grepet

Et godt grep om objektet avhenger av at håndflaten har direkte kontakt før fingrene griper rundt objektet^[1] I følge faglitteratur, er det helt elementært med sensorer, men dette er helt individuelt ut ifra hvilke oppgaver som skal utføres og hva slags gjenstander som skal manipuleres¹ med vårt system har vi en svært brukervennlig robot arm

Det er også nødvendig at alle kreftene i systemet er i likevekt. Påført dreiemoment endres som en funksjon av flere parametre; den eksterne kraften som virker på EE posisjon, størrelse og type grep.

For å kunne utføre ulike grep må verktøyet ha nok stivhet, men samtidig inneha fleksibilitet nok til at innfestninger og overganger ikke skades. Massesenteret bør ligge i "håndflaten" for å oppnå stabilitet i grepet. **[1]**

¹ Manipulasjon; ordet brukes som et verb for at en type robot hånd tar i et objekt.

1.3 Angående sensorer

Det ble lenge vurdert om vi måtte ha sensorer på fingertuppene for å kontrollere grepet. Det viser seg at UR5 har en rekke innebyggede følsomheter. På grunnlag av dette, valgte vi å se bort ifra sensorer foreløpig. Viser til UR5 dokumentet for detaljer om dette.

1.4 Friksjon

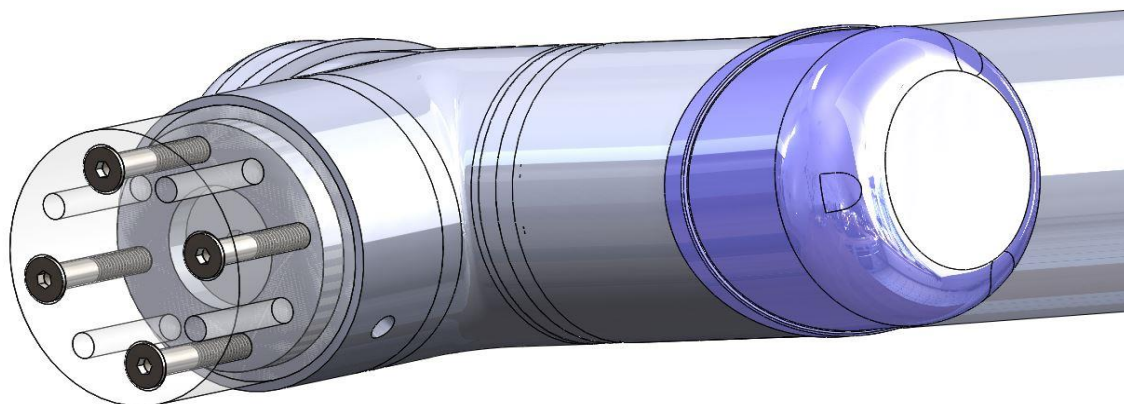
Både CG og EE er laget i plastmateriale. De har begge glatte overflater og lav friksjonskoeffisient, og det vil sannsynligvis bli vanskelig å trykke/holde inne knappene på CG. Ved å kle overflaten med et materiale som har høy friksjons-koeffisient (μ), vil vi oppnå et langt sikrere grep. Vi kan i stor grad unngå at fingrene sklir på overflaten til objektet som skal betjenes. Dette gar vi løst ved å kle typiske treffpunkt som fingertupper, innsiden av midtre ledd og håndflate med et gummi-aktig belegg.

2 End Effektor

Dette kapitlet beskriver gripeverktøyet i sin helhet. I et annet dokument blir oppbygging og tekniske spesifikasjoner for fingrene beskrevet. Disse ulike komponentene er også beskrevet i Testverifikasjon og risiko dokumentet, samt i brukermanualen.

2.1 Grensesnitt mellom UR5 og EE

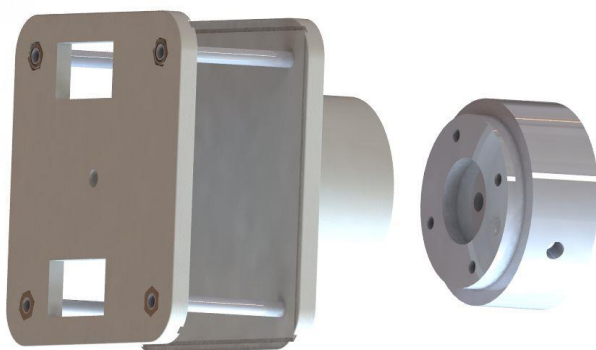
I enden av robot armen er det en sirkulær flate som utgjør et tverrsnitt av armen. På denne er det fire hull til feste av mekanisk grensesnitt. Vi skulle i utgangspunktet lage en hurtigkopling, men for å unngå flere avhengigheter, er det løst med gjengestang opp fra det mekaniske grensesnittet så kabinettet kan tres ned på skruene og festes med vingemuttre.



Figur 2 : Overgang UR5/EE

2.2 Kabinett

Videre ut fra grensesnittet, kommer kabinettet. Det er printet i ABS plast, noe vi som ble gjort etter et forsøk med plexi glass. I øvre del er det rom til innfelling av servomotorer. Disse plasseres slik for å unngå eventuell ubalanse. Det kan skje siden to av fingrene ofte er sammen på en side mot «tommelen».



Figur 3 : UR5 grensesnittet og kabinettet.

2.3 Plassering av fingrene

Plasseringen av de tre fingrene kan være en klar indikasjon på inspirasjonskilden, «The BarretHand». Det er ganske innlysende at vi mennesker er godt konstruert.

« ...dens finbevegelser kan forstås ved at håndens muskulatur og følenerver beslaglegger like stor del av hjernen som resten av kroppen gjør, med unntak av ansiktet « [5] sitat slutt.

3 Materialteknisk

Vårt produkt er 3D-printet i ABS. Dette anser vi som en midlertidig løsning, til EE har blitt plukket bort eventuelle barnesykdommer og er klar for et nytt «skall»

ABS er en termoplast med god balanse mellom fysiske og kjemiske egenskaper. Slagstyrke og aldringsbestandighet er særlig god. Bearbeidingsegenskapene er gode, og materialet lar seg sveise, lime og varm forme.

Tabell 3 : Materialeegenskaper

Noen egenskaper	Norm	Enhet	Verdi
Egenvekt	DIN 53479	g/m ³	1,06
Strekkefasthet	DIN 53455	N/mm ²	44
Bøyeplasthet	DIN 53452	N/mm ²	70
Slagseighet	DIN 53453	Ncm/cm ³	65-160
E-modul	DIN 53457	N/mm ²	2500
Forlengelse v/brudd	DIN 53455	%	35
Brukstemperatur	–	°C	÷40/+85
Temperaturutvidelse	ASTM D 896	mm/m/°C	0,08
Vannopptak	DIN 53495	%	0,3
Elektrisk motstand	ASTM D 257	ohm x cm	>1015
Dielektr. overflatemotstand	DIN 53482	ohm	1015
Brennbarhet	Brenner med sakte flamme		

ABS kategoriseres som en amorf² termoplast. Gjennom med ulike gummityper, kan vi endre egenskapene til materialet. Modifisering med butadien gir slagsikker styrenplast, og av disse har ABS de beste mekaniske egenskapene.[4] Det er ubegrensede muligheter for innfarging. Viser til dokument om fingrene.

² Amorf; Amorfe materialer er bygget opp med tilfeldig plassering av molekylerne, som også har varierende størrelse. Dette er en annerledes oppbygging enn hos metaller og andre typer plast, hvor det er en ordnet struktur.

Aktuelle materialer for videreutvikling kan være en aluminiumslegering eller titan. Titan innehar en indre fleksibilitet, og er samtidig slitesterk. Et fiberforsterket silikonmateriale/kompositt er også en god «kandidat» med mange muligheter.

3.1 3D print

En 3D printer kan mates med informasjon om en modell bygget i 3 dimensjoner, og produsere modellen i blandt annet plastmaterialer. Denne teknologien, kombinert med CAD verktøy, har gjort det mulig å utvikle produkter og fysiske prototyper langt mer effektivt og kostnadslønnsomt enn for bare 10 år siden. En klar fordel er at man kan teste produktet og eliminere eventuelle feil og gjøre endringer FØR det legges ressurser i produksjon.

Referanser

[1] Felip J, Morales A (2009) «Robust sensor-based grasp primitive for a three finger robot hand», IEEE/RSJ international conference on intelligent robots and systems(IROS 2009), pp 1811-1816

[2] Julia Borràs/Aron M. Dollar «Dimensional synthesis of three-fingered robot hands for maximal precision manipulation workspace». Computational Kinematics

[3] Astrup,(Ukjent). «Robust sensor-based grasp primitive for a three finger robot hand».
Tilgjengelig fra:

<http://astrup.no/Materialer-Produkter/Materialer/Plast-Standard/ABS>

Sist besøkt: 10.05.2016

[4] Anders Bjarbo, (1996), “ Konstruksjonsmaterialer”, NKI forlaget

Teknologidokument for grep

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAGSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Teknologidokument for grep
VERSJON / DATO	1.0 / 22. Mai 2016
FORFATTER	Aina Engen Nilsen
SIDEANTALL	35



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



i. Abstrakt

Dette dokumentet gir en oversikt over gripeverktøyets fingre. Innholdet er både design – og teknologiorientert, og vil ta lesere fra fingrenes spede begynnelse til første fungerende prototype.

ii. Innholdsfortegnelse

1 Gripeverktøyets fingre	3
1.1 Fra ide til konsept.....	4
1.2 Pugh Matrix – konseptutvelgelse design av fingre	5
1.3 Utforming design	7
2 Teknisk forarbeid	8
2.1 Beregninger	8
2.1.1 Kraftberegning fingertupp	8
2.1.2 kraftberegning rigid finger	9
2.1.3 Frihetsgrader	10
3 Teknisk data	11
4 FEM – analyser	17
4.1 FEM - analyse 1	18
4.1.1 Modellinformasjon.....	18
4.1.2 Analyseegenskaper	19
4.1.3 Materialelegenskaper	20
4.1.4 Laster og låsninger	21
4.1.5 Mesh detaljer	23
4.1.6 Resultater	24
4.2 FEM - analyse 2.....	27
4.2.1 Modellinformasjon.....	27
4.3.2 Studieegenskaper.....	27
4.3.3 Enheter.....	28
4.3.4 Materialelegenskaper	29
4.3.5 Laster og låsninger	29
4.3.6 Mesh informasjon	30
4.3.7 Mesh detaljer	30
4.3.8 Resultater	32
4.3.9 Konklusjon.....	33
5 Referanser.....	34

iii. Oversikt figurer

Figur 1: Skisser fra starten av prosjektet.....	4
Figur 2: Første forslag leddet finger og hånd/komponenthus	5
Figur 3: liten servomotor og plassering i finger	6
Figur 4: Første utkast modifisert sammenstilling hel finger	7
Figur 5: Rotasjonsakser i finger	8
Figur 6: koordinatsystem kraft fingertupp	8
Figur 7: Koordinatsystem kraft hel finger	9
Figur 8: Plassering servoer	9
Figur 9: Finger	10
Figur 10: Fingertupp	11
Figur 11: Fingertupp smal.....	12
Figur 12: Mellomledd	13
Figur 13: Baseledd	14
Figur 14: Sammenstilling smal.....	15
Figur 15: Sammenstilling normal	16
Figur 16: Utdrag fra materialegenskapene til ABS plast, Solid Works.	17
Figur 17: Finger i analyse	26

iv. Oversikt tabeller

Tabell 1: Dokumenthistorie	3
Tabell 2: Forkortelser.....	4
Tabell 3: Pugh matrise konseptvalg	5
Tabell 4: Analyseegenskaper FEM 1	19
Tabell 5: Laster og låsninger.....	21
Tabell 6: Mesh detaljer FEM 1	23
Tabell 7: Resultater FEM 1	24
Tabell 8: Studieegenskaper FEM 2.....	28
Tabell 9: Enheter FEM 2.....	28
Tabell 10: Laster og låsninger FEM 2	29
Tabell 11: Mesh informasjon.....	30
Tabell 12: Mesh detaljer	30
Tabell 13: Resultater FEM 2	32

v. Dokumenthistorie

Tabell 1: Dokumenthistorie

Versjon 1.0	22.05.2016	Omgjort fra kapittel til selvstendig dokument. Aina
-------------	------------	---

vi. Forkortelser

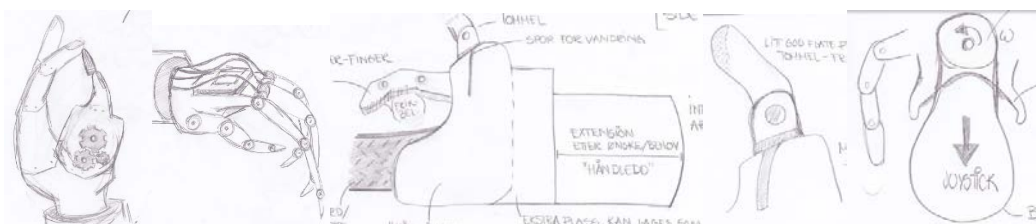
Tabell 2: Forkortelser

Forkortelser	Forklaring
KPS	Kongsberg Protech Systems
HSN	Høgskolen i Sørøst-Norge
ACTEE	Automated CROWS Testing End Effector
EE	End Effector, gripeverktøy
CG	Control Grip, styringsjoystick for våpenstasjon
DCP	Display Control Panel
CROWS	Common Remotely Operated Weapon Station
UR5 [1]	Universal Robots 5 robotarm

1 Gripeverktøyet sine fingre

1.1 Fra ide til konsept

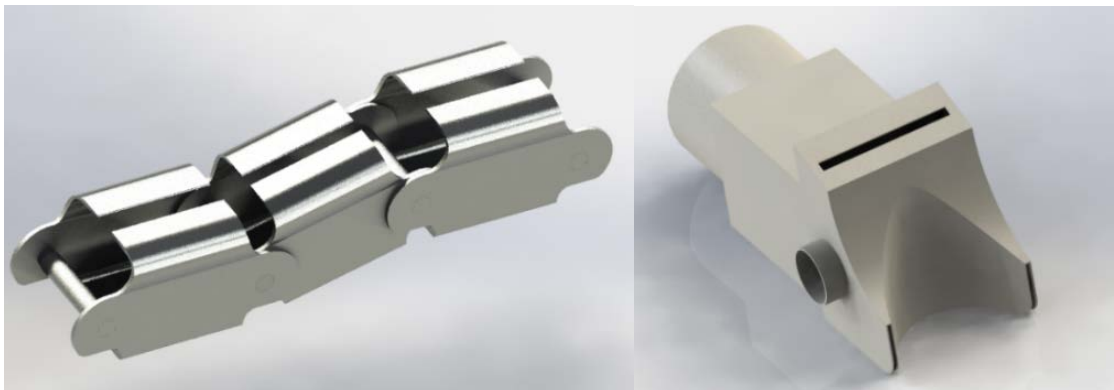
Tanke- og ideprosessen rundt designet på vårt gripeverktøy var i gang allerede i starten av januar. Løsninger diskuteres og skisseres, både på papir og i Solid Works.



Figur 1: Skisser fra starten av prosjektet.

For inspirasjon ser vi på flere ferdige modeller på markedet. Vi ser også at vår kompetanse sammen med tidsbegrensning for prosjektet vil stå i veien for å ta frem et helt eget produkt. Etter godkjenning fra oppdragsgiver, tar vi utgangspunkt i en allerede ferdig hånd; Barrett Hand [1]. Vi modifierer herfra en egen versjon av denne som vil møte oppdragsgivers krav.

For at gripeverktøyet skal gripe om å betjene CG på best mulig måte, ser vi for oss fingre med ledd. Hvordan leddene skal utformes, og hva slags styremekanisme som skal implementeres skal vise seg å bli en lang og omfattende utledingsprosess.



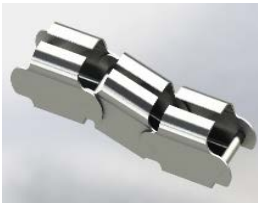
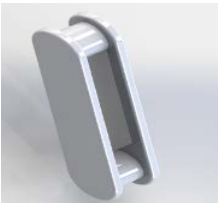
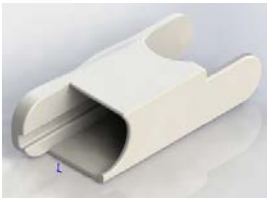
Figur 2: Første forslag leddet finger og hånd/komponenthus

1.2 Pugh Matrix – konseptutvelgelse design av fingre

Fingrenes utforming er av betydning da geometri avgjør grepets fleksibilitet. Styring og styringskomponenter sammen med behov for kraft i overføringer vektes tyngst i utvelgelsen. Dernext kommer grep etterfulgt av kompleksitet. Underliggende for utvelgelsen er oppdragsgivers krav til gripeverktøyet.

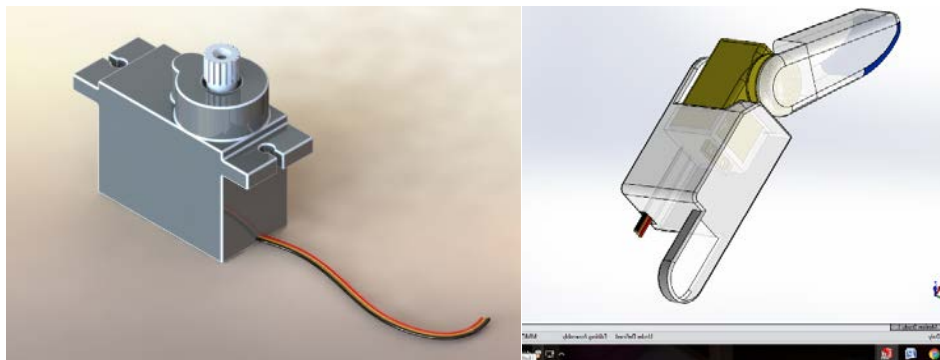
Konseptforslagene rangeres fra 1 – 3 hvor 3 er best. Rangeringen ganges med vekting for gjeldende kriterier, og produktet vises i uthevet kursiv til høyre i cellen. Konsept med høyest score (vertikal sum av tallene i kursiv) anses som best egnet, og blir vårt endelige konsept.

Tabell 3: Pugh matrise konseptvalg

Kriterier og vekting		A		B		C	
							
2	Grep/geometri	3	6	2	4	3	6
3	Styring	1	3	1	3	3	9
1	Kompleksitet	3	3	3	3	2	2
3	Kraftoverføring	2	6	2	6	3	9
Score		18		16		26	

Fingrenes nåværende design bærer preg av sin opprinnelse, men er samtidig modifisert en god del. Vi har lagt vekt på bevegelser - og kraft i disse - som er nødvendig for å utføre arbeidsoppgavene.

På bakgrunn av verktøyets arbeidsoppgaver og kraftbehov, falt valget på styringskomponenter i form av servomotorer [2]. Designet herfra bærer preg av servoenes krav til plass og plassering.



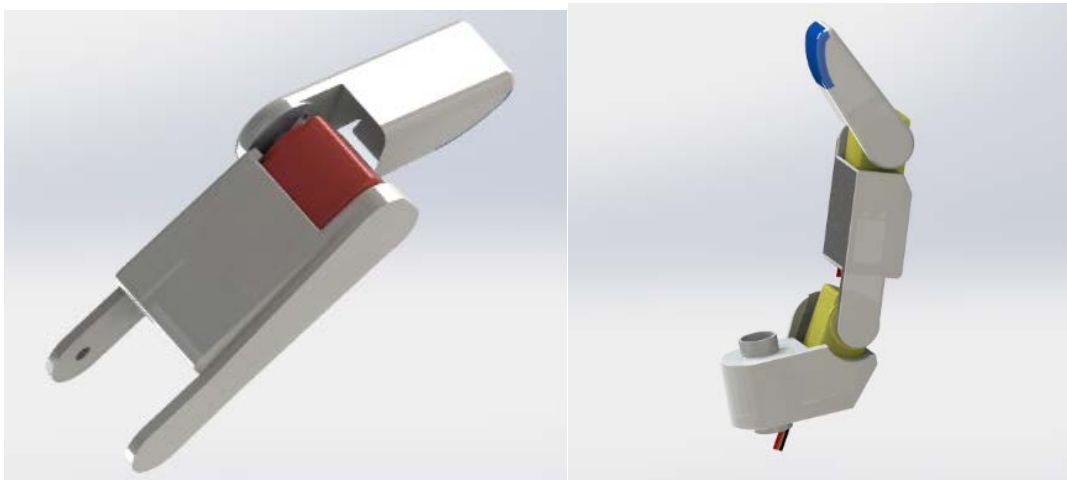
Figur 3: liten servomotor og plassering i finger

1.3 Utforming design

Fingrene er rektangulært utformet, hvilket er et spesifikt valg da denne utførelsen er spesielt godt egnet for et sikkert grep uansett geometri på last. Vi ser et behov for å kunne styre hvert ledd i fingrene separat, og i vår konseptutvelgelse veier dette tungt. Det er også behov for en viss kraft i grepet til verktøyet da grensesnitt som skal håndteres krever dette. I tillegg vil servomotor – løsningen gi enkelt vedlikehold og kort nedetid.

For at fingrene skal kunne kalles *fingernemme*, kreves det minst n uavhengige servomotorer. Noen ganger kreves det faktisk så mange som $n+1$ eller $2n$ av disse for å kunne drive n leddakser. Dessverre er servomotorer de komponentene som er mest kostbare, komplekse og mest plasskrevende. Så selv om den matematiske definisjonen av «fingernem» er aldri så elegant, vil det raskt lede frem til upraktisk design for faktiske produkter. Dersom vårt gripeverktøy skal følge den *matematiske definisjonen for fingernem*, vil det kreve mellom 10 og 16 motorer. Hånden ville bli altfor stor, kompleks og lite pålitelig for praktiske applikasjoner [3].

To av fingrene har tre rotasjonsakser hver, og er designet for å gjøre verktøyet fleksibelt. Den siste fingeren – tommelen – har to rotasjonsakser. Tommel har rigid baseledd, og har som oppgave støtte grep samt betjene øvrige knapper på baksiden av CG som et fremtidsmål.



Figur 4: Første utkast modifisert sammenstilling hel finger

2 Teknisk forarbeid

2.1 Beregninger

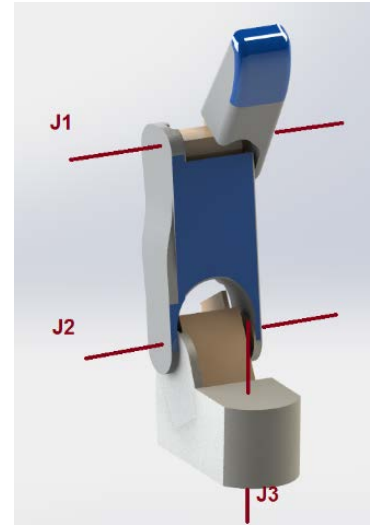
Det er et stort antall robothender på markedet, alle med forskjellig design. Noen design etterlikner menneskehånden, og innehar et stort antall frihetsgrader. Andre er designet for spesifikke oppgaver/applikasjoner, og vil dermed ikke være like antropomorfe. Siden robothånden blir mer og mer vanlig å se i industrielle applikasjoner og i menneskelige omgivelser, er det nærliggende å tro at deres design vil bli mer og mer oppgaveorientert.

Vår End Effector er nettopp dette – oppgaveorientert. Den består av tre fingre, rektangulært utformet for best mulig grep om et bredt utvalg nyttelast. Fingrene innehar to rotasjonsakser hver, i tillegg har to av fingrene en tredje rotasjonsakse ved innfestningen i 'håndflaten'.

For å kunne betjene DCP og CG må fingrene kunne virke på disse med stor nok kraft. Målinger med kraftmåler på de aktuelle knapper og brytere gir oss en oversikt over nødvendig kraft. Operasjonen med mest motstand trenger 10 N, vi bruker 15 N i våre beregninger.

2.1.1 Kraftberegning fingertupp

Vi ser på nødvendig kraft ytterst på fingertupp, både ved 45° og 90°. Vi er ute etter å finne nødvendig dreiemoment i øverste ledd, som her er plassert i origo i figur x.



Figur 5: Rotasjonsakser i finger

$$F = 15 \text{ N}$$

$$\nu = \alpha = 45^\circ$$

$$r = \text{lengde fingertupp} = 0,07 \text{ m}$$

$$L = r \sin \alpha = 0,07 \sin 45^\circ = 0,05 \text{ m}$$

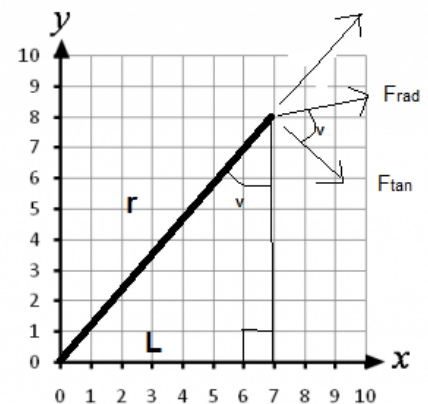
$$F_{\tan} = F \sin \alpha = 15 \sin 45^\circ = 10,6 \text{ N}$$

$$F_{\text{rad}} = F \cos \alpha = 15 \cos 45^\circ = 10,6 \text{ N}$$

$$\text{Worst case: } F_{\tan} = F \sin \alpha = 15 \sin 90^\circ = 15 \text{ N}$$

$$\mathcal{T}_{45^\circ} = FL = 10,6 \text{ N} * r \sin \alpha = 10,6 \text{ N} * 0,05 \text{ m} = \underline{0,53 \text{ Nm}}$$

$$\text{Worst case: } \mathcal{T}_{90^\circ} = FL = 15 \text{ N} * r \sin \alpha = 15 \text{ N} * 0,05 \text{ m} = \underline{0,75 \text{ Nm}}$$



Figur 6: koordinatsystem kraft fingertupp

2.1.2 kraftberegning rigid finger

Vi ser også på nødvendig kraft ytterst på fingertupp fra nederste ledd. Hel finger sees på som rigid. Vi er ute etter nødvendig dreiemoment i nederste ledd, som her er plassert i origo i figur under.

$$F = 15 \text{ N}$$

$$\nu = \alpha = 45^\circ$$

$$r = \text{lengde fingertupp} = 0.15 \text{ m}$$

$$L = r \sin \alpha = 0.15 \sin 45^\circ = 0.11 \text{ m}$$

$$F_{\tan} = F \sin \alpha = 15 \sin 45^\circ = 10.6 \text{ N}$$

$$F_{\text{rad}} = F \cos \alpha = 15 \cos 45^\circ = 10.6 \text{ N}$$

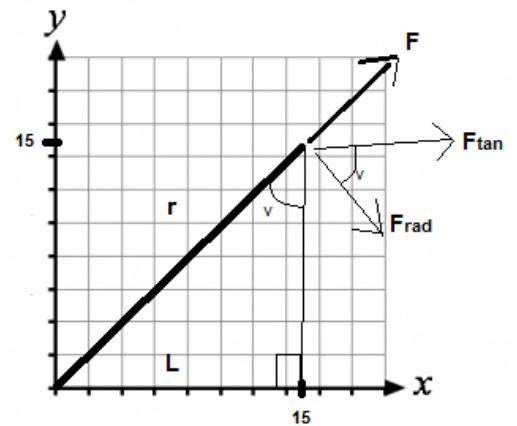
$$\text{Worst case: } F_{\tan} = F \sin \alpha = 15 \sin 90^\circ = 15 \text{ N}$$

$$\mathcal{T}_{45^\circ} = FL = 10.6 \text{ N} * r \sin \alpha = 10.6 \text{ N} * 0.11 \text{ m} = \underline{1.17 \text{ Nm}}$$

Worst case:

$$\mathcal{T}_{90^\circ} = FL = 15 \text{ N} * r \sin \alpha = 15 \text{ N} * 0.11 \text{ m} = \underline{1.65 \text{ Nm}}$$

8 servomotorer er valgt for styring, hvorav 6 av disse er implementert i selve fingrene. Dette fordi vi er ute etter separat bevegelse i hvert ledd, slik at grepet lettere kan tilpasses enkle og ulike kommandoer. I tillegg er det ønskelig å kunne styre kraft og fart separat. Servoene er plassert i hvert ledd, altså to i hver finger.



Figur 7: Koordinatsystem kraft hel finger



Figur 8: Plassering servoer

2.1.3 Frihetsgrader

Betrakter vi et system av n stive legemer som beveger seg i rommet, har dette $6n$ frihetsgrader målt i forhold til en fast ramme [4]. For å finne antall frihetsgrader i fingrene, må vi telle med basen de er montert på som et legeme. Dette gjøres for at mobiliteten skal være uavhengig av hva vi velger som den faste rammen. Vårt system må sees på som en kinematisk 'kjede' når det kommer til beregning av frihetsgrader, da de består av en sammenstilling av flere rigide legemer knyttet sammen i form av ledd.

$$M = 6n - \sum_{i=1}^j (6 - f_i) = 6(N - 1 - j) + \sum_{i=1}^j f_i$$

M = Mobilitet, bevegelse = frihetsgrader

N = Antall rigide legemer inkludert base =

n = Antall rigide legemer

$f_i = i = 1$ (frihetsgrad for servo) $\rightarrow c =$ begrensninger i ledd $= 6 - f_i = \underline{5}$ for hvert ledd.

J = antall ledd (her; servomotorer)

Ledd med styring i form av servoer fratar fingrene frihetsgrader, de gir oss begrensninger. Våre servoer innehar en egen frihetsgrad $f_i = 1$, som igjen gir hvert ledd en begrensning lik

$$C = 6 - f_i = 5$$

En finger består av 3 rigide legemer, 4 inkludert 'håndflaten' eller basen den monteres på. Antall ledd er 2 hvilke hver av de gir fingeren en tallfestet begrensning på 5. Vi får at:

$$M = 6(N - 1 - j) + \sum_{i=1}^j f_i = 6(4 - 1 - 2) + 2 = 8$$

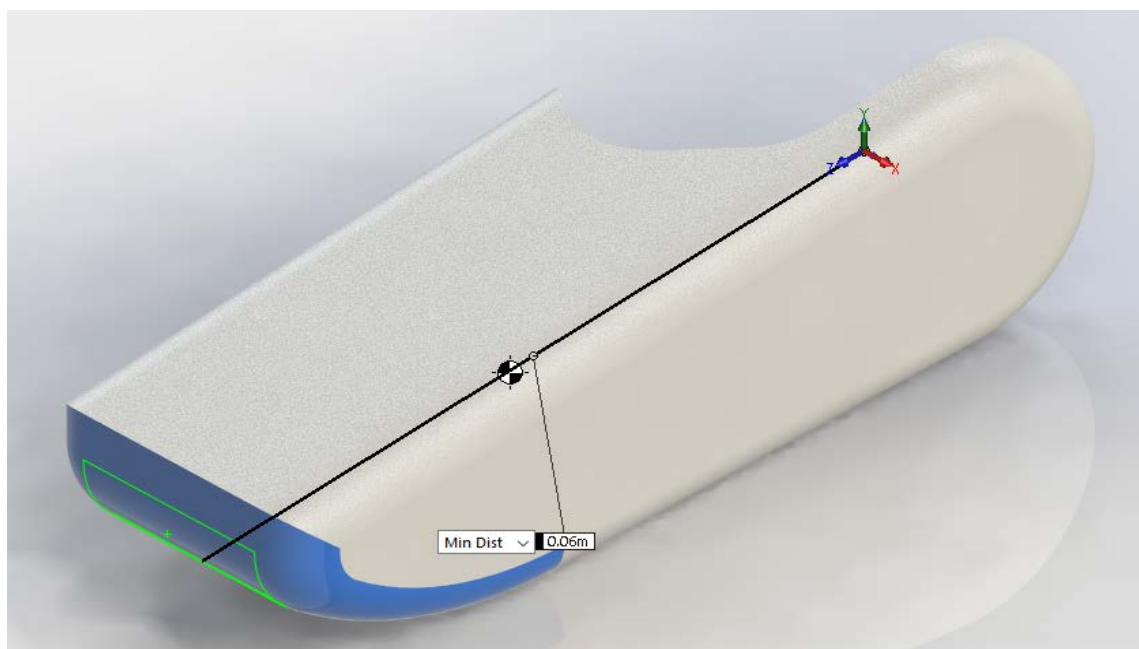
Fingren med bevegelig base har 8 frihetsgrader, hvilket gir vårt gripeverktøy 8 frihetsgrader. Dette er til sammenlikning det samme som vår Barret Hand, forskjellen er at sistnevnte kun har 4 motorer implementert mot våre 8.



Figur 9: Finger

3 Teknisk data

Masseegenskaper fingertupp	
Tetthet ABS	1.02 g/cm ³
Masse	23 g
Volum	22.6 cm ³
Overflateareal	5.74e+003 mm ²
Massesenter (mm)	X = 1.4 Y = -1.2 Z = 34.5

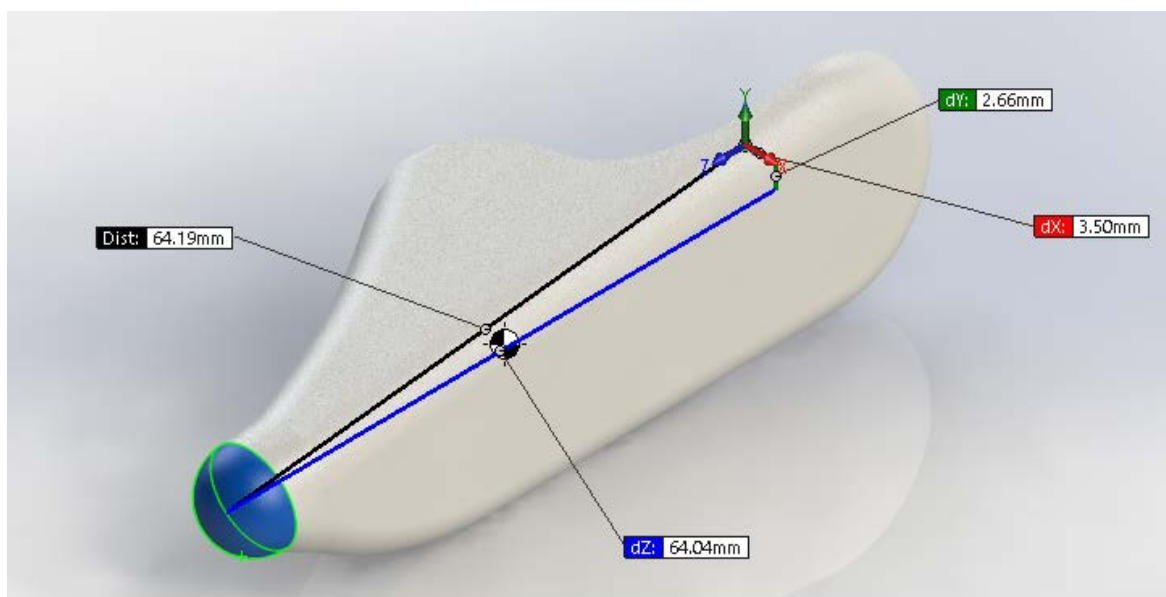
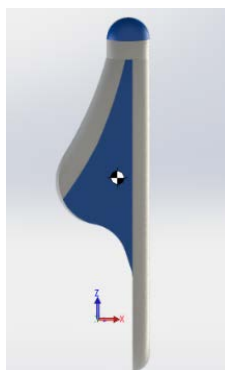


Figur 10: Fingertupp

Fingertuppen på to av fingrene er rektangulært utformet primært med tanke på oppgaver som å holde inne palm – knapp, trykke inn trigger – knapp, og betjene vippebrytere på et DCP – panel. Det er en fordel med bredere annleggsflate for jevnere kraftfordeling.

De blå feltene viser hvor vi ser for oss plassering av eventuelle sensorer ved videre utvikling.

Masseegenskaper fingertupp smal	
Tetthet ABS	1.02 g/cm ³
Masse	16.1 g
Volum	15.7 cm ³
Overflateareal	4.72e+003 mm ²
Massesenter (mm)	X = 4.5 Y = -1.2 Z = 32.5

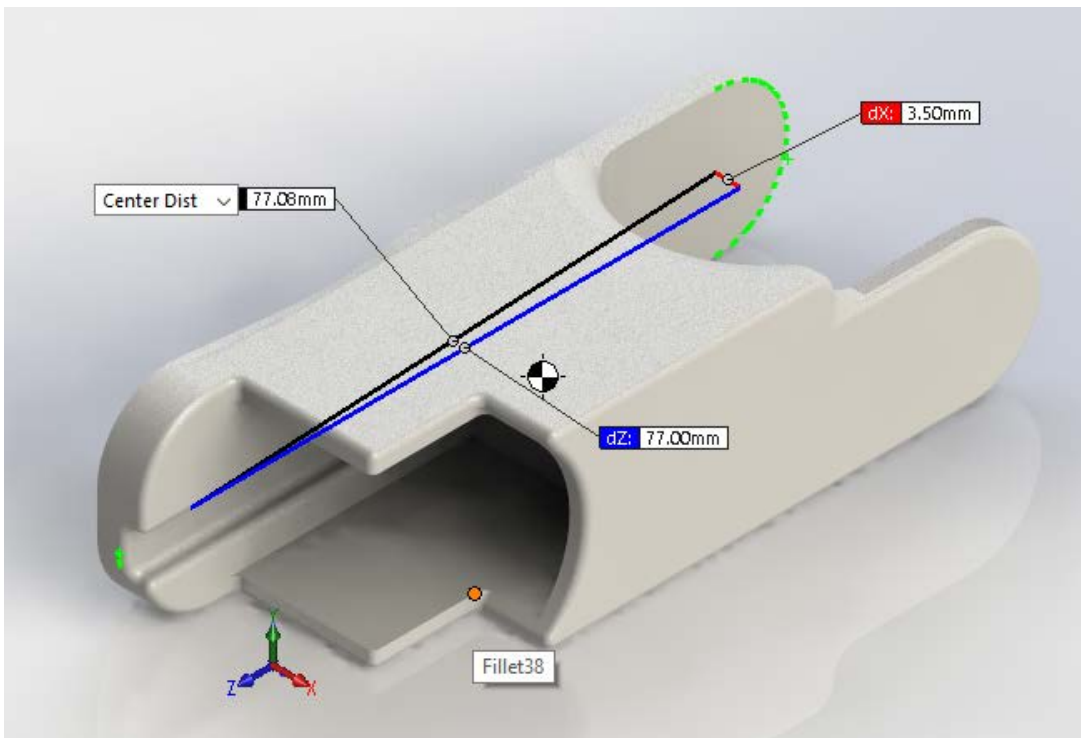
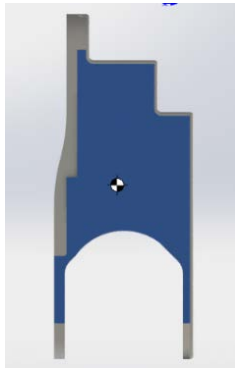


Figur 11: Fingertupp smal

Ved første test ser vi at den brede fingertuppen med fordel kan ha en smalere utforming for mest mulig presis trykking av knapper. Løsningen ble å designe en egen utforming til en av fingertuppene nettopp til dette formål.

Masseegenskaper mellomledd

Tetthet ABS	1.02 g/cm ³
Masse	33.2 g
Volum	32.5 cm ³
Overflateareal	1.39e+004 mm ²
Massesenter (mm)	X = -13.4 Y = 2.4 Z = -51.1



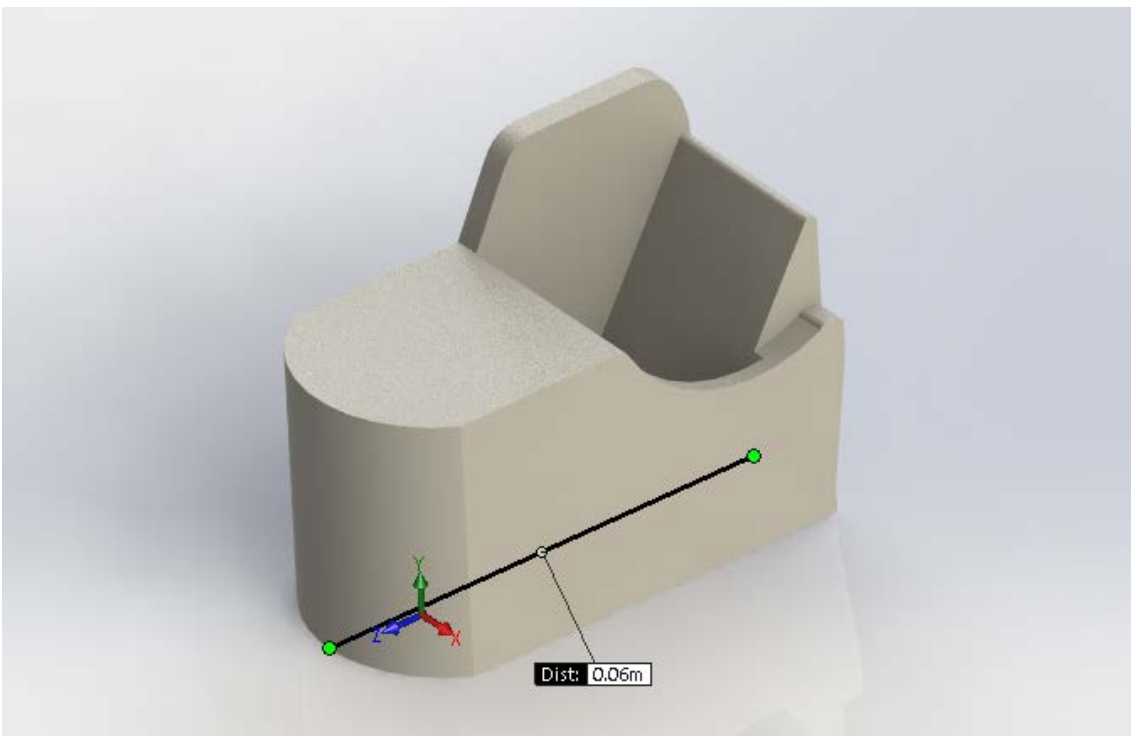
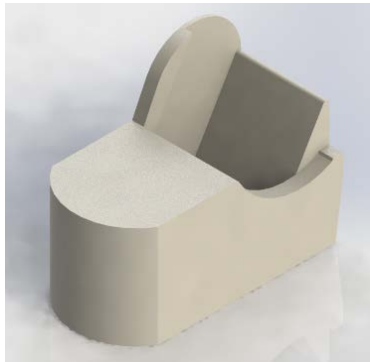
Figur 12: Mellomledd

Mellomledd er også utformet rektangulært med tanke på best mulig grep. Det er laget plass for servomotor med god klaring til drivhjul, og med støttevegger foran og bak. Nede i motorrommet er det laget et spor for feste av servo, som skrues fast fra utsiden. Motorrommet er dypt nok til at servoen sitter godt, men samtidig har en del av sitt overflateareal direkte mot luft for kjøling.

Det er planlagt plass for ledninger i et eget gjennomgående spor slisset inn i siden av leddet. På undersiden er det laget god klaring til servo montert i baseledd. Det er også laget et støtteben på motsatt side av servos drivhjul som festes for å forhindre uønskede torsjonsspenninger under arbeid, samt for stødigst mulig grep.

Masseegenskaper baseledd

Tetthet ABS	1.02 g/cm ³
Masse	61.3 g
Volum	60.1 cm ³
Overflateareal	1.41e+004 mm ²
Massesenter (mm)	X = -7.3 Y = 13.1 Z = -22.6

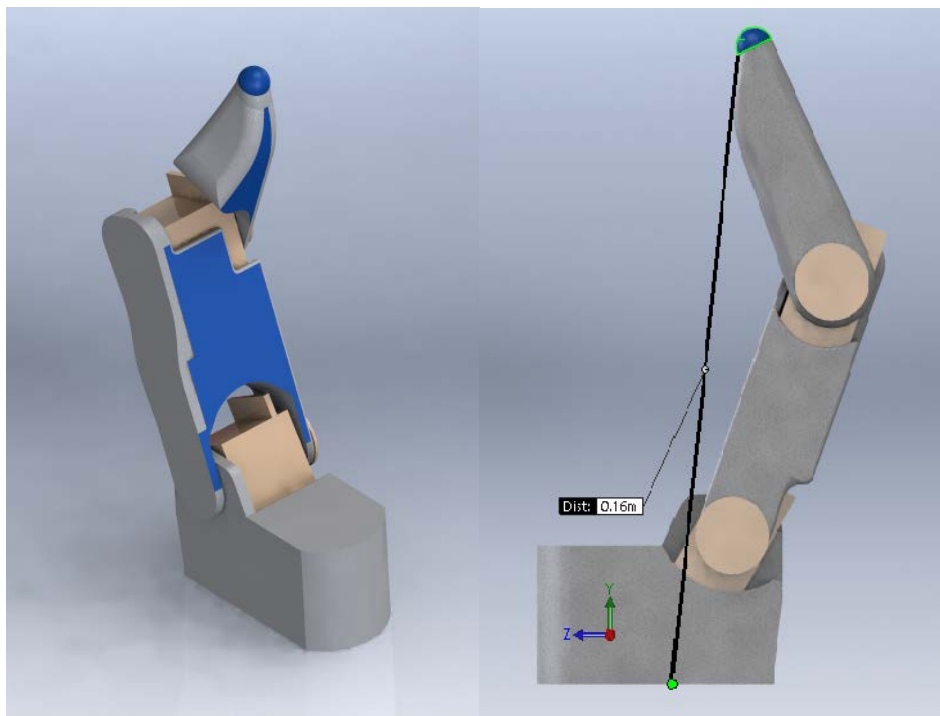
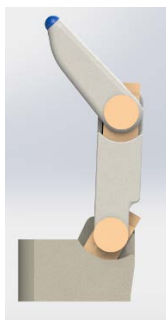


Figur 13: Baseledd

Baseleddet er utformet slik at servomotoren får en liten vinkel eller tilt bakover, for å danne et typisk grep. Baseleddet er robust utformet, og servoen er plassert godt ned i motorrom. Også her har vi tenkt på avkjøling, og har gode klaringer rundt drivhjul på servo. Det var i utgangspunktet tegnet inn et bredt spor for ledninger fra egen og overstående servo, men dette skapte noe hodebry ved printing. Sporet ble derfor tettet igjen, men burde planlegges inn igjen ved videre utvikling.

Masseegenskaper sammenstilling 2

Tetthet ABS	1.02 g/cm^3
Masse	210 g
Volum	$1.53\text{e}+005 \text{ mm}^3$
Overflateareal	$4.53\text{e}+004 \text{ mm}^2$
Massesenter (mm)	X = 6.3 Y = 44.0 Z = -35.2

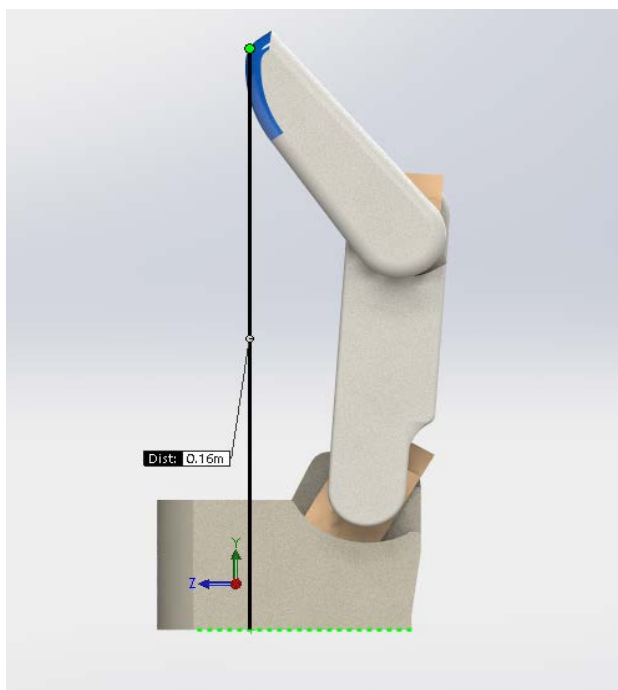
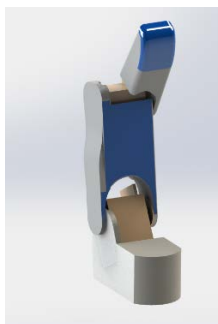


Figur 14: Sammenstilling smal

Sammenstilling av tommelfinger innehar de samme egenskaper som sammenstilling finger, men har en annen konfigurasjon av fingertuppen. Denne tuppen gjør det enklere o betjene 'trykk – knapper' på DCP.

Masseegenskaper sammenstilling 1

Tetthet ABS	1.02 g/cm^3
Masse	220 g
Volum	$1.62\text{e}+005\text{mm}^3$
Overflateareal	$4.69\text{e}+004\text{mm}^2$
Massesenter (mm)	X = 8.7 Y = 48.5 Z = -27.4



Figur 15: Sammenstilling normal

Sammenstillingen av enkeltledd og to servoer gir en finger med grep på ca 16 cm. Dette er ikke tilfeldig, da det samsvarer ganske godt med en menneskelig finger.

4 FEM – analyser

Beregning av Von Mises spenninger [5] i et materiale er ansett for å være en meget god metode for å sjekke at design og materiale vil fungere som tiltenkt. Beregninger av disse kreftene i en FEM - analyse vil gi oss en pekepinn på hvorvidt designet vil holde eller mislykkes dersom den maksimale verdien av von Mises spenninger induisert er mer enn styrken til materialet. Når vi skal beregne hva et materiale tåler av belastning, er det grovt sett to typer spenninger vi konsentrerer oss om; spenninger før varig formendring (elastisk sone), og spenninger før brudd (plastisk sone).

Slike analyser fungerer bra i de fleste tilfeller, spesielt når materialet er duktilt (formbart i naturen). Vi jobber derimot med ABS plast, og dette materialet er ikke duktilt – det har ingen flytegrense. Vi har allikevel valgt å kjøre en analyse i FEM for å teoretisk teste vårt design. Vi fokuserer da på spenningene før et brudd, og kort fortalt skal vi holde maksverdien av Von Mises -spenningene som induseres i materialet under materialets bruddstyrke.

Material properties
 Materials in the default library can not be edited. You must first copy the material to a custom library to edit it.

Model Type:

Units:

Category:

Name:

Default failure criterion:

Description:

Source:

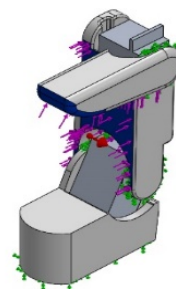
Sustainability:

Property	Value	Units
Elastic Modulus	2000	N/mm ²
Poisson's Ratio	0.394	N/A
Shear Modulus	318.9	N/mm ²
Mass Density	1020	kg/m ³
Tensile Strength	30	N/mm²
Compressive Strength		N/mm ²
Yield Strength		N/mm ²
Thermal Expansion Coefficient		/K
Thermal Conductivity	0.2256	W/(m·K)
Specific Heat	1386	J/(kg·K)
Material Damping Ratio		N/A

Figur 16: Utdrag fra materialegenskapene til ABS plast, Solid Works.

4.1 FEM - analyse 1

4.1.1 Modellinformasjon

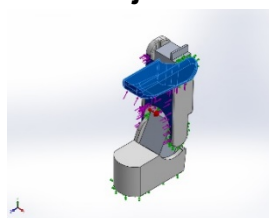


Modellnavn: Assembly bevegelig finger sprint 4 1

Konfigurasjon: Default

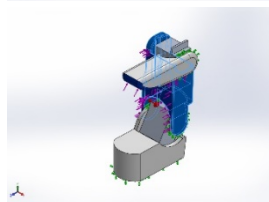
Solid Bodies

Informasjon



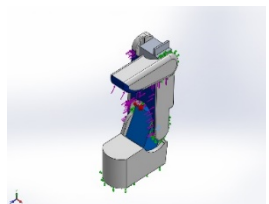
Solid
Body

Mass:0.0230384 kg
Volume:2.25867e-005 m³
Density:1020 kg/m³
Weight:0.225777 N



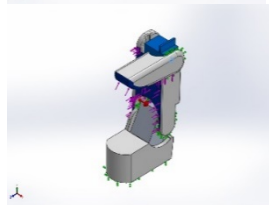
Solid
Body

Mass:0.0333176 kg
Volume:3.26643e-005 m³
Density:1020 kg/m³
Weight:0.326513 N



Solid
Body

Mass:0.0624323 kg
Volume:2.22179e-005 m³
Density:2810 kg/m³
Weight:0.611837 N



Solid
Body

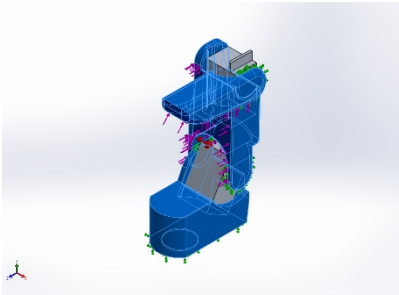
Mass:0.0624323 kg
Volume:2.22179e-005 m³
Density:2810 kg/m³
Weight:0.611837 N

4.1.2 Analyseegenskaper

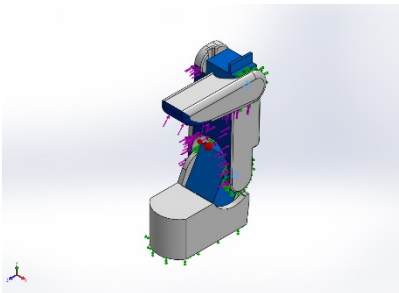
Tabell 4: Analyseegenskaper FEM 1

Study name	Static 1 bevegelig finger 4 1
Analysis type	Static
Mesh type	Mixed Mesh
Thermal Effect:	On
Thermal option	Include temperature loads
Zero strain temperature	298 Kelvin
Include fluid pressure effects from SOLIDWORKS Flow Simulation	Off
Solver type	FFEPlus
Inplane Effect:	Off
Soft Spring:	Off
Inertial Relief:	Off
Incompatible bonding options	Automatic
Large displacement	Off
Compute free body forces	On
Friction	Off
Use Adaptive Method:	Off
Result folder	SOLIDWORKS document (C:\Users\Aina\Documents\Bachelor Ing\Ing 3. år\HOVEDOPPGAVE\SolidWorks arbeider\Finger sprint 4)
Unit System	SI (MKS)
Length/Displacement	mm
Temperature	Kelvin
Angular velocity	Rad/sec
Pressure/Stress	N/m^2

4.1.3 Materialegenskaper



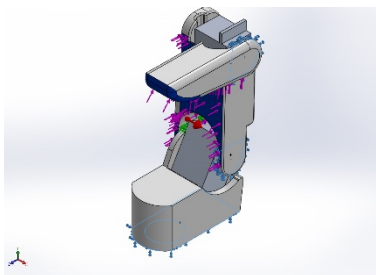
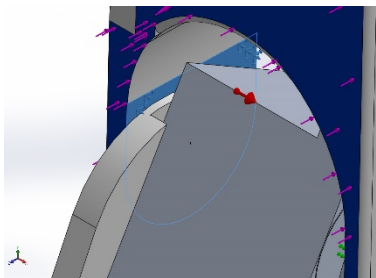
Name: ABS
Model type: Linear Elastic
Isotropic
Default failure criterion: Unknown
Tensile strength: $3e+007 \text{ N/m}^2$
Elastic modulus: $2e+009 \text{ N/m}^2$
Poisson's ratio: 0.394
Mass density: 1020 kg/m^3
Shear modulus: $3.189e+008 \text{ N/m}^2$

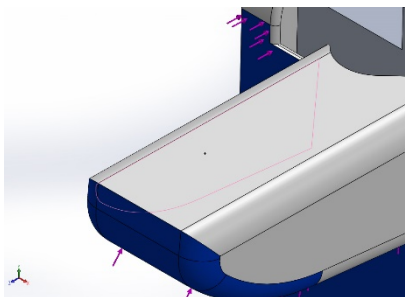
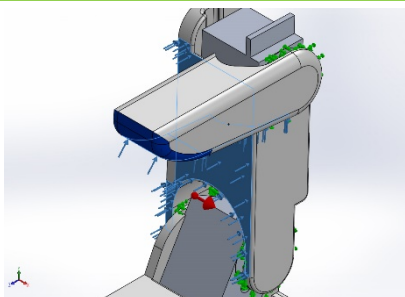


Name: 7075-T6 (SN)
Model type: Linear Elastic
Isotropic
Default failure criterion: Unknown
Yield strength: $5.05e+008 \text{ N/m}^2$
Tensile strength: $5.7e+008 \text{ N/m}^2$
Elastic modulus: $7.2e+010 \text{ N/m}^2$
Poisson's ratio: 0.33
Mass density: 2810 kg/m^3
Shear modulus: $2.69e+010 \text{ N/m}^2$
Thermal expansion coefficient: $2.4e-005 \text{ /Kelvin}$

Tabell 5: Laster og låsninger

4.1.4 Laster og låsninger

Fixed-1		Entities: 7 face(s) Type: Fixed Geometry		
Resultant Forces				
Components	X	Y	Z	Resultant
Reaction force(N)	-1.42068	-27.8141	15.6653	31.9538
Reaction Moment(N.m)	0	0	0	1e-033
Fixed-2		Entities: 1 face(s) Type: Fixed Geometry		
Resultant Forces				
Components	X	Y	Z	Resultant
Reaction force(N)	-0.357064	-0.0502363	7.13886	7.14796
Reaction Moment(N.m)	0	0	0	1e-033

Gravity-1		Reference: Face< 1 > Values: 0 0 -9.81 Units: SI
Force-1		Entities: 3 face(s) Type: Apply normal force Value: 15 N

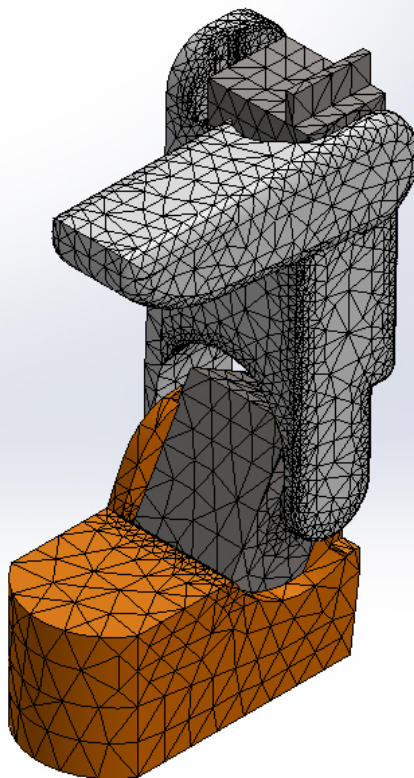
I denne analysen er det satt på krefter i form av gravitasjon og statisk, jevnt fordelt kraft der fingeren er ment å treffe de knapper den skal betjene. Kraftens størrelse er satt til litt over nødvendig kraft.

Tabell 6: Mesh detaljer FEM 1

4.1.5 Mesh detaljer

Total Nodes	43642
Total Elements	25031
Time to complete mesh(hh:mm:ss):	00:00:03
Computer name:	AINA-PC

Model name: Assembly bevegelig finger sprint 4 1
Study name: Static 1 bevegelig finger 4 1(-Default-)
Mesh type: Mixed Mesh

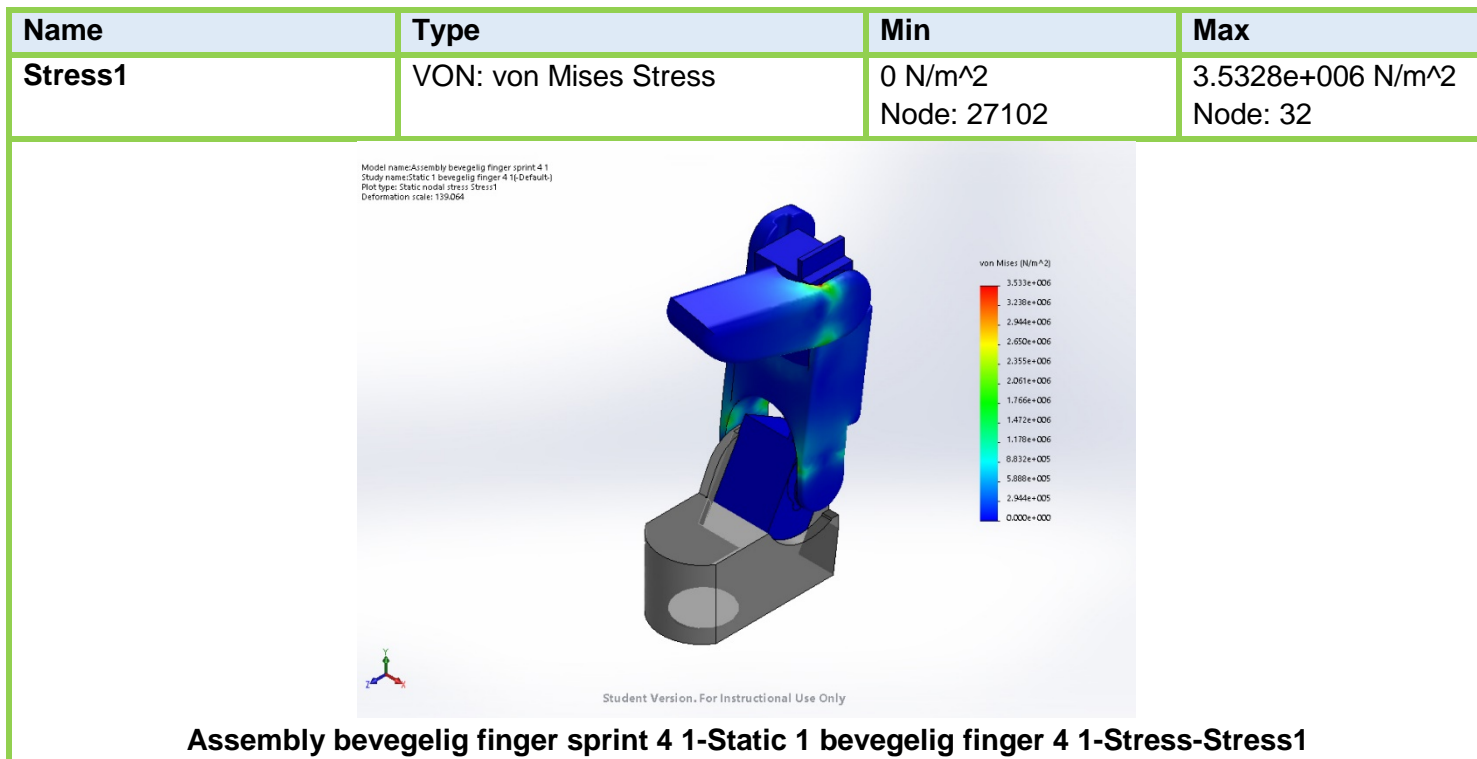


Student Version. For Instructional Use Only

Mesh type	Mixed Mesh
Mesher	Curvature based mesh
Jacobian points	4 Points
Jacobian check for shell	On
Maximum element size	7 mm
Minimum element size	1.4 mm
Mesh Quality	High
Remesh failed parts with incompatible mesh	Off

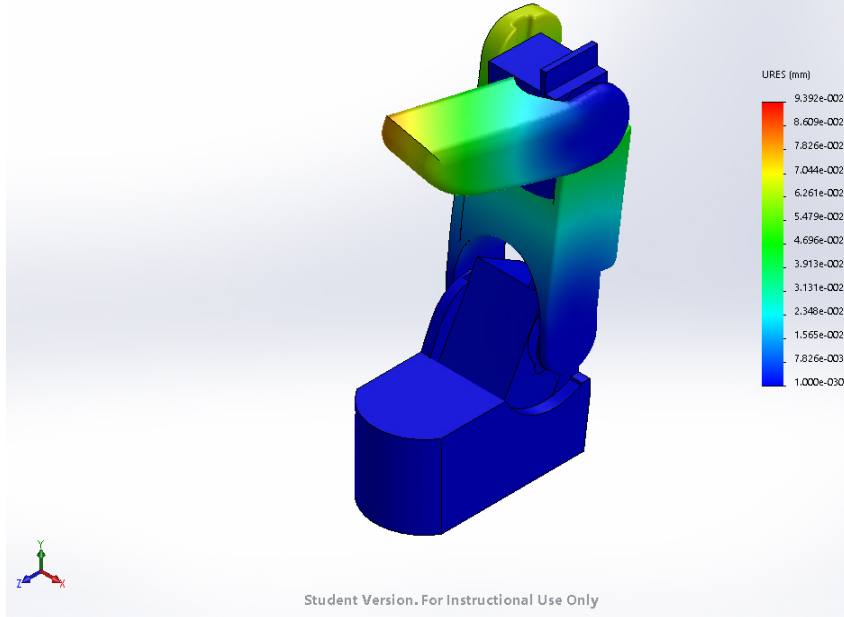
Tabell 7: Resultater FEM 1

4.1.6 Resultater



Name	Type	Min	Max
Displacement1	URES: Resultant Displacement	0 mm Node: 7	0.0939174 mm Node: 5259

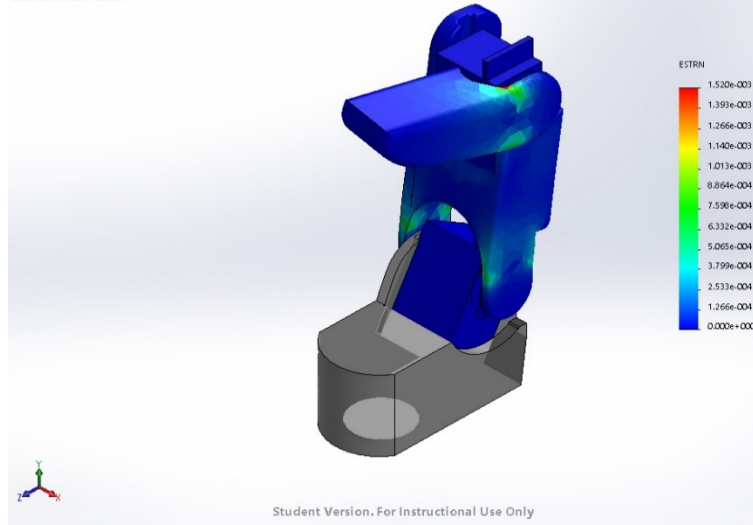
Model name: Assembly bevegelig finger sprint 4 1
Study name: Static 1 bevegelig finger 4 1 (Default)
Plot type: Static displacement Displacement1
Deformation scale: 139.064



Assembly bevegelig finger sprint 4 1-Static 1 bevegelig finger 4 1-Displacement-Displacement1

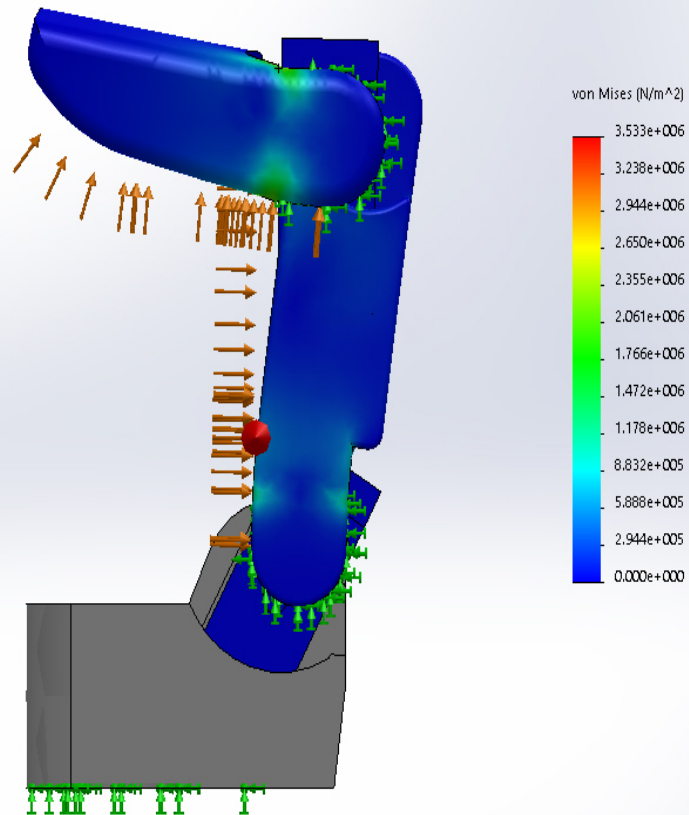
Name	Type	Min	Max
Strain1	ESTRN: Equivalent Strain	0 Element: 15374	0.0015196 Element: 1554

Model name: Assembly bevegelig finger sprint 4 1
Study name: Static 1 bevegelig finger 4 1 (Default)
Plot type: Static strain Strain1
Deformation scale: 139.064



Assembly bevegelig finger sprint 4 1-Static 1 bevegelig finger 4 1-Strain-Strain1

Model name: Assembly bevegelig finger sprint 4 1
Study name: Static 1 bevegelig finger 4 1 (-Default-)
Plot type: Static nodal stress Stress1
Deformation scale: 139.064



Student Version. For Instructional Use Only

Figur 17: Finger i analyse

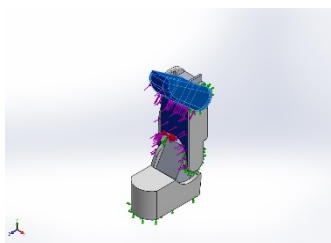
4.2 FEM - analyse 2

4.2.1 Modellinformasjon

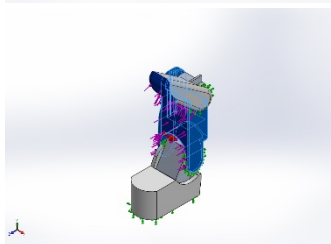
Modellnavn: Assembly bevegelig finger
sprint 4 1

Konfigurasjon: Default

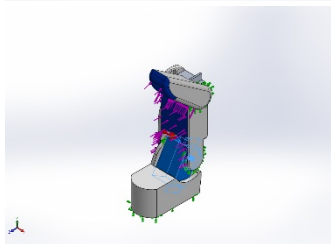
Solid Bodies



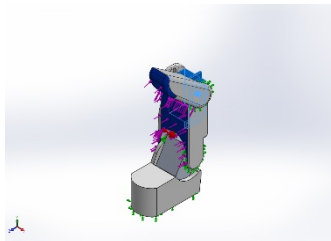
Mass:0.0230384 kg
Volume:2.25867e-005 m³
Density:1020 kg/m³
Weight:0.225777 N



Mass:0.0333176 kg
Volume:3.26643e-005 m³
Density:1020 kg/m³
Weight:0.326513 N



Mass:0.0624323 kg
Volume:2.22179e-005 m³
Density:2810 kg/m³
Weight:0.611837 N



Mass:0.0624323 kg
Volume:2.22179e-005 m³
Density:2810 kg/m³
Weight:0.611837 N

4.2.2 Studieegenskaper

Tabell 8: Studieegenskaper FEM 2

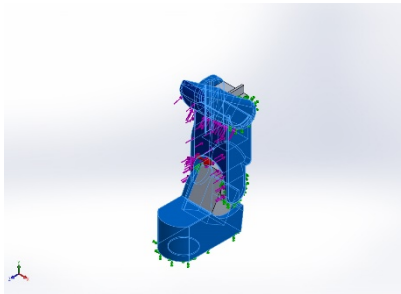
Study name	Static 2
Analysis type	Static
Mesh type	Mixed Mesh
Thermal Effect:	On
Thermal option	Include temperature loads
Zero strain temperature	298 Kelvin
Include fluid pressure effects from SOLIDWORKS Flow Simulation	Off
Solver type	FFEPlus
Inplane Effect:	Off
Soft Spring:	Off
Inertial Relief:	Off
Incompatible bonding options	Automatic
Large displacement	Off
Compute free body forces	On
Friction	Off
Use Adaptive Method:	Off
Result folder	SOLIDWORKS document (C:\Users\Aina\Documents\Bachelor Ing\Ing 3. år\HOVEDOPPGAVE\SolidWorks arbeider\Finger sprint 4)

4.2.3 Enheter

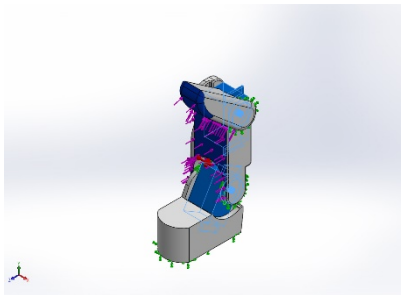
Tabell 9: Enheter FEM 2

Unit system:	SI (MKS)
Length/Displacement	mm
Temperature	Kelvin
Angular velocity	Rad/sec
Pressure/Stress	N/m ²

4.2.4 Materialegenskaper



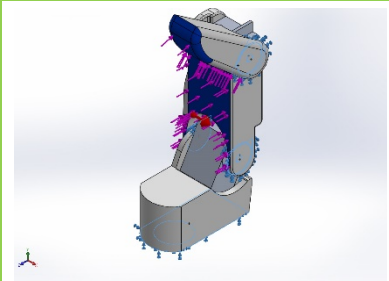
Name: ABS
Model type: Linear Elastic
 Isotropic
Default failure criterion: Unknown
Tensile strength: $3e+007 \text{ N/m}^2$
Elastic modulus: $2e+009 \text{ N/m}^2$
Poisson's ratio: 0.394
Mass density: 1020 kg/m^3
Shear modulus: $3.189e+008 \text{ N/m}^2$

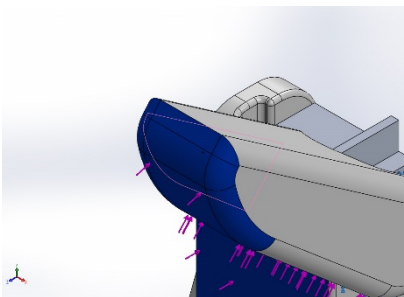
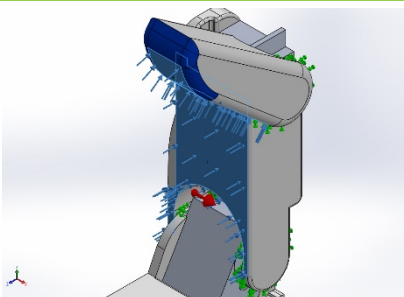


Name: 7075-T6 (SN)
Model type: Linear Elastic
 Isotropic
Default failure criterion: Unknown
Yield strength: $5.05e+008 \text{ N/m}^2$
Tensile strength: $5.7e+008 \text{ N/m}^2$
Elastic modulus: $7.2e+010 \text{ N/m}^2$
Poisson's ratio: 0.33
Mass density: 2810 kg/m^3
Shear modulus: $2.69e+010 \text{ N/m}^2$
Thermal expansion coefficient: $2.4e-005 / \text{Kelvin}$

4.2.5 Laster og låsninger

Tabell 10: Laster og låsninger FEM 2

Fixed-1		Entities: 8 face(s) Type: Fixed Geometry		
Resultantkraft				
Components	X	Y	Z	Resultant
Reaction force(N)	-1.77776	-18.9725	37.3369	41.9185
Reaction Moment(N.m)	0	0	0	1e-033

Gravity-1		Reference: Face< 1 > Values: 0 0 -9.81 Units: SI
Force-1		Entities: 3 face(s) Type: Apply normal force Value: 15 N

4.2.6 Mesh informasjon

Tabell 11: Mesh informasjon

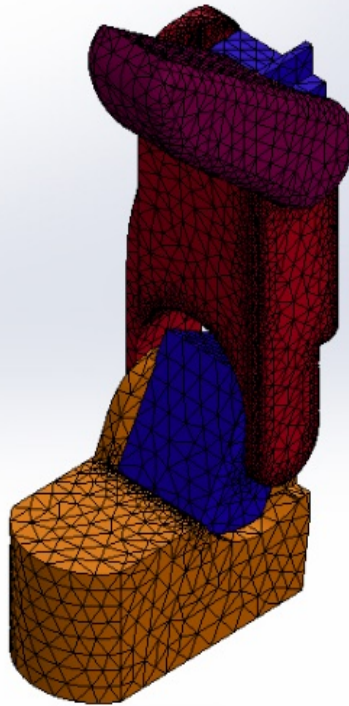
Mesh type	Mixed Mesh
Mesher Used:	Curvature based mesh
Jacobian points	4 Points
Jacobian check for shell	On
Maximum element size	5 mm
Minimum element size	1 mm
Mesh Quality	High
Remesh failed parts with incompatible mesh	Off

4.3.7 Mesh detaljer

Tabell 12: Mesh detaljer

Total Nodes	74863
Total Elements	44610
Time to complete mesh(hh:mm:ss):	00:00:05
Computer name:	AINA-PC

Model name: Assembly bevegelig finger sprint 4.1
Study name: Static 2(-Default-)
Mesh type: Mixed Mesh



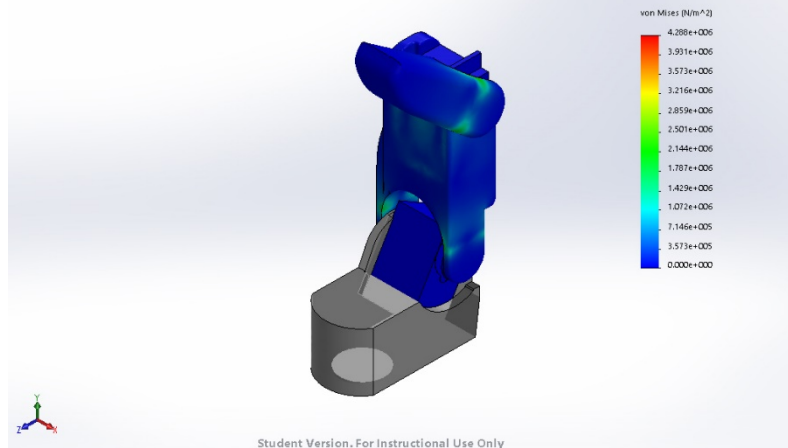
Student Version. For Instructional Use Only

4.2.8 Resultater

Tabell 13: Resultater FEM 2

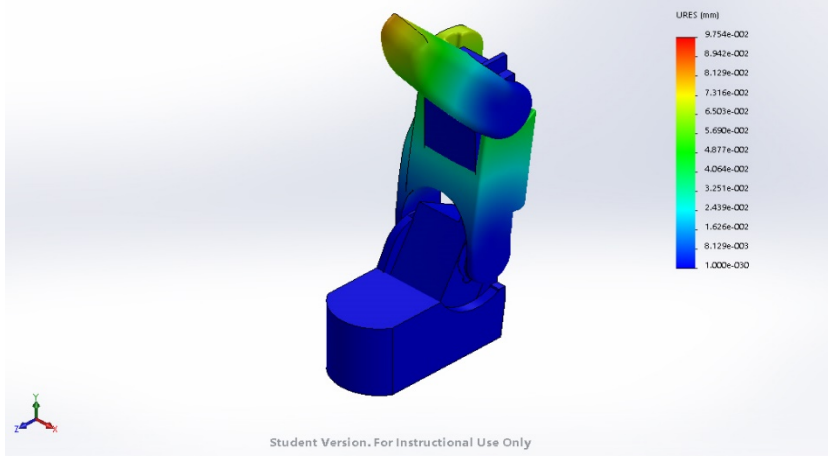
Navn	Type	Min	Max
Stress1	VON: von Mises Stress	0 N/m ² Node: 46428	4.28788e+006 N/m ² Node: 1199

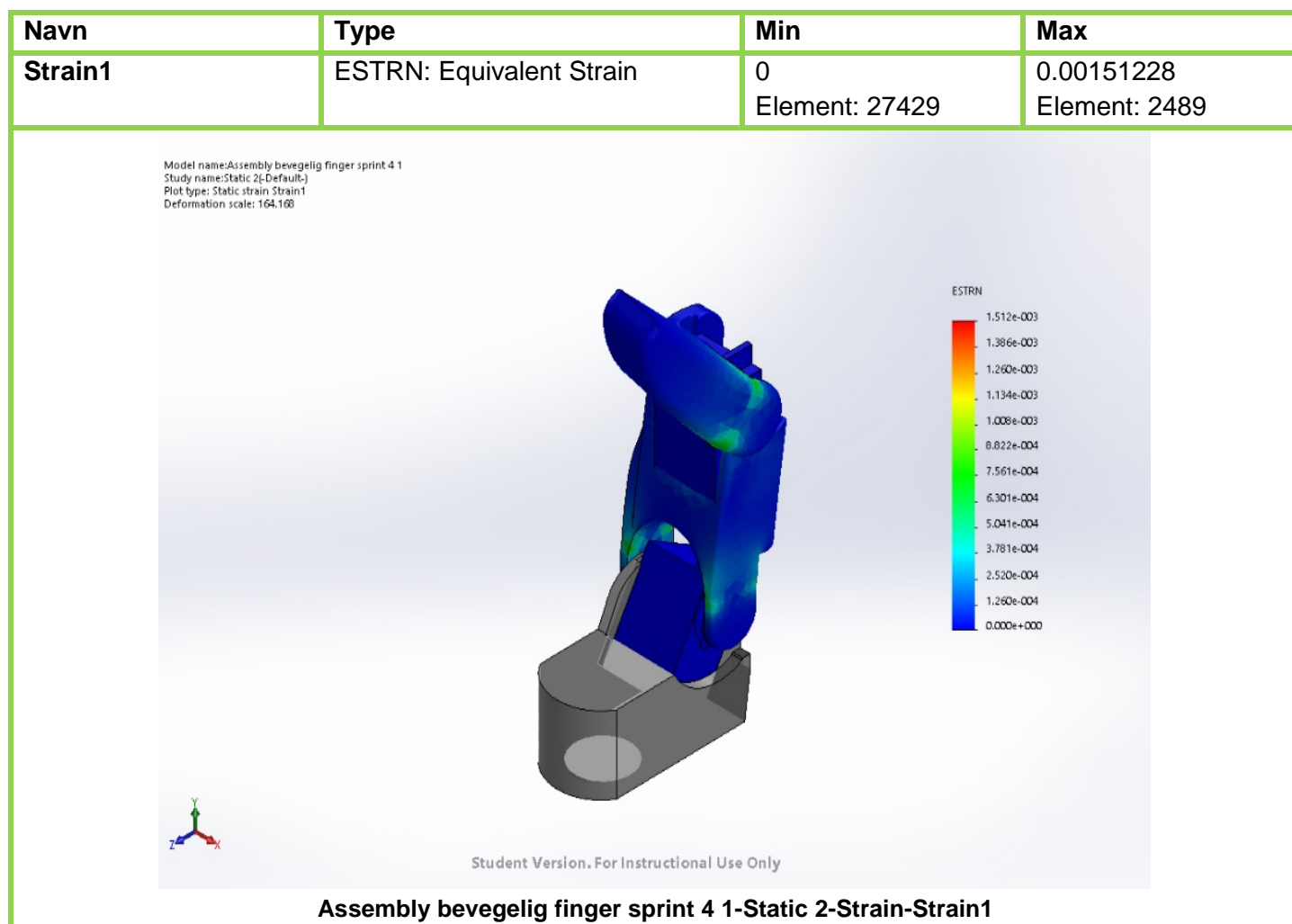
Model name: Assembly bevegelig finger sprint 4 1
 Study name: Static 2 (Default)
 Plot type: Static modal stress Stress1


Assembly bevegelig finger sprint 4 1-Static 2-Stress-Stress1

Navn	Type	Min	Max
Displacement1	URES: Resultant Displacement	0 mm Node: 7	0.0975439 mm Node: 6295

Model name: Assembly bevegelig finger sprint 4 1
 Study name: Static 2 (Default)
 Plot type: Static displacement Displacement1
 Deformation scale: 166.168


Assembly bevegelig finger sprint 4 1-Static 2-Displacement-Displacement1



4.2.9 Konklusjon

Vi ser av analysene at fingeren vil takle de krefter den skal utsettes for, vi er under materialets bruddgrense. Dette er vi fornøyd med, og velger å ta frem vår første fungerende prototype.

5 Referanser

[1] Barret Technology, LLC. (2016). *The amazing BarretHand*.

Tilgjengelig: <http://www.barrett.com/products-hand.htm>

Sist besøkt: 19.05.16

[2] Eget teknisk dokument elektronikk, kapittel 7 servomotor

[3] E. Simo-Serra, A. Perez-Gracia, H. Moon, N. Robson .(2012)

Kinematic Synthesis of Multi-Fingered Robotic Hands for Finite and Infinitesimal Tasks

Tilgjengelig: http://link.springer.com/chapter/10.1007%2F978-94-007-4620-6_22 Sist besøkt: 15.05.16

[4] Wikipedia. (2016) *Degree of freedom (DOF)*.

Tilgjengelig: https://en.wikipedia.org/wiki/Kinematic_chain

Sist besøkt: 16.05.16

[5] Learn engineering.org and Imajey Consulting Engineers pvt. (2011). *Use of Von Mises stress*.

Tilgjengelig fra: <http://www.learnengineering.org/2012/12/what-is-von-mises-stress.html>

Sist besøkt: 10.05.16

RISIKOANALYSE

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

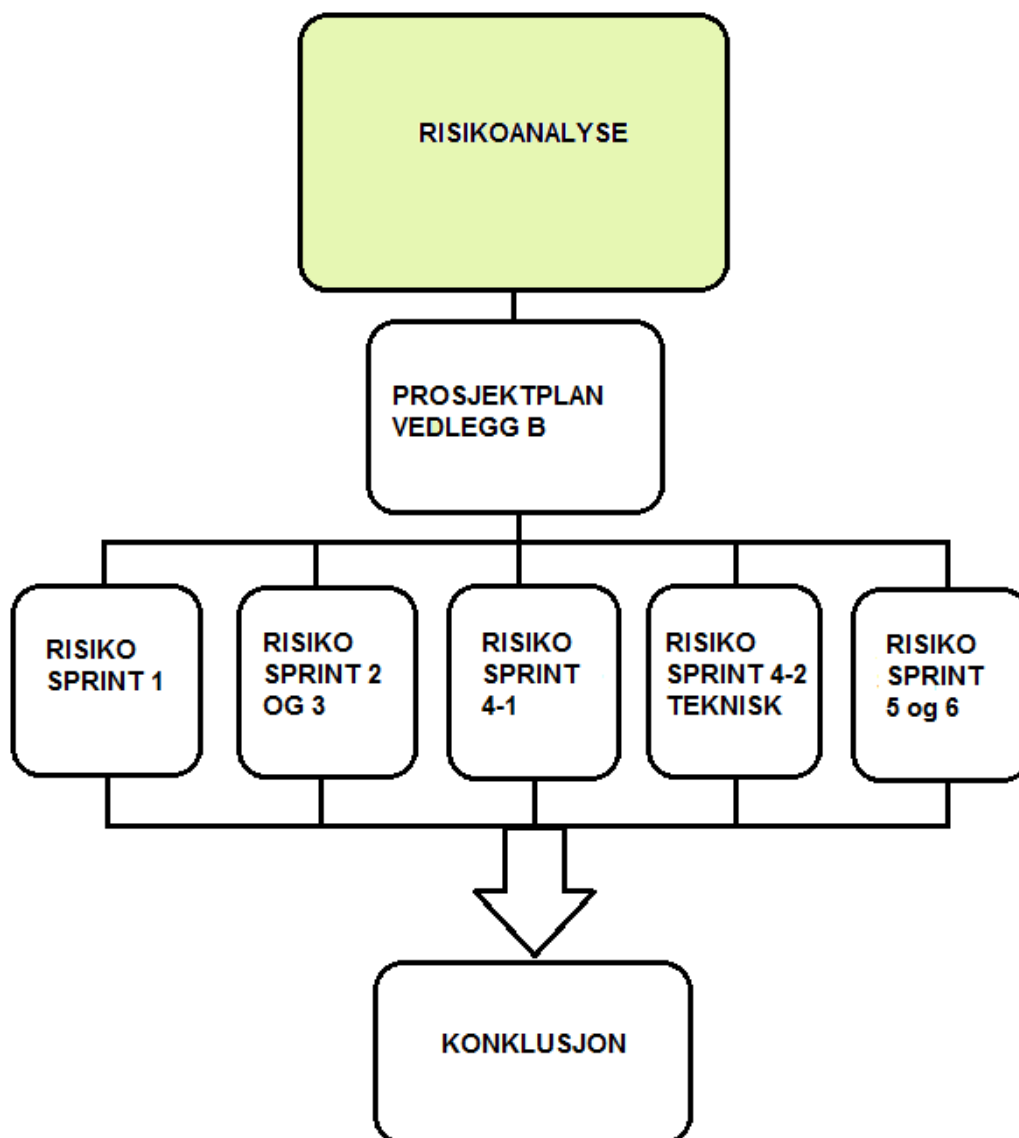
PROSJEKT	Automated CROWS testing end effector
OPPDRAKSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Risikoanalyse
VERSJON / DATO	2.0 / 18.05.2016
FORFATTER	Aina Engen Nilsen
SIDEANTALL	130



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER





I. Abstrakt

Risikoanalysen du nå holder i hånden er en samling av alle del - analyser prosjektet igjennom. Dokumentet er ordnet slik at det følger en oversikt foran hver enkelt med den tanke å forenkle navigeringen. Den typiske innholdsfortegnelsen er erstattet av en dokumentoversikt, og tabelloversikt er utelatt grunnet omfang.

Vedlagt er håndteringsverktøy brukt for risikohåndtering underveis i prosessen.

II. Samlet dokumenthistorie

Versjon 0.1	31.01.16	Opprettet av Aina E. Nilsen
Versjon 0.2	01.02.16	Endringer av Aina E. Nilsen
Versjon 0.3	02.02.16	Innhold ved Aina.
Versjon 0.4	05.02.16	Innhold ved Aina.
Versjon 0.5	07.02.16	Innhold ved Aina.
Versjon 0.6	09.02.16	Innhold, finpuss og ferdigstilling – Aina.
Versjon 1.0	09.02.16	Aina: FERDIGSTILT: opprettet versjon 1.0
Versjon 1.1	26.02.16	Thomas/Aina: gjennomgang og oppdatering risiko.
Versjon 1.2	09.03.16	Oppdatering og tilføyning teknisk risiko av Aina
Versjon 1.3	10.03.16	Tillegg og oppdatering risiko av Aina.
Versjon 1.4	14.04.16	Tillegg og oppdatering risiko for sprint 4 av Aina
Versjon 1.5	28.04.16	Tillegg og oppdatering risiko for sprint 4 av Aina
Versjon 1.6	02.05.16	Oppdatering sprint 4, retting (samt bruk mot test) av Aina
Versjon 1.7	11.05.16	Tillegg og oppdatering risiko for sprint 5 og 6, samt klargjøring ot innlevering dokumentasjon av Aina.
Versjon 2.0	18.05.16	Sammenstilling av stort dokument for siste levering. Finpuss og korrektur – Ferdigstilling dokument av Aina.

III. Innhold

	INNLEDNING	s.6
	KATEGORISERING SUKSESSFAKTORER	s.7
Dokument:	Prosjektplan Vedlegg B	s.9
Dokument:	Risiko sprint 1	s.34
Dokument:	Risiko sprint 2 og 3	s.58
Dokument:	Risiko sprint 4 -1	s.73
Dokument:	Risiko sprint 4 -2	s.91
Dokument:	Risiko sprint 5 og 6	s.108
	KONKLUSJON	s.125
	REFERANSER	s.127

IV. Vedleggsoversikt

1. Risikohåndtering sprint 4 - 1
2. Risikohåndtering sprint 4 - 2
3. Risikohåndtering sprint 5 og 6

V. Forkortelser

Forkortelser	Forklaring
KPS	Kongsberg Protech Systems
HSN	Høgskolen i Sørøst-Norge
ACT [2]	Automated CROWS Testing (foregående bachelorgruppe)
ACTEE	Automated CROWS Testing End Effector
EE	End Effector, gripeverktøy
CG	Control Grip, styringsjoystick for våpenstasjon
DCP	Deportable Control Panel
WS	Weapon station
CROWS	Common Remotely Operated Weapon Station
UR5 [1]	Universal Robots 5 robotarm
US	User Stories
AC	Acceptance Criterias
SM	Scrum Master
PO	Produkt Owner

1 Innledning

Risiko beskriver potensialet for å miste noe av verdi som kan påvirke prosjektets mål, vi ser dermed viktigheten av en gjennomgående risikoanalyse. Dette dokumentet gir oversikt over risiko i prosjektet, og kategorisering av disse. Risikoanalysen har økt i omfang etter hvert som prosjektet skrider frem, og kontinuerlig evaluering av scenarier har blitt tilføyet fortløpende.

Dette dokumentet sammensatt av samtlige risikoanalyser gjort gjennom hele prosjektet. Etterhvert som prosjektet har utviklet seg, ble behovet for risikohåndtering tydeligere og tydeligere. For at et prosjekt skal være lønnsomt, er det viktig med en solid risikoanalyse i bunnen. Når analysen foreligger, må vi kunne bruke denne så effektivt som mulig for håndtering av eventuelle scenario. Dette løser vi best ved å fokusere mer på mitigerende enn reaktive tiltak, og å implementere et håndteringsverktøy for de risikoscenarier som vil kunne ha størst innvirkning på vårt prosjekt.

Risikohåndtering er sammensatt, og kan knyttes opp mot både test og sikkerhet. Den vil også kunne bidra i tidsestimering av prosjekter, samt indikere kritiske områder eller faser i god tid.

Å vite noe om risiko er viktig for å kunne fatte beslutninger, både for individer, i organisasjoner/bedrifter og på samfunnsnivå. Risikoanalyse er derfor et viktig fagområde. Når vi driver risikoanalyse, må vi ta utgangspunkt i en presis og operasjonaliserbar definisjon. En måte å gjøre dette på er å si at risiko er beregnet sannsynlighet for bestemte uønskede hendelser, og sårbarheten, eller konsekvensene, av slike hendelser. Den estimerte sannsynligheten er dermed et uttrykk for usikkerheten, både med hensyn til hvorvidt uønskede hendelser vil inntreffe, og med hensyn til hva konsekvensene i så fall vil bli [3].

2 Kategorisering suksessfaktorer

Sannsynlighet for at en hendelse skal inntreffe rangeres fra 1 - 12, hvor 12 er nesten sikkert. Sårbarhet sier noe om hvor stor konsekvens av oppstått hendelse vi får, og rangeres også fra 1 - 12 hvor 12 er stor sårbarhet (tabell 2.1):

Tabell 1.1

Sannsynlighet (1-12)		Sårbarhet (1-12)	
1-3	Sjelden	1-3	Ubetydelig
4-6	Usannsynlig	4-6	Mindre
7-9	Moderat	7-9	Moderat
10-12	Sannsynlig	10-12	Stor

For å best illustrere risiko, samles data i en egen risikomatrise (tabell 2.2).

Produktet av sannsynlighet og sårbarhet gir oss en tallfestet risiko for aktuell hendelse, og deretter en fargekode for enkel oversikt.

Tabell 2.2

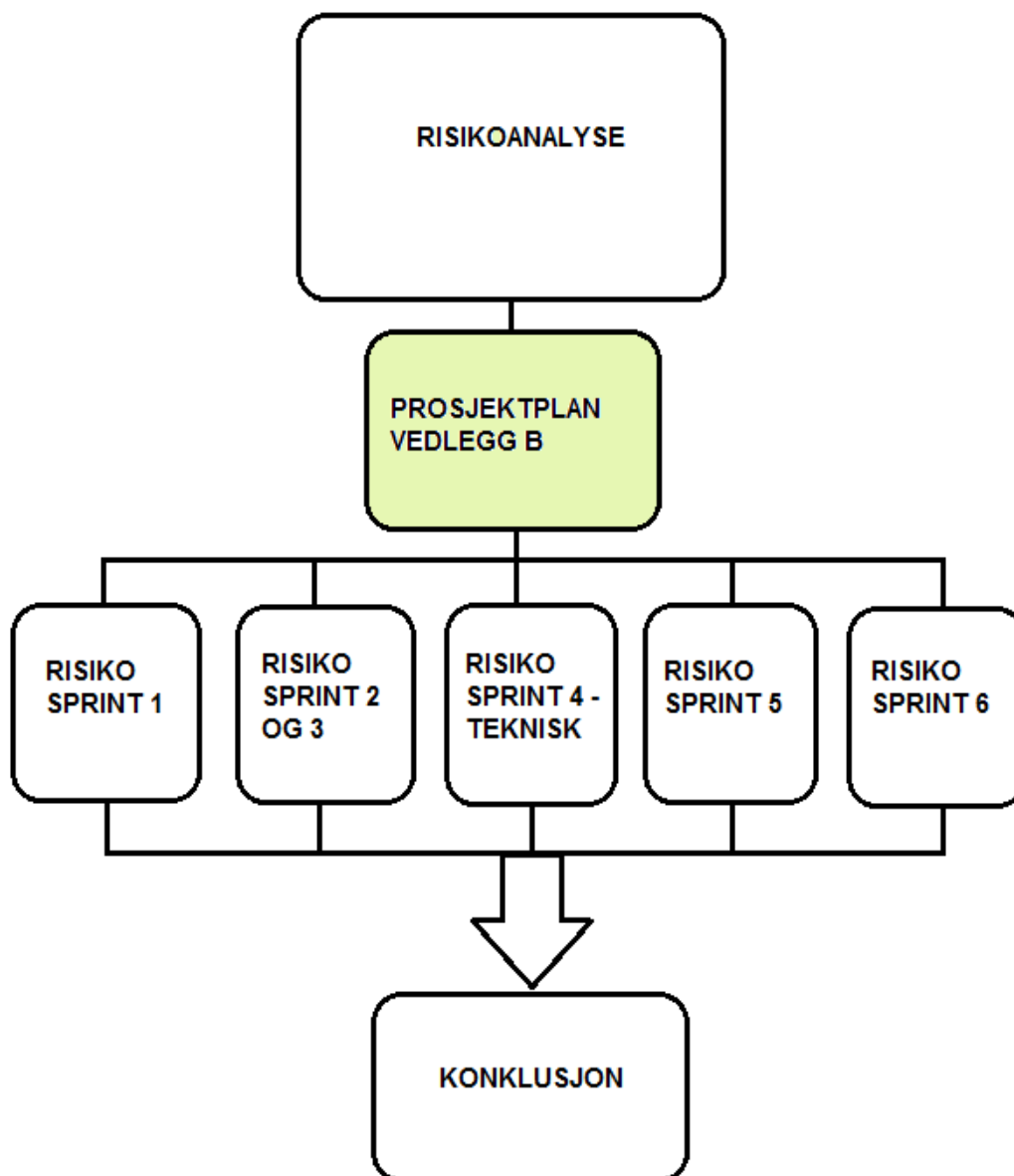
Sannsynlighet	Sårbarhet			
	Ubetydelig	Mindre	Moderat	Stor
Sannsynlig: Vil antagelig oppstå de fleste ganger	10-36	40-72	70-108	100-144
Moderat: Kan oppstå de fleste ganger	7-27	28-45	49-81	70-108
Usannsynlig: Kan oppstå en eller annen gang	4-18	16-36	28-45	40-72
Sjelden: Kan oppstå under sjeldne forhold	1-9	4-18	7-27	10-36

Grønn sone: lav risiko - håndteres ved rutineprosedyrer.

Gul sone: Moderat risiko - Ansvarsområder ved håndtering må spesifiseres.

Rød sone: Høy risiko - Bør vurderes i samråd med seniorrådgiver (erfaring og kompetanse).

Mørk rød sone: Ekstra høy risiko - Øyeblikkelig handling/tiltak kreves.



Prosjektplan Vedlegg B

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAGSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Prosjektplan Appendix B
VERSJON / DATO	1.1 / 09. Februar 2016
FORFATTER	Aina Engen Nilsen



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



2 Overordnet risikoanalyse

Personorienterte suksessfaktorer

Tabell 2.1

Risiko ID	Scenario	Konsekvens
PER - 01	Sykdom/ulykke	Gruppemedlem må trekke seg underveis
PER - 02	Dårlig Kommunikasjon	Misforståelser internt og eksternt. Vanskeligheter med å forklare / ordlegge seg
PER - 03	Tilstedeværelse	Manglende oppmøte, dårlig holdning for gruppearbeid, mangel på motivasjon
PER - 04	Interne konflikter	Uenighet i gruppen skaper splid
PER - 05	Manglende prosjektstyring	Nøkkelperson(er) for prosjektstyring uteblir, eller viser seg uegnet for oppgaven

2.1.0 Kategorisering

Tabell 2.1.1

PER - 01		Sykdom/ulykke		Gruppemedlem må trekke seg underveis	
Sannsynlighet		3	15		
Sårbarhet		5			
Konsekvens		Gruppen må fordele arbeid seg i mellom, og kompensere for eventuelle faglige mangler.			
Tiltak		a) Fordele arbeid og tilegne seg kunnskap. b) Redusere oppgavens omfang tilsvarende ressurstap. c) Få inn en erstatter.			

Tabell 2.1.2

PER - 02	Dårlig kommunikasjon		Misforståelser internt og eksternt. Vanskeligheter med å forklare/ordlegge seg.
Sannsynlighet	10	100	
Sårbarhet	10		
Konsekvens	Fører til alt fra frustrasjon til feiltolkninger, og kan lede til store feil og mangler.		
Tiltak	a) Bli enige på forhånd hvordan kommunikasjon skal foregå, og følge dette. b) Skriftlige avtaler og godkjenninger.		

Tabell 2.1.3

PER - 03	Tilstedeværelse		Manglende oppmøte, dårlig holdning til gruppearbeid, mangel på motivasjon
Sannsynlighet	7	35	
Sårbarhet	5		

Konsekvens	Merarbeid og frustrasjon. I verste fall mister gruppen medlemmer
Tiltak	a) Ha en dialog rundt hva som ønskes og forventes. b) Sørge for god dynamikk i gruppa, være bevisst på forskjeller. c) Ta opp problemet med aktuell person tidlig og på riktig måte.

Tabell 2.1.4

PER - 04		Interne konflikter		Uenighet som fører til splid i gruppen	
Sannsynlighet		6	48		
Sårbarhet		8			
Konsekvens		Forsinkelser i prosjekt, frafall av medlemmer			
Tiltak		a) Ha føringer på hvordan vi tar opp et problem. b) Ved store uenigheter som ikke lar seg løse internt, bør det hentes inn en utenforstående (eks. veileder) for bistand, eventuelt beslutning. c) Gruppen skal ha en intern arbeidskontrakt.			

Tabell 2.1.5

PER - 05		Manglende prosjektstyring		Nøkkelperson(er) for prosjektstyring uteblir, eller viser seg uegnet for oppgaven
Sannsynlighet		4	32	
Sårbarhet		8		
Konsekvens		Forsinkelser i prosjekt, ikke fullstendig produkt/feil produkt		
Tiltak		a) Velge en prosjektmodell som gir god oversikt for alle, samt har fokus på kollektiv fremdrift. b) Ha to personer med hovedansvar for fremdrift og oversikt.		

Prosjektorienterte suksessfaktorer oversikt

Tabell 2.2

Risiko ID	Scenario	Konsekvens
PRR - 01	Manglende Sikkerhetsklarering	Gruppemedlem blir ikke sikkerhetsklarert for oppgaven og må trekke seg fra gruppen
PRR - 02	Manglende kompetanse	Gruppen mangler kompetanse og får vanskelig tilegnet seg denne
PRR - 03	Tekniske problemer	Tap av filer, programvare, tekniske feil sub - disipliner, etc.
PRR - 04	Forsinkelser i arbeid og ferdigstilling	Klarer ikke overholde tidsplanen
PRR - 05	Tap av prosjekt	Oppdragsgiver går konkurs, eller er tvunget til å gjøre omgående prioriteringer av ulik årsak.
PRR - 06	Prosjektplan	Ikke realistisk, ikke avgrenset
PRR - 07	Sene endringer	Kravendringer fra oppdragsgiver sent i prosjektforløpet
PRR - 08	Budsjett	Vi mangler midler eller nødvendig økonomisk støtte for å kunne gjennomføre prosjektet tilfredsstillende.

2.2.0 Kategorisering

Tabell 2.2.1

PRR - 01	Manglende sikkerhetsklarering		Gruppemedlem blir ikke sikkerhetsklarert for oppgaven og må trekke seg fra gruppen
Sannsynlighet	1	5	
Sårbarhet	5		
Konsekvens	Omorganisering av gruppen, revurdering oppgavens innhold		
Tiltak	a) Omfordeler oppgaver/ansvar i gruppen		

	b) Får inn en erstatter
--	-------------------------

Tabell 2.2.2

PRR - 02		Manglende kompetanse	Gruppen mangler kompetanse og får vanskelig tilegnet seg denne	
Sannsynlighet		3	24	
Sårbarhet		8		
Konsekvens		Forsinkelser i prosjekt, klarer ikke levere ønsket produkt.		
Tiltak		a) Kjenne gruppens begrensninger b) Bruke tilgjengelig fagkompetanse internt og eksternt, aktivt oppsøke disse c) Tilpasse oppgavens kompleksitet best mulig		

Tabell 2.2.3

PRR - 03		Tekniske problemer		Tap av filer, programvare, tekniske feil i sub - disipliner, etc.
Sannsynlighet		10	120	
Sårbarhet		12		
Konsekvens		Merarbeid, forskyvning i tidsplanen		
Tiltak		a) Ha gode rutiner for backup, lagring og historikk		

Tabell 2.2.4

PRR - 04		Forsinkelser i arbeid og ferdigstilling		Klarer ikke overholde tidsplanen	
Sannsynlighet		8	64		
Sårbarhet		8			
Konsekvens		Forsinkelser i prosjektet som kan føre til ufullstendig resultat			
Tiltak		a) Ha en realistisk tidsplan b) Hele tiden revurdere tidsbruk og tidsplan c) Bryte ned oppgaver i mindre deler d) Revurdere vinklingen på problemstillinger og hvordan vi løser e) Revurdere oppgavens omfang			

Tabell 2.2.5

PRR - 05	Tap av prosjekt		Oppdragsgiver går konkurs, eller er tvunget til å gjøre omgående prioriteringer av ulik årsak.
Sannsynlighet	1	6	
Sårbarhet	6		
Konsekvens	Gruppen mister oppgaven og muligheten til ferdigstille utdannelsen		
Tiltak	a) Modifisere oppgaven slik at eventuell intern arbeidsgiver kan overta		

Tabell 2.2.6

PRR - 06	Prosjektplan		Ikke realistisk, ikke avgrenset
Sannsynlighet	3	15	
Sårbarhet	5		

Konsekvens	Uoversiktlig plan uten rammer gir ufullstendig produkt
Tiltak	a) Bruk tidsaspekt, gruppens kompetansesammensetning og oppgavens omfang på å sette rammer i prosjektplanen b) Oppdater og endre hyppig underveis

Tabell 2.2.7

PRR - 07	Sene endringer		Kravendringer fra oppdragsgiver sent i prosjektforløpet
Sannsynlighet	10	50	
Sårbarhet	5		
Konsekvens	Merarbeid, merutgifter og tidsutfordringer. Kan gi ufullstendig produkt		
Tiltak	a) Ha skriftlige føringer for dette tidlig i prosess		

Tabell 2.2.8

PRR - 08	Budsjett		Det mangler midler eller nødvendig økonomisk støtte for å kunne gjennomføre prosjektet tilfredsstillende.
Sannsynlighet	3	9	
Sårbarhet	3		
Konsekvens	Prosjektets produkt ville være mangelfullt		
Tiltak	a) Klareres skriftlig tidlig i prosessen		

3 Spesifikk risikoanalyse

Oversikt risiko elektro

Tabell 3.1

Risiko ID	Scenario Elektro
ELR - 01	Feil på bestilt komponent
ELR - 02	Bestilt komponent ødelagt eller forsvinner i frakt
ELR - 03	Overspenning
ELR - 04	For lite strøm fra UR5
ELR - 05	Elektromagnetisk forstyrrelse
ELR - 06	Feil på utregning av krets
ELR - 07	Feilkobling
ELR - 08	Komponenter som ryker gjennom test
ELR - 09	Strømforsyningssvikt
ELR - 10	Feil på simulering
ELR - 11	Simuleringsprogram får feilmelding
ELR - 12	Mistet dokument på krets
ELR - 13	Feil på tegning av krets

3.1.0 Kategorisering

Tabell 3.1.1

ELR - 01		Feil på bestilt komponent		
Sannsynlighet		4	48	
Sårbarhet		12		
Konsekvens		Merarbeid, merkostnad, forsinkelser		
Tiltak		a) Kontrollere komponent ved mottakelse.		

Tabell 3.1.2

ELR - 02	Bestilling forsvinner eller ødelegges under frakt		
Sannsynlighet	4	48	
Sårbarhet	12		
Konsekvens	Tidsplan forskyves, merkostnader		
Tiltak	a) Sporing av pakke b) Kontrollere pakke ved mottakelse		

Tabell 3.1.3

ELR - 03	Overspenning		
Sannsynlighet	3	36	
Sårbarhet	12		

Konsekvens	Ødeleggelser, tidsplan forskyves, merkostnader
Tiltak	a) Kontroll/" overspenningsvern" b) Forsikre at riktig mengde spenning/strøm føres til komponenter

Tabell 3.1.4

ELR - 04		For lite strøm fra UR5		
Sannsynlighet		5	40	
Sårbarhet		8		
Konsekvens		Systemet virker ikke		
Tiltak		a) Ekstra strømkilde/en booster		

Tabell 3.1.5

ELR - 05		Elektromagnetisk forstyrrelse		
Sannsynlighet		8	64	
Sårbarhet		8		
Konsekvens		Signalforstyrrelser, systemet virker ikke optimalt		
Tiltak		a) Isolering/filtrering av støy		

Tabell 3.1.6

ELR - 06		Feil på utregning av krets			
Sannsynlighet		1	5		
Sårbarhet		5			
Konsekvens		Systemet virker ikke			
Tiltak		a) Kontrollering før utnytting			

Tabell 3.1.7

ELR - 07	Feilkobling		
Sannsynlighet	5	60	
Sårbarhet	12		
Konsekvens			
Tiltak	a) Kontrollering før testing		

Tabell 3.1.8

ELR - 08		Komponenter som ryker i test		
Sannsynlighet		9	108	
Sårbarhet		12		
Konsekvens		Merkostnad, tidsforskyvninger		
Tiltak		a) Bytte komponent		

Tabell 3.1.9

ELR - 09		Strømsvikt		
Sannsynlighet		1	1	
Sårbarhet		1		
Konsekvens	System virker ikke			
Tiltak	a) Kontroll av kilde? b) Slå av/på sikring?			

Tabell 3.1.10

ELR - 10		Feil på simulering			
Sannsynlighet		4	4		
Sårbarhet		1			
Konsekvens		Forsinkelser			
Tiltak		a) Kontrollering			

Tabell 3.1.11

ELR - 11		Simuleringsprogram får feilmelding		
Sannsynlighet		6	36	
Sårbarhet		6		
Konsekvens	Forsinkelser			
Tiltak	a) Restart av program/lagre arbeid ofte			

Tabell 3.1.12

ELR - 12		Mistet dokument på krets		
Sannsynlighet		4	40	
Sårbarhet		10		
Konsekvens		Merarbeid, forsinkelser		
Tiltak		a) Jevnlig backup av dokumenter		

Tabell 3.1.13

ELR - 13		Feil på tegning av krets			
Sannsynlighet		4	16		
Sårbarhet		4			
Konsekvens		Merarbeid			
Tiltak		a) Kontrollering av krets ved simulering			

Oversikt risiko data

Tabell 3.2

Risk ID	Scenario
DAR - 01	Tidkrevende feil i kode.
DAR - 02	Mengden kompetanse som må tilegnes ila. prosjektet er for stor.
DAR - 03	Arbeidsmengde blir for stor.
DAR - 04	Skade på prototype forsinker programmering.
DAR - 05	Ikke klarer tidsfrist.
DAR - 06	Tap av data.
DAR - 07	Hardware som ikke er kompatibel med Software.
DAR - 08	Gold plating ("forgyelling")

3.2.0 Kategorisering

Tabell 3.2.1

DAR - 01	Tidkrevende feil i kode		
Sannsynlighet	12	96	
Sårbarhet	8		
Konsekvens	Forsinkelser, merarbeid		
Tiltak	a) Planlegge for feilretting, debugging		

Tabell 3.2.2

DAR - 02	Mengden kompetanse som må tilegnes ila. prosjektet er for stor.		
Sannsynlighet	8	80	
Sårbarhet	10		
Konsekvens	Forskyvning tidsplan, dårlig motivasjon og dertil fremdrift. Redusert kvalitet		
Tiltak	a) Skaffe en oversikt over nødvendig kompetanse og forberede kunnskapene.		

Tabell 3.2.3

DAR - 03	Arbeidsmengde blir for stor		
Sannsynlighet	10	80	
Sårbarhet	8		
Konsekvens	Dårlig motivasjon, dårlig gruppedynamikk og arbeidsflyt		
Tiltak	a) Gjøre riktige prioriteringer. b) Bruke tid på analyse av nytteverdi.		

Tabell 3.2.4

DAR - 04	Skade på prototype forsinker programmering		
Sannsynlighet	2	20	
Sårbarhet	10		

Konsekvens	Merkostnad, merarbeid og forsinket prosjekt
Tiltak	a) Produsere flere prototyper b) Retningslinjer i forhold til bruk av EE tilkoblet UR5

Tabell 3.2.5

DAR - 05	Ikke klarer tidsfrist.		
Sannsynlighet	10	50	
Sårbarhet	5		
Konsekvens	Forsinkelse i prosjektet, merarbeid og merkostnad		
Tiltak	a) Sette frister for ferdigstilling med margin til kritisk deadline.		

Tabell 3.2.6

DAR – 06	Tap av data		
Sannsynlighet	3	30	
Sårbarhet	10		
Konsekvens	Merarbeid		
Tiltak	a) Ha rutiner på backup av kildekode.		

Tabell 3.2.7

DAR – 07	Hardware som ikke er kompatibel med software		
Sannsynlighet	3	15	
Sårbarhet	5		
Konsekvens	Merarbeid, forsinkelser, merkostnader		
Tiltak	a) Sørge for god kommunikasjon innad i gruppen.		

Tabell 3.2.8

DAR – 08	Gold plating		
Sannsynlighet	3	9	
Sårbarhet	3		
Konsekvens			
Tiltak	a) Tidsestimere user stories, og strebe etter å holde disse estimatene.		

Oversikt risiko maskin

Tabell 3.3

Risk ID	Scenario
MAR - 01	Kommunikasjon subsystemer
MAR - 02	Tap av filer i Solid Works
MAR - 03	Ufullstendige 2D tegninger
MAR - 04	Manglende forståelse ved oppbygging av modell
MAR - 05	Feil bruk av relasjoner ved modellering
MAR - 06	Feil input i FEM – analyse
MAR - 07	Unøyaktig målsetting av deler under tegning
MAR - 08	Feil materialvalg
MAR - 09	For kompleks modell
MAR - 10	Ødelagt prototype/modell under test

3.3.0 Kategorisering

Tabell 3.3.1

MAR - 01	Kommunikasjon subsystemer		Dårlig kommunikasjon mellom maskin og elektro i forhold til styring og plassering av komponenter
Sannsynlighet	8	80	
Sårbarhet	10		
Konsekvens	Unødvendig bruk av ressurser, feil på modell, tidsforskyvning, merarbeid		
Tiltak	a) Kommunikasjon går gjennom ansvarlige for sitt felt b) Jobbe frem system sammen		

Tabell 3.3.2

MAR - 02	Tap av filer i Solid Works		Mister arbeid underveis i modellering, feil navngivning av filer, programhavari
Sannsynlighet	7	70	
Sårbarhet	10		
Konsekvens	Ufullstendig modell, tidsforskyvning og merarbeid		
Tiltak	a) Lagre hyppig under arbeid b) Sikkerhetskopi c) Sørge for å arbeide på maskin med tilstrekkelig kapasitet		

Tabell 3.3.3

MAR – 03	Ufullstendige 2D tegninger		
Sannsynlighet	6	72	
Sårbarhet	12		
Konsekvens	Feil i modell, ikke kompatibel med subsystemer		
Tiltak	a) Være nøye med mål b) Be om veiledning		

Tabell 3.3.4

MAR - 04	Manglende forståelse ved oppbygging av modell	Tar ikke hensyn til plan eller punkter, setter på ufordelaktige tilhørigheter og mål	
Sannsynlighet	4	36	
Sårbarhet	9		
Konsekvens	Usammenhengende modell, ikke brukbar modell, merarbeid, forsinkelser		

Tiltak	a) Sørge for å ha kontroll på plan og origo b) Be om veiledning tidlig i prosess
--------	---

Tabell 3.3.5

MAR – 05		Feil bruk av relasjoner ved modellering		
Sannsynlighet		4	24	
Sårbarhet		6		
Konsekvens		Tungt håndterlig modell, merarbeid, forsinkelser		
Tiltak		a) Være bevisst i forhold til relasjoner		

Tabell 3.3.6

MAR – 06		Feil input i FEM - analyse		Setter inn feil verdier i forhold til materialstyrke og belastninger	
Sannsynlighet		6	66		
Sårbarhet		11			
Konsekvens		Feil informasjon ift styrke og holdbarhet, tåler ikke planlagte belastninger, kort levetid, dyr			
Tiltak		a) Kontrollere inputs før analyse kjøres b) Være kritisk ovenfor resultater c) Be om veiledning			

Tabell 3.3.7

MAR - 07	Unøyaktig målsetting av deler under tegning		
Sannsynlighet	3	12	
Sårbarhet	4		
Konsekvens	Ikke forventet modell, inkompatibel med resten av system, merarbeid, merkostnad		
Tiltak	a) Ha god kommunikasjon mellom subsystemer og innad på felt		

Tabell 3.3.8

MAR – 08	Feil materialvalg		
Sannsynlighet	4	48	
Sårbarhet	12		
Konsekvens	Kostbar modell, uholdbar modell, for tung modell		
Tiltak	a) Grundig forstudie b) Be om veiledning		

Tabell 3.3.9

MAR – 09	For kompleks modell		Modellens kompleksitet stemmer ikke overens med utviklers kompetanse, eller tiltenkt bruk
Sannsynlighet	2	12	
Sårbarhet	6		
Konsekvens	For tidkrevende og dyr modell, rom for mange feil under utvikling, tidsforskyvning, merkostnad		

Tiltak	a) Være bevisst sine faglige begrensninger b) Kjenne prosjektets avgrensninger c) Søke veiledning
--------	---

Tabell 3.3.10

MAR - 10	Ødelagt prototype/modell under test		
Sannsynlighet	5	50	
Sårbarhet	10		
Konsekvens	Merarbeid, merkostnad, forsinkelser i prosjekt		
Tiltak	a) Sørge for optimalt materialvalg b) Bruke høy designfaktor for sikkerhet c) Ha planlagt to prototyper		

Tabell 3.3.11

MAR - 11	Dårlig kontakt, overflater sklir mot hverandre		Manglende eller for liten friksjon mellom kontaktflater
Sannsynlighet	12	84	
Sårbarhet	7		
Konsekvens	Ikke presis, unødvendig slitasje, merkostnad		
Tiltak	a) Kle overflate med f.eks. silikon eller gummi		

Tabell 3.3.12

MAR - 12	For stor belastning eller vibrasjoner		
Sannsynlighet	7		

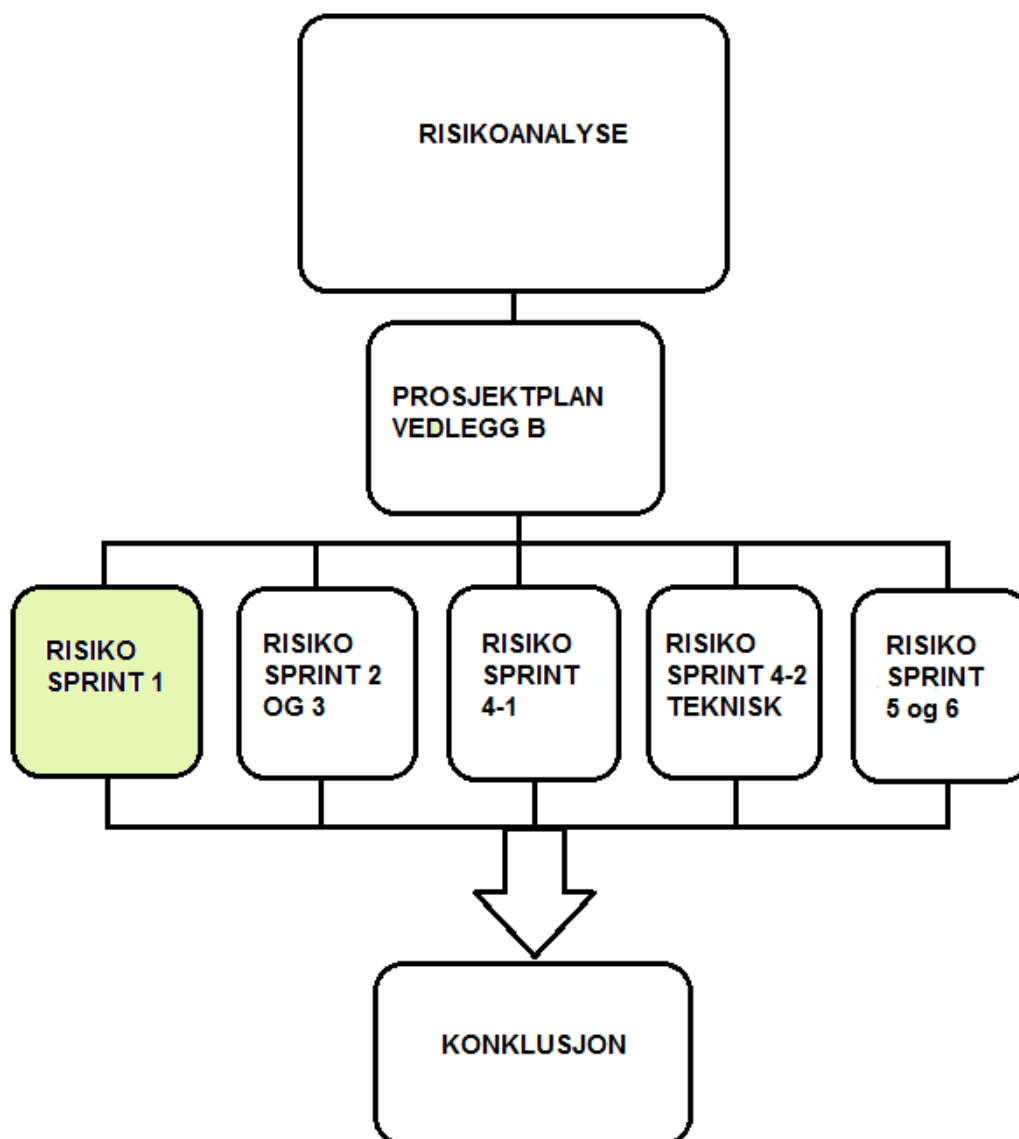
Sårbarhet	10	70	
Konsekvens	Brudd / tretthetsbrudd, merkostnad, ikke presis.		
Tiltak	a) Endre vinkler på utstyr. b) Kontrollere innfestninger.		

Tabell 3.3.13

MAR - 13	Brudd på deler		
Sannsynlighet	10	120	
Sårbarhet	12		
Konsekvens	Tidsforskyvning i prosjekt, merarbeid og merkostnad.		
Tiltak	a) Forsterke konstruksjon. b) Endre vinkel. c) Vurdere materialvalg		

Tabell 3.3.14

MAR - 14	Treghet/stor motstand pga friksjon i ledd		
Sannsynlighet	10	70	
Sårbarhet	7		
Konsekvens	Ikke ønsket funksjon/effekt, defekte deler/system.		
Tiltak	a) Sette inn lager. b) Større pasninger.		



Risiko sprint 1

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAGSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Prosjektplan Appendix B
VERSJON / DATO	1.1 / 26. Februar 2016
FORFATTER	Aina Engen Nilsen



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



Dokumenthistorie

Versjon 0.1	02.02.16	Opprettet av Aina E. Nilsen
Versjon 0.2	08.02.16	Endringer av Aina E. Nilsen: satt inn utkast risk for maskin, satt inn topp og bunntekst iht dokumentstil. Disposisjon. Endret dokumentnavn. Satt inn fire risiko maskin,
Versjon 1.0	08.02.16	Aina: FERDIGSTILT: opprettet versjon 1.0
Versjon 1.1	26.02.16	Thomas/Aina: gjennomgang og oppdatering risiko.

2 - OVERORDNET RISIKOANALYSE

Personorienterte suksessfaktorer

Tabell 2.1

Risiko ID	Scenario	Konsekvens
PER - 01	Sykdom/ulykke	Gruppemedlem må trekke seg underveis
PER - 02	Dårlig Kommunikasjon	Misforståelser internt og eksternt. Vanskeligheter med å forklare / ordlegge seg
PER - 03	Tilstedeværelse	Manglende oppmøte, dårlig holdning for gruppearbeid, mangel på motivasjon
PER - 04	Interne konflikter	Uenighet i gruppen skaper splid
PER - 05	Manglende prosjektstyring	Nøkkelperson(er) for prosjektstyring uteblir, eller viser seg uegnet for oppgaven

2.1.0 Kategorisering

Tabell 2.1.1

PER - 01		Sykdom/ulykke		Gruppemedlem må trekke seg underveis	
Sannsynlighet		3	15		
Sårbarhet		5			
Konsekvens		Gruppen må fordele arbeid seg i mellom, og kompensere for eventuelle faglige mangler.			
Tiltak		a) Fordele arbeid og tilegne seg kunnskap. b) Redusere oppgavens omfang tilsvarende ressurstap. c) Få inn en erstatte.			

Tabell 2.1.2

PER - 02		Dårlig kommunikasjon		Misforståelser internt og eksternt. Vanskeligheter med å forklare/ordlegge seg.
Sannsynlighet		6	60	
Sårbarhet		10		
Konsekvens		Fører til alt fra frustrasjon til feiltolkninger, og kan lede til store feil og mangler.		
Tiltak		a) Bli enige på forhånd hvordan kommunikasjon skal foregå, og følge dette. b) Skriftlige avtaler og godkjenninger c) Økt fokus på kommunikasjon i form av medarbeidersamtaler		

Tabell 2.1.3

PER - 03	Tilstedeværelse		Manglende oppmøte på bakgrunn av dårlig holdning til gruppearbeid, mangel på motivasjon
Sannsynlighet	2	10	
Sårbarhet	5		

Konsekvens	Merarbeid og frustrasjon. I verste fall mister gruppen medlemmer
Tiltak	a) Ha en dialog rundt hva som ønskes og forventes. b) Sørge for god dynamikk i gruppa, være bevisst på forskjeller. c) Ta opp problemet med aktuell person tidlig og på riktig måte.

Tabell 2.1.4

PER - 04		Interne konflikter		Uenighet som fører til splid i gruppen	
Sannsynlighet		3	24		
Sårbarhet		8			
Konsekvens		Forsinkelser i prosjekt, frafall av medlemmer			
Tiltak		a) Ha føringer på hvordan vi tar opp et problem. b) Ved store uenigheter som ikke lar seg løse internt, bør det hentes inn. En utenforstående (eks. veileder) for bistand, eventuelt beslutning. c) Gruppen skal ha en intern arbeidskontrakt. d) Fokus på kommunikasjon i form av medarbeidersamtaler			

Tabell 2.1.5

PER - 05	Manglende prosjektstyring		Nøkkelperson(er) for prosjektstyring uteblir, eller viser seg uegnet for oppgaven
Sannsynlighet	2	16	
Sårbarhet	8		
Konsekvens	Forsinkelser i prosjekt, ikke fullstendig produkt/feil produkt		
Tiltak	a) Velge en prosjektmodell som gir god oversikt for alle, samt har fokus på kollektiv fremdrift. b) Ha to personer med hovedansvar for fremdrift og oversikt.		

Prosjektorienterte suksessfaktorer

Tabell 2.2

Risiko ID	Scenario	Konsekvens
PRR - 02	Manglende kompetanse	Gruppen mangler kompetanse og får vanskelig tilegnet seg denne
PRR - 03	Tekniske problemer	Tap av filer, programvare, tekniske feil sub - disipliner, etc.
PRR - 04	Forsinkelser i arbeid og ferdigstilling	Klarer ikke overholde tidsplanen
PRR - 05	Tap av prosjekt	Oppdragsgiver går konkurs, eller er tvunget til å gjøre omgående prioriteringer av ulik årsak.
PRR - 06	Prosjektplan	Ikke realistisk, ikke avgrenset
PRR - 07	Sene endringer	Kravendringer fra oppdragsgiver sent i prosjektforløpet
PRR - 08	Budsjett	Vi mangler midler eller nødvendig økonomisk støtte for å kunne gjennomføre prosjektet tilfredsstillende.

2.2.0 Kategorisering

Tabell 2.2.2

PRR - 02		Manglende kompetanse		Gruppen mangler kompetanse og får vanskelig tilegnet seg denne	
Sannsynlighet		3	24		
Sårbarhet		8			
Konsekvens		Forsinkelser i prosjekt, klarer ikke levere ønsket produkt.			
Tiltak		a) Kjenne gruppens begrensninger b) Bruke tilgjengelig fagkompetanse internt og eksternt, aktivt oppsøke disse c) Tilpasse oppgavens kompleksitet best mulig			

Tabell 2.2.3

PRR - 03		Tekniske problemer		Tap av filer, programvare, tekniske feil i sub - disipliner, etc.
Sannsynlighet		6	76	
Sårbarhet		12		
Konsekvens		Merarbeid, forskyvning i tidsplanen		
Tiltak		a) Ha gode rutiner for backup, lagring og historikk		

Tabell 2.2.4

PRR - 04		Forsinkelser i arbeid og ferdigstilling		Klarer ikke overholde tidsplanen
Sannsynlighet		8	64	
Sårbarhet		8		
Konsekvens		Forsinkelser i prosjektet som kan føre til ufullstendig resultat		
Tiltak		a) Ha en realistisk tidsplan b) Hele tiden revurdere tidsbruk og tidsplan c) Bryte ned oppgaver i mindre deler d) Revurdere vinklingen på problemstillinger og hvordan vi løser e) Revurdere oppgavens omfang		

Tabell 2.2.5

PRR - 05	Tap av prosjekt		Oppdragsgiver går konkurs, eller er tvunget til å gjøre omgående prioriteringer av ulik årsak.
Sannsynlighet	1	6	
Sårbarhet	6		

Konsekvens	Gruppen mister oppgaven og muligheten til ferdigstille utdannelsen
Tiltak	a) Modifisere oppgaven slik at eventuell intern arbeidsgiver kan overta

Tabell 2.2.6

PRR - 06		Prosjektplan		Ikke realistisk, ikke avgrenset	
Sannsynlighet		2	10		
Sårbarhet		5			
Konsekvens		Uoversiktlig plan uten rammer gir ufullstendig produkt			
Tiltak		a) Bruk tidsaspekt, gruppens kompetansesammensetning og oppgavens omfang på å sette rammer i prosjektplanen b) Oppdater og endre hyppig underveis c) Tilpass og gjør endringer i modell der det er behov for dette			

Tabell 2.2.7

PRR - 07	Sene endringer		Kravendringer fra oppdragsgiver sent i prosjektforløpet
Sannsynlighet	10	20	
Sårbarhet	2		
Konsekvens	Merarbeid, merutgifter og tidsutfordringer. Kan gi ufullstendig produkt		
Tiltak	a) Ha skriftlige føringer for dette tidlig i prosess b) Ha plan for smidige endringer/ endringshåndtering c) Bruke en tilrettelagt /egnet prosjektmodell		

Tabell 2.2.8

PRR - 08	Budsjett		Det mangler midler eller nødvendig økonomisk støtte for å kunne gjennomføre prosjektet tilfredsstillende.
Sannsynlighet	2	4	
Sårbarhet	2		
Konsekvens	Prosjektets produkt ville være mangelfullt		
Tiltak	a) Klareres skriftlig tidlig i prosessen b) Ha et fornuftig forhold til utvikling og kostnad på bachelornivå c) Gjøre god og nødvendig research d) Ha flere alternativer e) Be om vareprøver		

3 Spesifikk risikoanalyse

Oversikt risiko elektro

Tabell 3.1

Risiko ID	Scenario Elektro
ELR - 01	Feil på bestilt komponent
ELR - 02	Bestilt komponent ødelagt eller forsvinner i frakt
ELR - 03	Overspenning
ELR - 04	For lite strøm fra UR5
ELR - 05	Elektromagnetisk forstyrrelse

ELR - 06	Feil på utregning av krets
ELR - 07	Feilkobling
ELR - 08	Komponenter som ryker gjennom test
ELR - 09	Strømforsyningssvikt
ELR - 10	Feil på simulering
ELR - 11	Simuleringsprogram får feilmelding
ELR - 12	Mistet dokument på krets
ELR - 13	Feil på tegning av krets

3.1.0 Kategorisering

Tabell 3.1.1

ELR - 01		Feil på bestilt komponent		
Sannsynlighet		4	24	
Sårbarhet		6		
Konsekvens		Merarbeid, merkostnad, forsinkelser		
Tiltak		a) Kontrollere komponent ved mottakelse b) Være bevisst på valgt av leverandør		

Tabell 3.1.2

ELR - 02		Bestilling forsvinner eller ødelegges under frakt		
Sannsynlighet		4	24	
Sårbarhet		6		
Konsekvens		Tidsplan forskyves, merkostnader		
Tiltak		a) Sporing av pakke b) Kontrollere pakke ved mottakelse c) Være bevisst på valg av leverandør		

Tabell 3.1.3

ELR - 03		Overspenning		
Sannsynlighet		3	30	
Sårbarhet		10		
Konsekvens	Ødeleggelser, tidsplan forskyves, merkostnader			
Tiltak	a) Kontroll/" overspenningsvern" b) Forsikre at riktig mengde spenning/strøm føres til komponenter			

Tabell 3.1.4

ELR - 04	For lite strøm fra UR5		
Sannsynlighet	4	32	
Sårbarhet	8		
Konsekvens	Systemet virker ikke		

Tiltak	a) Ekstra strømkilde/en booster b) Bruk av servomotorer kontra DC
--------	--

Tabell 3.1.5

ELR - 05		Elektromagnetisk forstyrrelse		
Sannsynlighet		8	64	
Sårbarhet		8		
Konsekvens		Signalforstyrrelser, systemet virker ikke optimalt		
Tiltak		a) Isolering/filtrering av støy		

Tabell 3.1.6

ELR - 06		Feil på utregning av krets			
Sannsynlighet		1	5		
Sårbarhet		5			
Konsekvens		Systemet virker ikke			
Tiltak		a) Kontrollering før utnytting b) Simulering			

Tabell 3.1.7

ELR - 07	Feilkobling		
Sannsynlighet	3	30	
Sårbarhet	10		

Konsekvens	
Tiltak	a) Kontrollering før testing

Tabell 3.1.8

ELR - 08		Komponenter som ryker i test		
Sannsynlighet		3	18	
Sårbarhet		6		
Konsekvens		Merkostnad, tidsforskyvninger		
Tiltak		a) Bytte komponent b) Bestille backup på forhånd		

Tabell 3.1.9

ELR - 09		Strømsvikt		
Sannsynlighet		1	1	
Sårbarhet		1		
Konsekvens	System virker ikke			
Tiltak	a) Kontroll av kilde? b) Slå av/på sikring?			

Tabell 3.1.10

ELR - 10		Feil på simulering			
Sannsynlighet		6	18		
Sårbarhet		3			
Konsekvens		Forsinkelser			
Tiltak		a) Kontrollering			

Tabell 3.1.11

ELR - 11		Simuleringsprogram får feilmelding		
Sannsynlighet		6	6	
Sårbarhet		1		
Konsekvens	Forsinkelser			
Tiltak	a) Restart av program b) Lagre arbeid ofte			

Tabell 3.1.12

ELR - 12		Mistet dokument på krets		
Sannsynlighet		2	20	
Sårbarhet		10		
Konsekvens		Merarbeid, forsinkelser		
Tiltak		a) Jevnlig backup av dokumenter		

Tabell 3.1.13

ELR - 13		Feil på tegning av krets		
Sannsynlighet		4	16	
Sårbarhet		4		
Konsekvens	Merarbeid			
Tiltak	a) Kontrollering av krets ved simulering			

Oversikt risiko data

Tabell 3.2

Risk ID	Scenario
DAR - 01	Tidkrevende feil i kode.
DAR - 02	Mengden kompetanse som må tilegnes ila. Prosjektet er for stor.
DAR - 03	Arbeidsmengde blir for stor.
DAR - 04	Skade på prototype forsinker programmering.
DAR - 05	Ikke klarer tidsfrist.
DAR - 06	Tap av data.
DAR - 07	Hardware som ikke er kompatibel med software.
DAR - 08	Gold plating ("forgylling")

3.2.0 Kategorisering

Tabell 3.2.1

DAR - 01	Tidkrevende feil i kode		
Sannsynlighet	12	96	
Sårbarhet	8		
Konsekvens	Forsinkelser, merarbeid		
Tiltak	a) Planlegge for feilretting, debugging		

Tabell 3.2.2

DAR - 02	Mengden kompetanse som må tilegnes ila. prosjektet er for stor.		
Sannsynlighet	8	80	
Sårbarhet	10		
Konsekvens	Forskyvning tidsplan, dårlig motivasjon og dertil fremdrift. Redusert kvalitet		
Tiltak	a) Skaffe en oversikt over nødvendig kompetanse og forberede kunnskapene.		

Tabell 3.2.3

DAR - 03	Arbeidsmengde blir for stor		
Sannsynlighet	12	36	
Sårbarhet	3		

Konsekvens	Gjenstår arbeid når prosjekt avsluttes
Tiltak	a) Gjøre riktige prioriteringer. b) Bruke tid på analyse av nytteverdi.

Tabell 3.2.4

DAR - 04		Skade på prototype forsinket programmering		Som følge av for store krefter/overbelastning	
Sannsynlighet		2	20		
Sårbarhet		10			
Konsekvens		Merkostnad, merarbeid og forsinket prosjekt			
Tiltak		a) Produsere flere prototyper b) Retningslinjer i forhold til bruk av EE tilkoblet UR5			

Tabell 3.2.5

DAR - 05	Ikke klarer tidsfrist.		
Sannsynlighet	10	50	
Sårbarhet	5		
Konsekvens	Forsinkelse i prosjektet, merarbeid og merkostnad		
Tiltak	a) Sette frister for ferdigstilling med margin til kritisk deadline.		

Tabell 3.2.6

DAR - 06	Tap av data	
Sannsynlighet	1	

Sårbarhet	10	10	
Konsekvens	Merarbeid		
Tiltak	a) Ha rutiner på backup av kildekode.		

Tabell 3.2.7

DAR - 07	Hardware som ikke er kompatibel med software		
Sannsynlighet	3	15	
Sårbarhet	5		
Konsekvens	Merarbeid, forsinkelser, merkostnader		
Tiltak	a) Sørge for god kommunikasjon innad i gruppen.		

Tabell 3.2.8

DAR - 08	Gold plating		
Sannsynlighet	8	8	
Sårbarhet	1		
Konsekvens			
Tiltak	a) Tidsestimere US, og strebe etter å holde disse estimatene.		

Oversikt risiko maskin

Tabell 3.3

Risk ID	Scenario
MAR - 01	Kommunikasjon subsystemer
MAR - 02	Tap av filer i Solid Works
MAR - 03	Ufullstendige 2D tegninger
MAR - 04	Manglende forståelse ved oppbygging av modell
MAR - 05	Feil bruk av relasjoner ved modellering
MAR - 06	Feil input i FEM - analyse
MAR - 07	Unøyaktig målsetting av deler under tegning
MAR - 08	Feil materialvalg
MAR - 09	For kompleks modell
MAR - 10	Ødelagt prototype/modell under test

3.3.0 Kategorisering

Tabell 3.3.1

MAR - 01	Kommunikasjon subsystemer		Dårlig kommunikasjon mellom maskin og elektro ift styring og plassering av komponenter
Sannsynlighet	6	60	
Sårbarhet	10		
Konsekvens	Unødvendig bruk av ressurser, feil på modell, tidsforskyvning, merarbeid		
Tiltak	a) Kommunikasjon går gjennom ansvarlige for sitt felt b) Jobbe frem system sammen		

Tabell 3.3.2

MAR - 02		Tap av filer i Solid Works		Mister arbeid underveis i modellering, feil navngivning av filer, programhavari	
Sannsynlighet		7	70		
Sårbarhet		10			
Konsekvens		Ufullstendig modell, tidsforskyvning og merarbeid			
Tiltak		a) Lagre hyppig under arbeid b) Sikkerhetskopi c) Sørge for å arbeide på maskin med tilstrekkelig kapasitet			

Tabell 3.3.3

MAR - 03	Ufullstendige 2D tegninger		
Sannsynlighet	6	72	
Sårbarhet	12		
Konsekvens	Feil i modell, ikke kompatibel med subsystemer		
Tiltak	a) Være nøye med mål b) Be om veiledning		

Tabell 3.3.4

MAR - 04	Manglende forståelse ved oppbygging av modell		Tar ikke hensyn til plan eller punkter, setter på ufordelaktige tilhørigheter og mål
Sannsynlighet	2	18	
Sårbarhet	9		
Konsekvens	Usammenhengende modell, ikke brukbar modell, merarbeid, forsinkelser		

Tiltak	a) Sørge for å ha kontroll på plan og origo b) Be om veiledning tidlig i prosess
--------	---

Tabell 3.3.5

MAR - 05	Feil bruk av relasjoner ved modellering		
Sannsynlighet	4	24	
Sårbarhet	6		
Konsekvens	Tungt håndterlig modell, merarbeid, forsinkelser		
Tiltak	a) Være bevisst i forhold til relasjoner		

Tabell 3.3.6

MAR - 06	Feil input i FEM - analyse		Setter inn feil verdier ift materialstyrke og belastninger
Sannsynlighet	8	24	
Sårbarhet	3		
Konsekvens	Feil informasjon ift styrke og holdbarhet, tåler ikke planlagte belastninger, kort levetid, dyr		
Tiltak	a) Kontrollere inputs før analyse kjøres b) Være kritisk ovenfor resultater c) Be om veiledning		

Tabell 3.3.7

MAR - 07	Unøyaktig målsetting av deler under tegning		
Sannsynlighet	3	12	
Sårbarhet	4		

Konsekvens	Ikke forventet modell, inkompatibel med resten av system, merarbeid, merkostnad
Tiltak	a) Ha god kommunikasjon mellom subsystemer og innad på felt

Tabell 3.3.8

MAR - 08	Feil materialvalg		
Sannsynlighet	4	48	
Sårbarhet	12		
Konsekvens	Kostbar modell, uholdbar modell, for tung modell		
Tiltak	a) Grundig forstudie b) Be om veiledning		

Tabell 3.3.9

MAR - 09	For kompleks modell		Modellens kompleksitet stemmer ikke overens med utviklers kompetanse, eller tiltenkt bruk
Sannsynlighet	2	12	
Sårbarhet	6		
Konsekvens	For tidkrevende og dyr modell, rom for mange feil under utvikling, tidsforskyvning, merkostnad		
Tiltak	a) Være bevisst sine faglige begrensninger b) Kjenne prosjektets avgrensninger c) Søke veiledning		

Tabell 3.3.10

MAR - 10	Ødelagt prototype/modell under test		
Sannsynlighet	4		

Sårbarhet	4	16	
Konsekvens	Merarbeid, merkostnad, forsinkelser i prosjekt		
Tiltak	a) Sørge for optimalt materialvalg b) Bruke høy designfaktor for sikkerhet c) Ha planlagt to prototyper		

Tabell 3.3.11

MAR - 11	Dårlig kontakt, overflater sklir mot hverandre		Manglende eller for liten friksjon mellom kontaktflater
Sannsynlighet	2	14	
Sårbarhet	7		
Konsekvens	Ikke presis, unødvendig slitasje, merkostnad		
Tiltak	a) Kle overflate med f.eks. silikon eller gummi		

Tabell 3.3.12

MAR - 12	For stor belastning eller vibrasjoner		
Sannsynlighet	7	21	
Sårbarhet	3		
Konsekvens	Brudd / tretthetsbrudd, merkostnad, ikke presis.		
Tiltak	a) Endre vinkler på utstyr. b) Kontrollere innfestninger.		

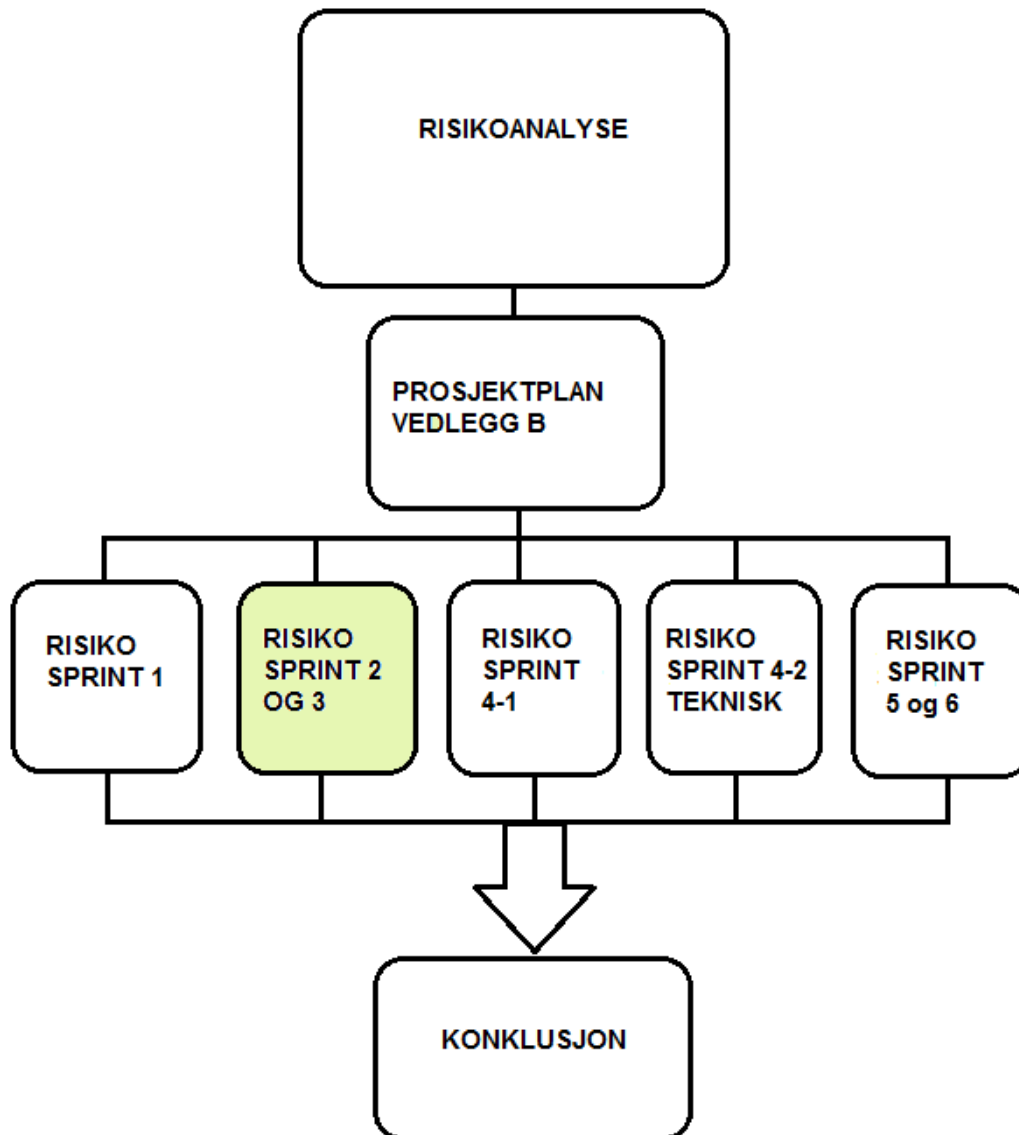
Tabell 3.3.13

MAR - 13	Brudd på deler		
-----------------	-----------------------	--	--

Sannsynlighet	10	30	
Sårbarhet	3		
Konsekvens	Tidsforskyvning i prosjekt, merarbeid og merkostnad.		
Tiltak	a) Forsterke konstruksjon. b) Endre vinkel. c) Vurdere materialvalg		

Tabell 3.3.14

MAR - 14		Treghet/stor motstand pga friksjon i ledd		
Sannsynlighet		5	15	
Sårbarhet		3		
Konsekvens		Ikke ønsket funksjon/effekt, defekte deler/system.		
Tiltak		a) Sette inn lager. b) Større pasninger c) Velge konsept som eliminerer dette		



Risiko sprint 2 og 3

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAUGSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Risiko sprint 2 og 3
VERSJON / DATO	2.0 / 15. Mars 2016
FORFATTER	Aina Engen Nilsen



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



2 Overordnet risikoanalyse

Personorienterte suksessfaktorer

Tabell 2

Risiko ID	Scenario	Konsekvens
PER - 01	Sykdom/ulykke	Gruppemedlem må trekke seg underveis.
PER - 02	Dårlig Kommunikasjon	Misforståelser internt og eksternt. Vanskeligheter med å forklare / ordlegge seg.
PER - 04	Interne konflikter	Uenighet i gruppen skaper splid.
PER - 05	Manglende prosjektstyring	Nøkkelperson(er) for prosjektstyring uteblir
PER - 06	Overbelastning/ Arbeidsmengde	Skjevfordeling arbeidsmengde og individuell grenser for stress

2.1.0 Kategorisering

Tabell 3

PER - 01	Sykdom/ulykke		Gruppemedlem må trekke seg underveis
Sannsynlighet	7	42	
Sårbarhet	6		
Konsekvens	Gruppen må fordele arbeid seg i mellom, og kompensere for eventuelle faglige mangler.		
Tiltak	<div>a) Redusere oppgavens omfang tilsvarende ressurstap</div> <div>b) Være meget bevisst arbeidsfordeling, og ta medmenneskelige hensyn</div> <div>c) Sørge for en god og åpen dialog rundt temaet hele veien</div> <div>d) Sørge for å ha tilgang til hverandres arbeid i form av felles plattform</div>		

Tabell 4

PER - 02		Dårlig kommunikasjon		Misforståelser internt og eksternt. Vanskeligheter med å forklare/ordlegge seg.
Sannsynlighet		6	60	
Sårbarhet		10		
Konsekvens		Fører til alt fra frustrasjon til feiltolkninger, og kan lede til store feil og mangler.		
Tiltak		a) Bli enige på forhånd hvordan kommunikasjon skal foregå, og følge dette. b) Skriftlige avtaler og godkjenninger av dokumenter c) Økt fokus på kommunikasjon i form av medarbeidersamtaler d) Samtaleteknikk (bevisst forhold til avsender - mottaker, tydelighet) e) Gruppedynamikk og lav terskel for diskusjoner		

Tabell 5

PER - 04		Interne konflikter		Uenighet som fører til splid i gruppen	
Sannsynlighet		3	24		
Sårbarhet		8			
Konsekvens		Forsinkelser i prosjekt, frafall av medlemmer			
Tiltak		a) Ha føringer på hvordan vi tar opp et problem. b) Ved store uenigheter som ikke lar seg løse internt, bør det hentes inn en utenforstående (eks. veileder) for bistand, eventuelt beslutning. c) Gruppen skal ha en intern arbeidskontrakt. d) Fokus på kommunikasjon i form av medarbeidersamtaler e) Gruppedynamikk og lav terskel for diskusjoner			

Tabell 6

PER - 05	Manglende prosjektstyring		Nøkkelperson(er) for prosjektstyring uteblir
Sannsynlighet	6	42	
Sårbarhet	7		
Konsekvens	Forsinkelser i prosjekt, ikke fullstendig produkt/feil produkt		
Tiltak	a) Velge en prosjektmodell som gir god oversikt for alle, samt har fokus på kollektiv fremdrift. b) Ha to personer med hovedansvar for fremdrift og oversikt.		

Tabell 7

PER - 06	Overbelastning/ Arbeidsmengde		Skjevfordeling arbeidsmengde og individuell grenser for stress
Sannsynlighet	7	49	
Sårbarhet	7		
Konsekvens	Unødvendig frustrasjon som kan føre til fravær eller frafall i gruppen. Ressurstap og forsinkelser		
Tiltak	a) Være bevisst egne begrensninger og tidlige tegn på overarbeidelse b) Være bevisst på og interessert i medstudenters ve og vel c) Forsøker fordele arbeid så jevnt det lar seg gjøre, og ta hensyn der det er nødvendig d) Være flinke til å si i fra så tidlig som mulig		

Prosjektrelaterte suksessfaktorer

Tabell 8

Risiko ID	Scenario	Konsekvens
PRR - 03	Tekniske problemer	Tap av filer, programvare, tekniske feil sub - disipliner, etc.
PRR - 04	Forsinkelser i arbeid og ferdigstilling	Klarer ikke overholde tidsplanen
PRR - 07	Sene endringer	Kravendringer fra oppdragsgiver sent i prosjektforløpet
PRR - 09	Avhengigheter mellom disipliner	Vi klarer ikke avdekke vesentlige avhengigheter mellom disiplinene tidlig nok, og blir avventende i forhold til hverandre
PRR - 10	Ikke tilstrekkelig teknisk kommunikasjon mellom disipliner	For lite tydelig kommunikasjon og felles planlegging mellom disipliner som har direkteavhengighet i form av grensesnitt
PRR - 11	Mangel på tid fører til lite gjennomtenkte løsninger	Fare for at ambisjonsnivået kombinert med tidsbegrensning leder til forhastede og lite gjennomtenkte slutninger

2.2.0 Kategorisering

Tabell 9

PRR - 03		Tekniske problemer		Tap av filer, programvare, tekniske feil i sub - disipliner, etc.
Sannsynlighet		6	76	
Sårbarhet		12		
Konsekvens		Merarbeid, forskyvning i tidsplanen		
Tiltak		a) Ha gode rutiner for backup, lagring og historikk		

Tabell 10

PRR - 04		Forsinkelser i arbeid og ferdigstilling		Klarer ikke overholde tidsplanen	
Sannsynlighet		8	64		
Sårbarhet		8			
Konsekvens		Forsinkelser i prosjektet som kan føre til ufullstendig resultat			
Tiltak		a) Ha en realistisk tidsplan b) Hele tiden revurdere tidsbruk og tidsplan c) Bryte ned oppgaver i mindre deler d) Revurdere vinklingen på problemstillinger og hvordan vi løser			

Tabell 11

PRR - 07		Sene endringer		Kravendringer fra oppdragsgiver sent i prosjektførsløpet	
Sannsynlighet		10	20		
Sårbarhet		2			
Konsekvens		Merarbeid, merutgifter og tidsutfordringer. Kan gi ufullstendig produkt			
Tiltak		a) Ha skriftlige føringer for dette tidlig i prosess b) Ha plan for smidige endringer/ endringshåndtering c) Bruke en tilrettelagt /egnet prosjektmodell			

Tabell 12

PRR - 09	Avhengighet mellom disipliner		Vi klarer ikke avdekke vesentlige avhengigheter mellom disiplinene tidlig nok, og blir avventende i forhold til hverandre
Sannsynlighet	10	80	
Sårbarhet	8		
Konsekvens	Frustrasjon, forsinkelse i prosjekt, merkostnad		
Tiltak	a) Avklare på forhånd flest mulig overordnede (forventede) avhengigheter b) Være tydelige i vår kommunikasjon om hva som må foreligge i hvilken rekkefølge under egen disiplin c) Vise interesse for og tilegne seg nødvendig kunnskap om de andre disipliners områder i prosjektet		

Tabell 13

PRR - 10	Manglene forklarende teknisk kommunikasjon mellom disipliner	For lite tydelig kommunikasjon og felles planlegging mellom disipliner rundt tekniske avhengigheter (interfaces) av hverandre	
Sannsynlighet	8	48	
Sårbarhet	6		
Konsekvens	Feil og mangler som følge av vesentlige misforståelser, forsinkelser og frustrasjon		
Tiltak	a) Ha føringer for god og tydelig kommunikasjon b) Legge til rette for planlegging sammen c) Være bevisst sitt tekniske språk og forklaringer opp mot medstudenter d) Skriftlige avtaler på løsninger		

Tabell 14

PRR - 11		Mangel på tid fører til lite gjennomtenkte løsninger		Fare for at ambisjonsnivået kombinert med tidsbegrensning leder til forhastede og lite gjennomtenkte beslutninger
Sannsynlighet		7	49	
Sårbarhet		7		
Konsekvens		Vesentlige feil og mangler, ufullstendig produkt. Forsinkelser og merarbeid/kostnader		
Tiltak		a) Forsøker å ha en realistisk forestilling om innhold i oppgaven, samt tidsbruk b) Ha føringer for hvordan vi best velger løsninger c) Holde korte møter ved alle veivalg, tydelig kommunikasjonen mellom fagretningene for å forsikre oss om at vi sitter med samme oppfatning.		

3 Spesifikk risikoanalyse

Oversikt risiko elektro

Tabell 15

Risiko ID	Scenario Elektro
ELR - 01	Feil på bestilt komponent
ELR - 02	Bestilt komponent ødelagt eller forsvinner i frakt
ELR - 03	Overspenning
ELR - 04	For lite strøm fra UR5
ELR - 05	Elektromagnetisk forstyrrelse

ELR - 07	Feilkobling
ELR - 08	Komponenter som ryker gjennom test

3.1.0 Kategorisering

Tabell 16

ELR - 01		Feil på bestilt komponent		
Sannsynlighet		4	40	
Sårbarhet		10		
Konsekvens		Merarbeid, merkostnad, forsinkelser		
Tiltak		a) Kontrollere komponent ved mottakelse b) Være bevisst på valgt av leverandør		

Tabell 17

ELR - 02		Bestilling forsvinner eller ødelegges under frakt		
Sannsynlighet		4	40	
Sårbarhet		10		
Konsekvens		Tidsplan forskyves, merkostnader		
Tiltak		a) Sporing av pakke b) Kontrollere pakke ved mottakelse c) Være bevisst på valg av leverandør		

Tabell 18

ELR - 03		Overspenning		
Sannsynlighet		3	30	
Sårbarhet		10		
Konsekvens		Ødeleggelser, tidsplan forskyves, merknader		
Tiltak		a) Kontroll/" overspenningsvern" b) Forsikre at riktig mengde spenning/strøm føres til komponenter		

Tabell 19

ELR - 04	For lite strøm fra UR5		
Sannsynlighet	4	32	
Sårbarhet	8		
Konsekvens	Systemet virker ikke		
Tiltak	a) Ekstra strømkilde/en booster b) Bruk av servomotorer kontra DC		

Tabell 20

ELR - 05	Elektromagnetisk forstyrrelse		
Sannsynlighet	8	64	
Sårbarhet	8		
Konsekvens	Signalforstyrrelser, systemet virker ikke optimalt		

Tiltak	a) Isolering/filtrering av støy
--------	---------------------------------

Tabell 21

ELR - 07	Feilkobling		
Sannsynlighet	3	30	
Sårbarhet	10		
Konsekvens			
Tiltak	a) Kontrollering før testing		

Tabell 22

ELR - 08		Komponenter som ryker i test		
Sannsynlighet		5	30	
Sårbarhet		6		
Konsekvens		Merkostnad, tidsforskyvninger		
Tiltak		a) Bestille backup på forhånd b) Sørge for sikkert testmiljø/område c) Kontroll på inngangsverdier		

Oversikt risiko data

Tabell 23

Risk ID	Scenario
DAR - 01	Tidkrevende feil i kode.
DAR - 02	Mengden kompetanse som må tilegnes ila. prosjektet er for stor.

DAR - 05	Ikke klarer tidsfrist.
-----------------	------------------------

3.2.0 Kategorisering

Tabell 24

DAR - 01	Tidkrevende feil i kode		
Sannsynlighet	10	80	
Sårbarhet	8		
Konsekvens	Forsinkelser, merarbeid		
Tiltak	a) Planlegge for feilretting, debugging		

Tabell 25

DAR - 02		Mengden kompetanse som må tilegnes ila. prosjektet er for stor.		
Sannsynlighet		6	60	
Sårbarhet		10		
Konsekvens		Forskyvning tidsplan, dårlig motivasjon og dertil fremdrift. Redusert kvalitet		
Tiltak		a) Skaffe en oversikt over nødvendig kompetanse og forberede kunnskapene.		

Tabell 26

DAR - 05	Klarer ikke tidsfrist.	
-----------------	-------------------------------	--

Sannsynlighet	10	50	
Sårbarhet	5		
Konsekvens	Forsinkelse i prosjektet, merarbeid og merkostnad		
Tiltak	a) Sette frister for ferdigstilling med margin til kritisk deadline. b) Prioriterer realistisk slik at de viktigste funksjonene blir fullført.		

Oversikt risiko maskin

Tabell 27

Risk ID	Scenario
MAR - 12	For stor belastning
MAR - 15	Problemer med gjennomføring av simulering
MAR - 18	Avhengigheter av andres arbeid for egen fremdrift

3.3.0 Kategorisering

Tabell 28

MAR - 12	For stor belastning på prototype og utstyr		
Sannsynlighet	7	42	
Sårbarhet	6		
Konsekvens	Brudd / tretthetsbrudd, merkostnad, ikke presis.		
Tiltak	a) Endre vinkler på utstyr. b) Kontrollere innfestninger. c) Kontrollere inngangsverdier d) Testing		

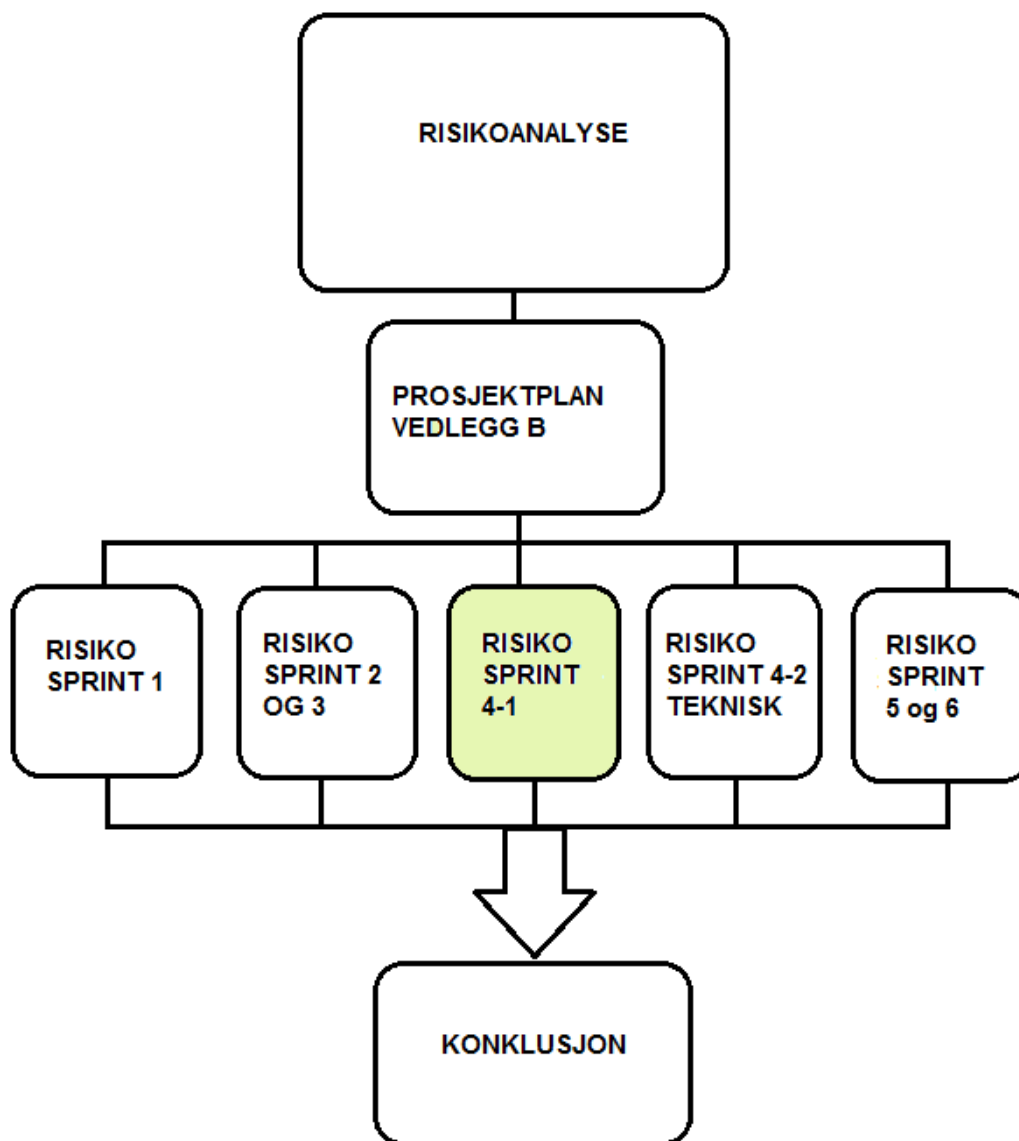
	e) Dimensjonering
--	-------------------

Tabell 29

MAR - 15		Problemer med gjennomføring av simulering		Vanskeligheter med gjennomføring av simulering slik at vi ikke får verifikasjonen vi trenger	
Sannsynlighet		8	80		
Sårbarhet		10			
Konsekvens		Usikkerhet rundt mål og funksjonalitet, kan føre til unødvendig tids - og ressursbruk			
Tiltak		a) Forenkle modell så godt det lar seg gjøre før simulering b) Be om veiledning c) Begrense kompleksitet og omfang av simulering			

Tabell 30

MAR - 18		Avhengighet av andres arbeid for egen fremdrift		At filer ikke blir delt tidsnok, arbeid ikke utføres som avtalt eller etter planen.	
Sannsynlighet		10	60		
Sårbarhet		6			
Konsekvens		Forsinkelser i fremdrift og frustrasjon. Merarbeid og skjev arbeidsfordeling			
Tiltak		a) Være tydelige i kommunikasjon om hva som er nødvendig for videre fremdrift b) Sette tidsfrist på spesifikke oppgaver og holde disse c) Forsøke å fordele arbeidsmengde best mulig, både med tanke på kapasitet og kompetanse/ forutsetning for tilegning av kompetanse			



Risikoanalyse sprint 4-1

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAKSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Risikoanalyse sprint 4-1
VERSJON / DATO	1.4 / 14. April 2016
FORFATTER	Aina Engen Nilsen



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



2 OVERORDNET RISIKOANALYSE

Personorienterte suksessfaktorer

Tabell 2.1

Risiko ID	Scenario	Konsekvens
PER - 01	Sykdom/ulykke	Gruppemedlem må trekke seg underveis.
PER - 02	Dårlig Kommunikasjon	Misforståelser internt og eksternt. Vanskeligheter med å forklare / ordlegge seg.
PER - 04	Interne konflikter	Uenighet i gruppen skaper splid.
PER - 05	Manglende prosjektstyring	Nøkkelperson(er) for prosjektstyring uteblir
PER - 06	Overbelastning/ Arbeidsmengde	Skjevfordeling arbeidsmengde og individuell grenser for stress

2.1.0 Kategorisering

Tabell 2.1.1

PER - 01	Sykdom/ulykke		Gruppemedlem må trekke seg underveis
Sannsynlighet	7	42	
Sårbarhet	6		
Konsekvens	Gruppen må fordele arbeid seg i mellom, og kompensere for eventuelle faglige mangler.		
Tiltak	<div>a) Redusere oppgavens omfang tilsvarende ressurstap</div> <div>b) Være meget bevisst arbeidsfordeling, og ta medmenneskelige hensyn</div> <div>c) Sørge for en god og åpen dialog rundt temaet hele veien</div> <div>d) Sørge for å ha tilgang til hverandres arbeid i form av felles plattform</div>		

Tabell 2.1.2

PER - 02	Dårlig kommunikasjon		Misforståelser internt og eksternt. Vanskeligheter med å forklare/ordlegge seg.
Sannsynlighet	6	60	
Sårbarhet	10		
Konsekvens	Fører til alt fra frustrasjon til feiltolkninger, og kan lede til store feil og mangler.		
Tiltak	a) Bli enige på forhånd hvordan kommunikasjon skal foregå, og følge dette. b) Skriftlige avtaler og godkjenninger av dokumenter c) Økt fokus på kommunikasjon i form av medarbeidersamtaler d) Samtaleteknikk (bevisst forhold til avsender - mottaker, tydelighet) e) Gruppedynamikk og lav terskel for diskusjoner		

Tabell 2.1.4

PER - 04		Interne konflikter		Uenighet som fører til splid i gruppen	
Sannsynlighet		3	24		
Sårbarhet		8			
Konsekvens		Forsinkelser i prosjekt, frafall av medlemmer			
Tiltak		a) Ha føringer på hvordan vi tar opp et problem. b) Ved store uenigheter som ikke lar seg løse internt, bør det hentes inn. En utenforstående (eks. veileder) for bistand, eventuelt beslutning. c) Gruppen skal ha en intern arbeidskontrakt. d) Fokus på kommunikasjon i form av medarbeidersamtaler e) Gruppedynamikk og lav terskel for diskusjoner			

Tabell 2.1.5

PER - 05		Manglende prosjektstyring		Nøkkelperson(er) for prosjektstyring uteblir
Sannsynlighet		6	42	
Sårbarhet		7		
Konsekvens	Forsinkelser i prosjekt, ikke fullstendig produkt/feil produkt			
Tiltak	a) Velge en prosjektmodell som gir god oversikt for alle, samt har fokus på kollektiv fremdrift. b) Ha to personer med hovedansvar for fremdrift og oversikt.			

Tabell 2.1.6

PER - 06		Overbelastning/ Arbeidsmengde		Skjevfordeling arbeidsmengde og individuell grenser for stress	
Sannsynlighet		7	49		
Sårbarhet		7			
Konsekvens		Unødvendig frustrasjon som kan føre til fravær eller frafall i gruppen. Ressurstap og forsinkelser			
Tiltak		a) Være bevisst egne begrensninger og tidlige tegn på overarbeidelse b) Være bevisst på og interessert i medstudenters ve og vel c) Forsøker fordele arbeid så jevnt det lar seg gjøre, og ta hensyn der det er nødvendig d) Være flinke til å si i fra så tidlig som mulig			

Prosjektorienterte suksessfaktorer

Tabell 2.2

Risiko ID	Scenario	Konsekvens
PRR - 03	Tekniske problemer	Tap av filer, programvare, tekniske feil sub - disipliner, etc.
PRR - 04	Forsinkelser i arbeid og ferdigstilling	Klarer ikke overholde tidsplanen
PRR - 07	Sene endringer	Kravendringer fra oppdragsgiver sent i prosjektforløpet
PRR - 09	Avhengigheter mellom disipliner	Vi klarer ikke avdekke vesentlige avhengigheter mellom disiplinene tidlig nok, og blir avventende i forhold til hverandre
PRR - 10	Ikke tilstrekkelig teknisk kommunikasjon mellom disipliner	For lite tydelig kommunikasjon og felles planlegging mellom disipliner som har direkteavhengighet i form av grensesnitt
PRR - 11	Mangel på tid fører til lite gjennomtenkte løsninger	Fare for at ambisjonsnivået kombinert med tidsbegrensning leder til forhastede og lite gjennomtenkte slutninger
PRR - 12	Fokus utenfor User Stories definert i sprint	Fare for at ikke oppgavene blir fullført innen tidsfrist og dernest forsinkelser i prosjektet og/eller ufullstendig produkt.

2.2.0 Kategorisering

Tabell 2.2.3

PRR - 03	Tekniske problemer		Tap av filer, programvare, tekniske feil i sub - disipliner, etc.
Sannsynlighet	6	76	
Sårbarhet	12		

Konsekvens	Merarbeid, forskyvning i tidsplanen
Tiltak	a) Ha gode rutiner for backup, lagring og historikk

Tabell 2.2.4

PRR - 04		Forsinkelser i arbeid og ferdigstilling		Klarer ikke overholde tidsplanen	
Sannsynlighet		8	64		
Sårbarhet		8			
Konsekvens		Forsinkelser i prosjektet som kan føre til ufullstendig resultat			
Tiltak		a) Ha en realistisk tidsplan b) Hele tiden revurdere tidsbruk og tidsplan c) Bryte ned oppgaver i mindre deler d) Revurdere vinklingen på problemstillinger og hvordan vi løser			

Tabell 2.2.7

PRR - 07	Sene endringer		Kravendringer fra oppdragsgiver sent i prosjektførsløpet
Sannsynlighet	5	40	
Sårbarhet	8		
Konsekvens	Merarbeid, merutgifter og tidsutfordringer. Kan gi ufullstendig produkt		
Tiltak	a) Ha skriftlige føringer for dette tidlig i prosess b) Ha plan for smidige endringer/ endringshåndtering c) Bruke en tilrettelagt /egnet prosjektmodell		

Tabell 2.2.9

PRR - 09	Avhengighet mellom disipliner		Vi klarer ikke avdekke vesentlige avhengigheter mellom disiplinene tidlig nok, og blir avventende i forhold til hverandre	
Sannsynlighet	10	100		
Sårbarhet	10			
Konsekvens	Frustrasjon, forsinkelse i prosjekt, merkostnad			
Tiltak	a) Avklare på forhånd flest mulig overordnede (forventede) avhengigheter b) Være tydelige i vår kommunikasjon om hva som må foreligge i hvilken rekkefølge under egen disiplin c) Vise interesse for og tilegne seg nødvendig kunnskap om de andre disipliners områder i prosjektet			

Tabell 2.2.10

PRR - 10	Manglene forklarende teknisk kommunikasjon mellom disipliner		For lite tydelig kommunikasjon og felles planlegging mellom disipliner rundt tekniske avhengigheter (interfaces) av hverandre
Sannsynlighet	7	42	
Sårbarhet	6		
Konsekvens	Feil og mangler som følge av vesentlige misforståelser, forsinkelser og frustrasjon		
Tiltak	a) Ha føringer for god og tydelig kommunikasjon b) Legge til rette for planlegging sammen c) Være bevisst sitt tekniske språk og forklaringer opp mot medstudenter d) Skriftlige avtaler på løsninger		

Tabell 2.2.11

PRR - 11	Mangel på tid fører til lite gjennomtenkte løsninger		Fare for at ambisjonsnivået kombinert med tidsbegrensning leder til forhastede og lite gjennomtenkte beslutninger
Sannsynlighet	7	49	
Sårbarhet	7		
Konsekvens	Vesentlige feil og mangler, ufullstendig produkt. Forsinkelser og merarbeid/kostnader		
Tiltak	a) Forsøker å ha en realistisk forestilling om innhold i oppgaven, samt tidsbruk b) Ha føringer for hvordan vi best velger løsninger c) Holde korte møter ved alle veivalg, tydelig kommunikasjonen mellom fagretningene for å forsikre oss om at vi sitter med samme oppfatning.		

Tabell 2.2.12

PRR - 12		Fokus utenfor User Stories definert i sprint		Fare for at ikke oppgavene blir fullført innen tidsfrist og dernest forsinkelser i prosjektet og/eller ufullstendig produkt.
Sannsynlighet		7	49	
Sårbarhet		7		
Konsekvens		Ufullstendig produkt, forsinkelser og frustrasjon		
Tiltak		a) Daily scrum b) Eierskap til US i sprint c) Kommunikasjon og tydelig ledelse.		

3 Spesifikk risikoanalyse

Oversikt risiko elektro

Tabell 3.1

Risiko ID	Scenario Elektro
ELR - 01	Feil på bestilt komponent
ELR - 02	Bestilt komponent ødelagt eller forsvinner i frakt
ELR - 03	Overspenning
ELR - 05	Elektromagnetisk forstyrrelse
ELR - 07	Feilkobling
ELR - 08	Komponenter som ryker gjennom test

3.1.0 Kategorisering

Tabell 3.1.1

ELR - 01		Feil på bestilt komponent		
Sannsynlighet		4	40	
Sårbarhet		10		
Konsekvens		Merarbeid, merkostnad, forsinkelser		
Tiltak		a) Kontrollere komponent ved mottakelse b) Være bevisst på valgt av leverandør		

Tabell 3.1.2

ELR - 02		Bestilling forsvinner eller ødelegges under frakt		
Sannsynlighet		4	40	
Sårbarhet		10		
Konsekvens		Tidsplan forskyves, merkostnader		
Tiltak		a) Sporing av pakke b) Kontrollere pakke ved mottakelse c) Være bevisst på valg av leverandør		

Tabell 3.1.3

ELR - 03		Overspenning		
Sannsynlighet		3	30	
Sårbarhet		10		
Konsekvens	Ødeleggelser, tidsplan forskyves, merkostnader			
Tiltak	a) Kontroll/" overspenningsvern" b) Forsikre at riktig mengde spenning/strøm føres til komponenter			

Tabell 3.1.5

ELR - 05		Elektromagnetisk forstyrrelse		
Sannsynlighet		3	24	
Sårbarhet		8		
Konsekvens	Signalforstyrrelser, systemet virker ikke optimalt			

Tiltak	a) Isolering/filtrering av støy
--------	---------------------------------

Tabell 3.1.7

ELR - 07	Feilkobling		
Sannsynlighet	5	50	
Sårbarhet	10		
Konsekvens			
Tiltak	a) Kontrollering før testing		

Tabell 3.1.8

ELR - 08		Komponenter som ryker i test		
Sannsynlighet		3	30	
Sårbarhet		10		
Konsekvens		Merkostnad, tidsforskyvninger		
Tiltak		a) Bestille backup på forhånd b) Sørge for sikkert testmiljø/område c) Kontroll på inngangsverdier		

Oversikt risiko data

Tabell 3.2

Risk ID	Scenario
DAR - 01	Tidkrevende feil i kode.
DAR - 02	Mengden kompetanse som må tilegnes ila. prosjektet er for stor.

DAR - 05

Ikke klarer tidsfrist.

3.2.0 Kategorisering

Tabell 3.2.1

DAR - 01	Tidkrevende feil i kode		
Sannsynlighet	7	56	
Sårbarhet	8		
Konsekvens	Forsinkelser, merarbeid		
Tiltak	a) Planlegge for feilretting, debugging		

Tabell 3.2.2

DAR - 02	Mengden kompetanse som må tilegnes ila. prosjektet er for stor.		
Sannsynlighet	6	60	
Sårbarhet	10		
Konsekvens	Forskyvning tidsplan, dårlig motivasjon og dertil fremdrift. Redusert kvalitet		
Tiltak	a) Skaffe en oversikt over nødvendig kompetanse og forberede kunnskapene.		

Tabell 3.2.5

DAR - 05	Klarer ikke tidsfrist.		
Sannsynlighet	10	50	
Sårbarhet	5		
Konsekvens	Forsinkelse i prosjektet, merarbeid og merkostnad		
Tiltak	a) Sette frister for ferdigstilling med margin til kritisk deadline. b) Prioriterer realistisk slik at de viktigste funksjonene blir fullført.		

Oversikt risiko maskin

Tabell 3.3

Risk ID	Scenario
MAR - 12	For stor belastning
MAR - 15	Problemer med gjennomføring av simulering
MAR - 19	3D - printer utilgjengelig/ute av drift
MAR - 20	Filer for print ikke egnet
MAR - 21	Overdimensjonering/vekt

3.3.0 Kategorisering

Tabell 3.3.12

MAR - 12	For stor belastning på prototype og utstyr		
Sannsynlighet	7	42	
Sårbarhet	6		
Konsekvens	Brudd / tretthetsbrudd, merkostnad, ikke presis.		
Tiltak	a) Endre vinkler på utstyr. b) Kontrollere innfestninger. c) Kontrollere inngangsverdier d) Testing e) Dimensjonering		

Tabell 3.3.15

MAR - 15	Problemer med gjennomføring av simulering		Vanskeligheter med gjennomføring av simulering slik at vi ikke får verifikasjonen vi trenger
Sannsynlighet	8	80	
Sårbarhet	10		
Konsekvens	Usikkerhet rundt mål og funksjonalitet, kan føre til unødvendig tids - og ressursbruk		
Tiltak	a) Forenkle modell så godt det lar seg gjøre før simulering b) Be om veiledning c) Begrense kompleksitet og omfang av simulering		

Tabell 3.3.19

MAR - 19	3D - printer utilgjengelig/ute av drift		Vi får ikke printet deler til prototype når vi trenger/planlegger for dette i sprint
Sannsynlighet	10		

Sårbarhet	6	60	
Konsekvens	Ufullstendig produkt, forsinkelser		
Tiltak	a) Sørge for å ha flere bein (les: Printere) og stå på b) God planlegging		

Tabell 3.3.20

MAR - 20	Filer for print ikke egnet		Vi har 3D - filer hvor modelloppbygging ikke er tilpasset for print slik at printer vil få problemer eller print ikke kan kjøres.
Sannsynlighet	8	48	
Sårbarhet	8		
Konsekvens	Merarbeid, forsinkelser		
Tiltak	a) Forsikre oss om oppbygging i forhold til print i god tid. b) Be om veiledning i god tid.		

Tabell 3.3.21

MAR - 21	Overdimensjonerin g/vekt		Modell er overdimensjonert og bidrar til uønsket vekt på prototype (maks 5 kg)
Sannsynlighet	8	48	
Sårbarhet	8		
Konsekvens	Ufullstendig produkt, merarbeid og forsinkelser		
Tiltak	a) Ha fokus på vekt gjennom hele designprosessen. b) 'Kontrollveie' modell underveis (teoretisk)		

Oversikt risiko grensesnitt

Tabell 3.4

Risk ID	Scenario
GRR - 01	Inkompatible grensesnitt EE prototype 1
GRR - 02	Kommunikasjonssvikt mellom ulike grensesnitt
GRR - 03	Uforutsett interferens og/eller for høy friksjon under fysisk test

3.4.0 Kategorisering

Tabell 3.4.1

GRR - 01		Inkompatible grensesnitt EE prototype 1		
Sannsynlighet		10	100	
Sårbarhet		10		
Konsekvens				
Tiltak		a) Være bevisst målsetting b) God kommunikasjon mellom disiplinene c) Gjøre beregninger		

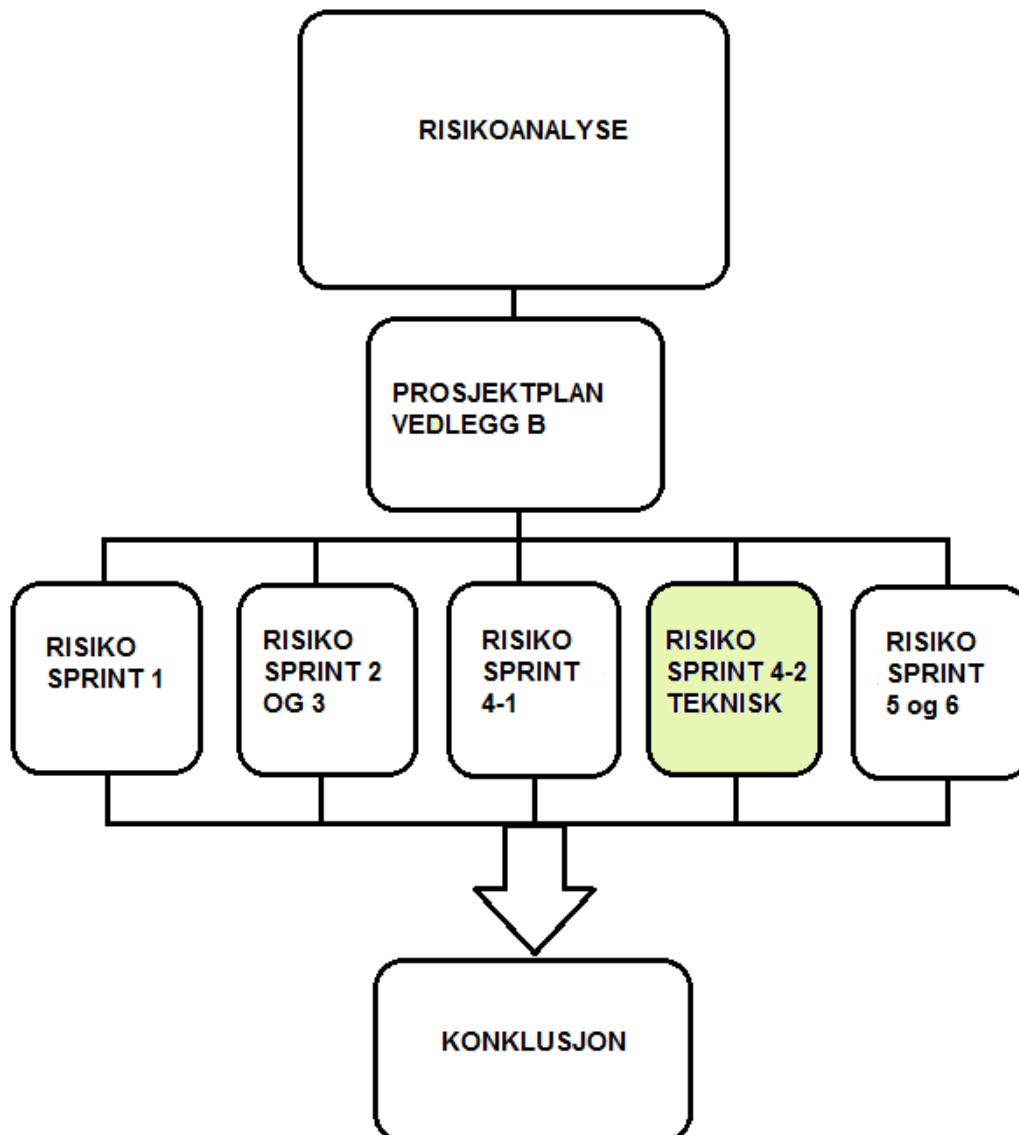
Tabell 3.4.2

GRR - 02			
Sannsynlighet	10	100	
Sårbarhet	10		

Konsekvens	
Tiltak	a) Kjøre tester b) Eksternt styresystem

Tabell 3.4.3

GRR - 03			
Sannsynlighet			
Sårbarhet			
Konsekvens			
Tiltak	a) Være bevisst målsetting. b) Beregne klaringer. c) Studere sammenstilling/visuell teoretisk testing. d) Smøring.		



Risikoanalyse sprint 4-2

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAKSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Risikoanalyse sprint 4-2
VERSJON / DATO	1.5 / 28. April 2016
FORFATTER	Aina Engen Nilsen



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



2 OVERORDNET RISIKOANALYSE

Personorienterte suksessfaktorer

Tabell 2.1

Risiko ID	Scenario	Konsekvens
PER - 01	Sykdom/ulykke	Gruppemedlem må trekke seg underveis.
PER - 02	Dårlig Kommunikasjon	Misforståelser internt og eksternt. Vanskeligheter med å forklare / ordlegge seg.
PER - 04	Interne konflikter	Uenighet i gruppen skaper splid.
PER - 05	Manglende prosjektstyring	Nøkkelperson(er) for prosjektstyring uteblir
PER - 06	Overbelastning/ Arbeidsmengde	Skjevfordeling arbeidsmengde og individuell grenser for stress
PER - 07	Konflikt mellom gruppe og andre involverte parter	Dette tar fokus og kan forhindre gruppen i å utføre sin oppgave på en tilfredsstillende måte

2.1.0 Kategorisering

Tabell 2.1.1

PER - 01	Sykdom/ulykke		Gruppemedlem må trekke seg underveis
Sannsynlighet	7	42	
Sårbarhet	6		
Konsekvens	Gruppen må fordele arbeid seg i mellom, og kompensere for eventuelle faglige mangler.		
Tiltak	<div>a) Redusere oppgavens omfang tilsvarende ressurstap</div> <div>b) Være meget bevisst arbeidsfordeling, og ta medmenneskelige hensyn</div> <div>c) Sørge for en god og åpen dialog rundt temaet hele veien</div> <div>d) Sørge for å ha tilgang til hverandres arbeid i form av felles plattform</div>		

Tabell 2.1.2

PER - 02	Dårlig kommunikasjon		Misforståelser internt og eksternt. Vanskeligheter med å forklare/ordlegge seg.
Sannsynlighet	6	60	
Sårbarhet	10		
Konsekvens	Fører til alt fra frustrasjon til feiltolkninger, og kan lede til store feil og mangler.		
Tiltak	a) Bli enige på forhånd hvordan kommunikasjon skal foregå, og følge dette. b) Skriftlige avtaler og godkjenninger av dokumenter c) Økt fokus på kommunikasjon i form av medarbeidersamtaler d) Samtaleteknikk (bevisst forhold til avsender - mottaker, tydelighet) e) Gruppedynamikk og lav terskel for diskusjoner		

Tabell 2.1.4

PER - 04		Interne konflikter		Uenighet som fører til splid i gruppen	
Sannsynlighet		3	24		
Sårbarhet		8			
Konsekvens		Forsinkelser i prosjekt, frafall av medlemmer			
Tiltak		a) Ha føringer på hvordan vi tar opp et problem. b) Ved store uenigheter som ikke lar seg løse internt, bør det hentes inn. En utenforstående (eks. veileder) for bistand, eventuelt beslutning. c) Gruppen skal ha en intern arbeidskontrakt. d) Fokus på kommunikasjon i form av medarbeidersamtaler e) Gruppedynamikk og lav terskel for diskusjoner			

Tabell 2.1.5

PER - 05		Manglende prosjektstyring		Nøkkelperson(er) for prosjektstyring uteblir
Sannsynlighet		6	42	
Sårbarhet		7		
Konsekvens		Forsinkelser i prosjekt, ikke fullstendig produkt/feil produkt		
Tiltak		a) Velge en prosjektmodell som gir god oversikt for alle, samt har fokus på kollektiv fremdrift. b) Ha to personer med hovedansvar for fremdrift og oversikt.		

Tabell 2.1.6

PER - 06		Overbelastning/ Arbeidsmengde	Skjevfordeling arbeidsmengde og individuell grenser for stress
Sannsynlighet		7	
Sårbarhet		7	
		49	
Konsekvens	Unødvendig frustrasjon som kan føre til fravær eller frafall i gruppen. Ressurstap og forsinkelser		
Tiltak	a) Være bevisst egne begrensninger og tidlige tegn på overarbeidelse b) Være bevisst på og interessert i medstudenters ve og vel c) Forsøker fordele arbeid så jevnt det lar seg gjøre, og ta hensyn der det er nødvendig d) Være flinke til å si i fra så tidlig som mulig		

Tabell 2.1.7

PER - 07	Konflikt mellom gruppe og involverte parter		Forhindrer gruppen fra å holde fokus på oppgaven. Generer usikkerhet og stress, samt er mentalt belastende
Sannsynlighet	8	80	
Sårbarhet	10		
Konsekvens	Feil fokus, forhindrer fremdrift, dårligere resultat		

Tiltak	a) Følge prosedyrer og retningslinjer for håndtering av personalsaker og personvern (ledelse) b) God og riktig kommunikasjon
--------	---

Prosjektorienterte suksessfaktorer

Tabell 2.2

Risiko ID	Scenario	Konsekvens
PRR - 03	Tekniske problemer	Tap av filer, programvare, tekniske feil sub - disipliner, etc.
PRR - 04	Forsinkelser i arbeid og ferdigstilling	Klarer ikke overholde tidsplanen
PRR - 07	Sene endringer	Kravendringer fra oppdragsgiver sent i prosjektforløpet
PRR - 09	Avhengigheter mellom disipliner	Vi klarer ikke avdekke vesentlige avhengigheter mellom disiplinene tidlig nok, og blir avventende i forhold til hverandre
PRR - 10	Ikke tilstrekkelig teknisk kommunikasjon mellom disipliner	For lite tydelig kommunikasjon og felles planlegging mellom disipliner som har direkteavhengighet i form av grensesnitt
PRR - 11	Mangel på tid fører til lite gjennomtenkte løsninger	Fare for at ambisjonsnivået kombinert med tidsbegrensning leder til forhastede og lite gjennomtenkte slutninger
PRR - 12	Fokus utenfor User Stories definert i sprint	Fare for at ikke oppgavene blir fullført innen tidsfrist og dernest forsinkelser i prosjektet og/eller ufullstendig produkt.

2.2.0 Kategorisering

Tabell 2.2.3

PRR - 03	Tekniske problemer		Tap av filer, programvare, tekniske feil i sub - disipliner, etc.
Sannsynlighet	6	76	
Sårbarhet	12		
Konsekvens	Merarbeid, forskyvning i tidsplanen		
Tiltak	a) Ha gode rutiner for backup, lagring og historikk		

Tabell 2.2.4

PRR - 04		Forsinkelser i arbeid og ferdigstilling		Klarer ikke overholde tidsplanen	
Sannsynlighet		8	64		
Sårbarhet		8			
Konsekvens		Forsinkelser i prosjektet som kan føre til ufullstendig resultat			
Tiltak		a) Ha en realistisk tidsplan b) Hele tiden revurdere tidsbruk og tidsplan c) Bryte ned oppgaver i mindre deler d) Revurdere vinklingen på problemstillinger og hvordan vi løser			

Tabell 2.2.7

PRR - 07	Sene endringer		Kravendringer fra oppdragsgiver sent i prosjektførsløpet
Sannsynlighet	5	40	
Sårbarhet	8		

Konsekvens	Merarbeid, merutgifter og tidsutfordringer. Kan gi ufullstendig produkt
Tiltak	a) Ha skriftlige føringer for dette tidlig i prosess b) Ha plan for smidige endringer/ endringshåndtering c) Bruke en tilrettelagt /egnet prosjektmodell

Tabell 2.2.9

PRR - 09	Avhengighet mellom disipliner		Vi klarer ikke avdekke vesentlige avhengigheter mellom disiplinene tidlig nok, og blir avventende i forhold til hverandre
Sannsynlighet	10	100	
Sårbarhet	10		
Konsekvens	Frustrasjon, forsinkelse i prosjekt, merkostnad		
Tiltak	a) Avklare på forhånd flest mulig overordnede (forventede) avhengigheter b) Være tydelige i vår kommunikasjon om hva som må foreligge i hvilken rekkefølge under egen disiplin c) Vise interesse for og tilegne seg nødvendig kunnskap om de andre disipliners områder i prosjektet		

Tabell 2.2.10

PRR - 10		Manglene forklarende teknisk kommunikasjon mellom disipliner		For lite tydelig kommunikasjon og felles planlegging mellom disipliner rundt tekniske avhengigheter (interfaces) av hverandre
Sannsynlighet		7	42	
Sårbarhet		6		
Konsekvens		Feil og mangler som følge av vesentlige misforståelser, forsinkelser og frustrasjon		
Tiltak		a) Ha føringer for god og tydelig kommunikasjon b) Legge til rette for planlegging sammen c) Være bevisst sitt tekniske språk og forklaringer opp mot medstudenter d) Skriftlige avtaler på løsninger		

Tabell 2.2.11

PRR - 11		Mangel på tid fører til lite gjennomtenkte løsninger		Fare for at ambisjonsnivået kombinert med tidsbegrensning leder til forhastede og lite gjennomtenkte beslutninger
Sannsynlighet		10	100	
Sårbarhet		10		
Konsekvens		Vesentlige feil og mangler, ufullstendig produkt. Forsinkelser og merarbeid/kostnader		
Tiltak		a) Forsøker å ha en realistisk forestilling om innhold i oppgaven, samt tidsbruk b) Ha føringer for hvordan vi best velger løsninger c) Holde korte møter ved alle veivalg, tydelig kommunikasjonen mellom fagretningene for å forsikre oss om at vi sitter med samme oppfatning.		

Tabell 2.2.12

PRR - 12		Fokus utenfor User Stories definert i sprint		Fare for at ikke oppgavene blir fullført innen tidsfrist og dernest forsinkelser i prosjektet og/eller ufullstendig produkt.
Sannsynlighet		9	90	
Sårbarhet		10		
Konsekvens		Ufullstendig produkt, forsinkelser og frustrasjon		
Tiltak		a) Daily scrum b) Eierskap til US i sprint c) Kommunikasjon og tydelig ledelse.		

3 Spesifikk risikoanalyse

Oversikt risiko elektro

Tabell 3.1

Risiko ID	Scenario Elektro
ELR - 01	Feil på bestilt komponent
ELR - 02	Bestilt komponent ødelagt eller forsvinner i frakt
ELR - 03	Overspenning
ELR - 05	Elektromagnetisk forstyrrelse
ELR - 07	Feilkobling
ELR - 08	Komponenter som ryker gjennom test

3.1.0 Kategorisering

Tabell 3.1.1

ELR - 01		Feil på bestilt komponent		
Sannsynlighet		4	40	
Sårbarhet		10		
Konsekvens		Merarbeid, merkostnad, forsinkelser		
Tiltak		a) Kontrollere komponent ved mottakelse b) Være bevisst på valgt av leverandør		

Tabell 3.1.2

ELR - 02		Bestilling forsvinner eller ødelegges under frakt		
Sannsynlighet		4	40	
Sårbarhet		10		
Konsekvens		Tidsplan forskyves, merkostnader		
Tiltak		a) Sporing av pakke b) Kontrollere pakke ved mottakelse c) Være bevisst på valg av leverandør		

Tabell 3.1.3

ELR - 03		Overspenning		
Sannsynlighet		3	30	
Sårbarhet		10		
Konsekvens	Ødeleggelser, tidsplan forskyves, merkostnader			
Tiltak	a) Kontroll/" overspenningsvern" b) Forsikre at riktig mengde spenning/strøm føres til komponenter			

Tabell 3.1.5

ELR - 05		Elektromagnetisk forstyrrelse		
Sannsynlighet		3	24	
Sårbarhet		8		
Konsekvens		Signalforstyrrelser, systemet virker ikke optimalt		

Tiltak	a) Isolering/filtrering av støy
--------	---------------------------------

Tabell 3.1.7

ELR - 07	Feilkobling		
Sannsynlighet	5	50	
Sårbarhet	10		
Konsekvens			
Tiltak	a) Kontrollering før testing		

Tabell 3.1.8

ELR - 08		Komponenter som ryker i test		
Sannsynlighet		3	30	
Sårbarhet		10		
Konsekvens		Merkostnad, tidsforskyvninger		
Tiltak		a) Bestille backup på forhånd b) Sørge for sikkert testmiljø/område c) Kontroll på inngangsverdier		

Oversikt risiko data

Tabell 3.2

Risk ID	Scenario
DAR - 01	Tidkrevende feil i kode.
DAR - 05	Ikke klarer tidsfrist.

3.2.0 Kategorisering

Tabell 3.2.1

DAR - 01	Tidkrevende feil i kode		
Sannsynlighet	7	56	
Sårbarhet	8		
Konsekvens	Forsinkelser, merarbeid		
Tiltak	a) Planlegge for feilretting, debugging		

Tabell 3.2.5

DAR - 05	Klarer ikke tidsfrist.		
Sannsynlighet	10	50	
Sårbarhet	5		
Konsekvens	Forsinkelse i prosjektet, merarbeid og merkostnad		
Tiltak	a) Sette frister for ferdigstilling med margin til kritisk deadline. b) Prioriterer realistisk slik at de viktigste funksjonene blir fullført.		

Oversikt risiko maskin

Tabell 3.3

Risk ID	Scenario
MAR - 12	For stor belastning
MAR - 15	Problemer med gjennomføring av simulering

MAR - 19	3D - printer utilgjengelig/ute av drift
MAR - 21	Overdimensjonering/vekt

3.3.0 Kategorisering

Tabell 3.3.12

MAR - 12	For stor belastning på prototype og utstyr		
Sannsynlighet	7	42	
Sårbarhet	6		
Konsekvens	Brudd / tretthetsbrudd, merkostnad, ikke presis.		
Tiltak	a) Endre vinkler på utstyr. b) Kontrollere innfestninger. c) Kontrollere inngangsverdier d) Testing e) Dimensjonering		

Tabell 3.3.15

MAR - 15	Problemer med gjennomføring av simulering		Vanskeligheter med gjennomføring av simulering slik at vi ikke får verifikasjonen vi trenger
Sannsynlighet	8	80	
Sårbarhet	10		
Konsekvens	Usikkerhet rundt mål og funksjonalitet, kan føre til unødvendig tids - og ressursbruk		
Tiltak	a) Forenkle modell så godt det lar seg gjøre før simulering b) Be om veiledning c) Begrense kompleksitet og omfang av simulering		

Tabell 3.3.19

MAR - 19	3D - printer utilgjengelig/ute av drift		Vi får ikke printet deler til prototype når vi trenger/planlegger for dette i sprint
Sannsynlighet	10	60	
Sårbarhet	6		
Konsekvens	Ufullstendig produkt, forsinkelser		
Tiltak	a) Sørge for å ha flere bein (les: Printere) og stå på b) God planlegging		

Tabell 3.3.21

MAR - 21	Overdimensjonerin g		Modell er overdimensjonert og bidrar til uønsket vekt på prototype (maks 5 kg), samt overdimensjonering vil kunne skape problemer for arbeidsutførelse
Sannsynlighet	9	90	
Sårbarhet	10		
Konsekvens	Ufullstendig produkt, merarbeid og forsinkelser		
Tiltak	a) Ha fokus på vekt gjennom hele designprosessen. b) 'Kontrollveie' modell underveis (teoretisk) c) Fysisk veiing og testing		

Oversikt risiko grensesnitt

Tabell 3.4

Risk ID	Scenario
GRR - 01	Inkompatible grensesnitt EE prototype 1
GRR - 02	Kommunikasjonssvikt mellom ulike grensesnitt
GRR - 03	Uforutsett interferens og/eller for høy friksjon under fysisk test
GRR - 04	EE ikke godt nok tilpasset CG, og får ikke utført sin oppgave

3.4.0 Kategorisering

Tabell 3.4.1

GRR - 01		Inkompatible grensesnitt EE prototype 1		Ledd/deler av og til EE passer ikke med hverandre	
Sannsynlighet		6	60		
Sårbarhet		10			
Konsekvens					
Tiltak		a) Være bevisst målsetting b) God kommunikasjon mellom disiplinene c) Gjøre beregninger d) Testing			

Tabell 3.4.2

GRR - 02	Kommunikasjonssvikt mellom de ulike grensesnittene	System vil ikke fungere	
Sannsynlighet	10		

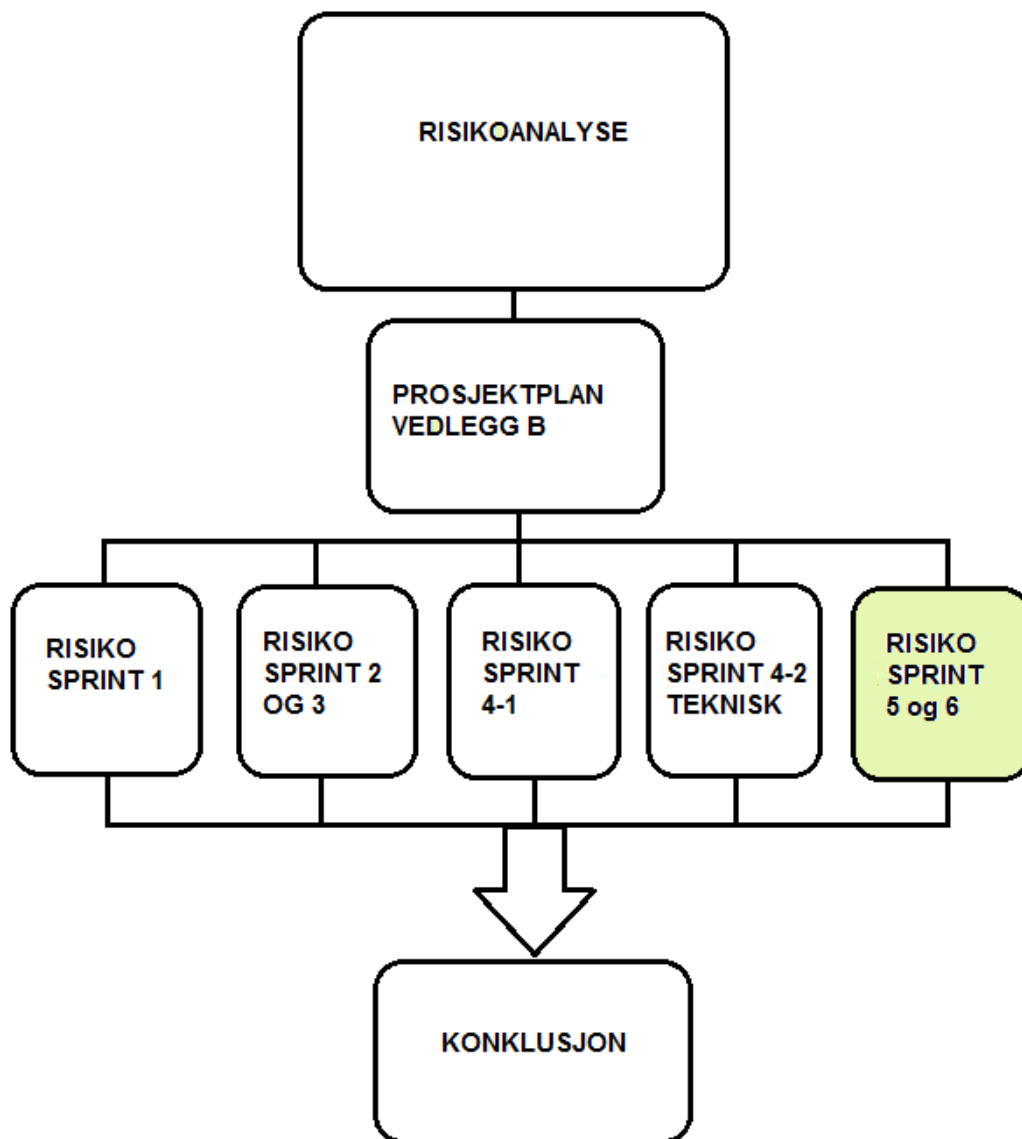
Sårbarhet	10	100	
Konsekvens	Ufullstendig produkt		
Tiltak	a) Kjøre tester b) Eksternt styresystem		

Tabell 3.4.3

GRR - 03		Uforutsett interferens og/eller for høy friksjon under fysisk testing		Problemer med å utføre oppgaver, uønsket slitasje og behov for mer kraft enn først beregnet	
Sannsynlighet		6	60		
Sårbarhet		10			
Konsekvens		Merkostnad, merarbeid og forsinkelser			
Tiltak		a) Være bevisst målsetting. b) Beregne klaringer. c) Studere sammenstilling/visuell teoretisk testing. d) Smøring.			

Tabell 3.4.4

GRR - 04	EE ikke godt nok tilpasset CG, og får ikke utført sin oppgave		Utforming på fingre eller hånd i sin helhet gjør at grepet ikke passer til CG
Sannsynlighet	10	60	
Sårbarhet	6		
Konsekvens	Systemet får ikke utført sin oppgave		
Tiltak	a) Testing b) Utbedring		



Risikoanalyse sprint 5 og 6

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAKSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Risikoanalyse sprint 5 og 6
VERSJON / DATO	1.7 / 11.05. 2016
FORFATTER	Aina Engen Nilsen



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



2 OVERORDNET RISIKOANALYSE

Personorienterte suksessfaktorer

Tabell 2.1

Risiko ID	Scenario	Konsekvens
PER - 01	Sykdom/ulykke	Gruppemedlem må trekke seg underveis.
PER - 02	Dårlig Kommunikasjon	Misforståelser internt og eksternt. Vanskeligheter med å forklare / ordlegge seg.
PER - 04	Interne konflikter	Uenighet i gruppen skaper splid.
PER - 05	Manglende prosjektstyring	Nøkkelperson(er) for prosjektstyring uteblir
PER - 06	Overbelastning/ Arbeidsmengde	Skjevfordeling arbeidsmengde og individuell grenser for stress
PER - 07	Konflikt mellom gruppe og andre involverte parter	Dette tar fokus og kan forhindre gruppen i å utføre sin oppgave på en tilfredsstillende måte
PER - 08	Manglende formidlingsevne Skriftlig/muntlig	Vil ha problemer med å formidle innhold i oppgaven til eksterne/interne involverte, skriftlig og/eller muntlig.

2.1.0 Kategorisering

Tabell 2.1.1

PER - 01	Sykdom/ulykke		Gruppemedlem må trekke seg underveis
Sannsynlighet	7	42	
Sårbarhet	6		
Konsekvens	Gruppen må fordele arbeid seg i mellom, og kompensere for eventuelle faglige mangler.		
Tiltak	a) Redusere oppgavens omfang tilsvarende ressurstap b) Være meget bevisst arbeidsfordeling, og ta medmenneskelige		

	hensyn c) Sørge for en god og åpen dialog rundt temaet hele veien d) Sørge for å ha tilgang til hverandres arbeid i form av felles plattform
--	--

Tabell 2.1.2

PER - 02	Dårlig kommunikasjon		Misforståelser internt og eksternt. Vanskeligheter med å forklare/ordlegge seg.
Sannsynlighet	6	60	
Sårbarhet	10		
Konsekvens	Fører til alt fra frustrasjon til feiltolkninger, og kan lede til store feil og mangler.		
Tiltak	a) Bli enige på forhånd hvordan kommunikasjon skal foregå, og følge dette. b) Skriftlige avtaler og godkjenninger av dokumenter c) Økt fokus på kommunikasjon i form av medarbeidersamtaler d) Samtaleteknikk (bevisst forhold til avsender - mottaker, tydelighet) e) Gruppedynamikk og lav terskel for diskusjoner		

Tabell 2.1.4

PER - 04		Interne konflikter		Uenighet som fører til splid i gruppen	
Sannsynlighet		3	24		
Sårbarhet		8			
Konsekvens		Forsinkelser i prosjekt, frafall av medlemmer			
Tiltak		a) Ha føringer på hvordan vi tar opp et problem. b) Ved store uenigheter som ikke lar seg løse internt, bør det hentes inn. En utenforstående (eks. veileder) for bistand, eventuelt beslutning. c) Gruppen skal ha en intern arbeidskontrakt. d) Fokus på kommunikasjon i form av medarbeidersamtaler e) Gruppedynamikk og lav terskel for diskusjoner			

Tabell 2.1.5

PER - 05		Manglende prosjektstyring		Nøkkelperson(er) for prosjektstyring uteblir
Sannsynlighet		6	42	
Sårbarhet		7		
Konsekvens		Forsinkelser i prosjekt, ikke fullstendig produkt/feil produkt		
Tiltak		a) Velge en prosjektmodell som gir god oversikt for alle, samt har fokus på kollektiv fremdrift. b) Ha to personer med hovedansvar for fremdrift og oversikt.		

Tabell 2.1.6

PER - 06		Overbelastning/ Arbeidsmengde	Skjevfordeling arbeidsmengde og individuell grenser for stress
Sannsynlighet		7	
Sårbarhet		7	
		49	
Konsekvens	Unødvendig frustrasjon som kan føre til fravær eller frafall i gruppen. Ressurstap og forsinkelser		
Tiltak	a) Være bevisst egne begrensninger og tidlige tegn på overarbeidelse b) Være bevisst på og interessert i medstudenters ve og vel c) Forsøker fordele arbeid så jevnt det lar seg gjøre, og ta hensyn der det er nødvendig d) Være flinke til å si i fra så tidlig som mulig		

Tabell 2.1.7

PER - 07	Konflikt mellom gruppe og involverte parter		Forhindrer gruppen fra å holde fokus på oppgaven. Generer usikkerhet og stress, samt er mentalt belastende
Sannsynlighet	8	80	
Sårbarhet	10		
Konsekvens	Feil fokus, forhindrer fremdrift, dårligere resultat		

Tiltak	a) Følge prosedyrer og retningslinjer for håndtering av personalsaker og personvern (ledelse) b) God og riktig kommunikasjon
--------	---

Tabell 2.1.8

PER - 08	Manglende formidlingsevne Skriftlig/muntlig		Problemer med å formidle innhold i oppgaven til eksterne/interne involverte, skriftlig og/eller muntlig.
Sannsynlighet	6	60	
Sårbarhet	10		
Konsekvens	Helhetlig presentasjon av prosjekt svekkes, karakter		
Tiltak	a) Tilrettelegge arbeid og oppgaver		

Prosjektorienterte suksessfaktorer oversikt

Tabell 2.2

Risiko ID	Scenario	Konsekvens
PRR - 03	Tekniske problemer	Tap av filer, programvare, tekniske feil sub - disipliner, etc.
PRR - 04	Forsinkelser i arbeid og ferdigstilling	Klarer ikke overholde tidsplanen
PRR - 07	Sene endringer	Kravendringer fra oppdragsgiver sent i prosjektforløpet
PRR - 09	Avhengigheter mellom disipliner	Vi klarer ikke avdekke vesentlige avhengigheter mellom disiplinene tidlig nok, og blir avventende i forhold til hverandre
PRR - 10	Ikke tilstrekkelig teknisk kommunikasjon mellom disipliner	For lite tydelig kommunikasjon og felles planlegging mellom disipliner som har direkteavhengighet i form av grensesnitt
PRR - 11	Mangel på tid fører til lite gjennomtenkte løsninger	Fare for at ambisjonsnivået kombinert med tidsbegrensning leder til forhastede og lite gjennomtenkte slutninger

PRR - 12	Fokus utenfor User Stories definert i sprint	Fare for at ikke oppgavene blir fullført innen tidsfrist og dernest forsinkelser i prosjektet og/eller ufullstendig produkt.
-----------------	--	--

2.2.0 Kategorisering

Tabell 2.2.3

PRR - 03		Tekniske problemer		Tap av filer, programvare, tekniske feil i sub - disipliner, etc.	
Sannsynlighet		6	76		
Sårbarhet		12			
Konsekvens		Merarbeid, forskyvning i tidsplanen			
Tiltak		a) Ha gode rutiner for backup, lagring og historikk			

Tabell 2.2.4

PRR - 04		Forsinkelser i arbeid og ferdigstilling		Klarer ikke overholde tidsplanen	
Sannsynlighet		8	64		
Sårbarhet		8			
Konsekvens		Forsinkelser i prosjektet som kan føre til ufullstendig resultat			
Tiltak		a) Ha en realistisk tidsplan b) Hele tiden revurdere tidsbruk og tidsplan c) Bryte ned oppgaver i mindre deler d) Revurdere vinklingen på problemstillinger og hvordan vi løser			

Tabell 2.2.7

PRR - 07	Sene endringer		Kravendringer fra oppdragsgiver sent i prosjektforløpet
Sannsynlighet	5	40	
Sårbarhet	8		
Konsekvens	Merarbeid, merutgifter og tidsutfordringer. Kan gi ufullstendig produkt		
Tiltak	a) Ha skriftlige føringer for dette tidlig i prosess b) Ha plan for smidige endringer/ endringshåndtering c) Bruke en tilrettelagt /egnet prosjektmodell		

Tabell 2.2.9

PRR - 09	Avhengighet mellom disipliner		Vi klarer ikke avdekke vesentlige avhengigheter mellom disiplinene tidlig nok, og blir avventende i forhold til hverandre
Sannsynlighet	10	100	
Sårbarhet	10		
Konsekvens	Frustrasjon, forsinkelse i prosjekt, merkostnad		
Tiltak	a) Avklare på forhånd flest mulig overordnede (forventede) avhengigheter b) Være tydelige i vår kommunikasjon om hva som må foreligge i hvilken rekkefølge under egen disiplin c) Vise interesse for og tilegne seg nødvendig kunnskap om de andre disipliners områder i prosjektet		

Tabell 2.2.10

PRR - 10	Manglene forklarende teknisk kommunikasjon mellom disipliner	For lite tydelig kommunikasjon og felles planlegging mellom disipliner rundt tekniske avhengigheter (interfaces) av hverandre	
Sannsynlighet	7		

Sårbarhet	6	42	
Konsekvens	Feil og mangler som følge av vesentlige misforståelser, forsinkelser og frustrasjon		
Tiltak	a) Ha føringer for god og tydelig kommunikasjon b) Legge til rette for planlegging sammen c) Være bevisst sitt tekniske språk og forklaringer opp mot medstudenter d) Skriftlige avtaler på løsninger		

Tabell 2.2.11

PRR - 11		Mangel på tid fører til lite gjennomtenkte løsninger		Fare for at ambisjonsnivået kombinert med tidsbegrensning leder til forhastede og lite gjennomtenkte beslutninger	
Sannsynlighet		10	100		
Sårbarhet		10			
Konsekvens		Vesentlige feil og mangler, ufullstendig produkt. Forsinkelser og merarbeid/kostnader			
Tiltak		a) Forsøker å ha en realistisk forestilling om innhold i oppgaven, samt tidsbruk b) Ha føringer for hvordan vi best velger løsninger c) Holde korte møter ved alle veivalg, tydelig kommunikasjonen mellom fagretningene for å forsikre oss om at vi sitter med samme oppfatning.			

Tabell 2.2.12

PRR - 12		Fokus utenfor User Stories definert i sprint		Fare for at ikke oppgavene blir fullført innen tidsfrist og dernest forsinkelser i prosjektet og/eller ufullstendig produkt.
Sannsynlighet		9	90	
Sårbarhet		10		
Konsekvens	Ufullstendig produkt, forsinkelser og frustrasjon			

Tiltak	a) Daily scrum b) Eierskap til US i sprint c) Kommunikasjon og tydelig ledelse.
--------	---

3 Spesifikk risikoanalyse

Oversikt risiko elektro

Tabell 3.1

Risiko ID	Scenario Elektro
ELR - 01	Feil på bestilt komponent
ELR - 02	Bestilt komponent ødelagt eller forsvinner i frakt
ELR - 03	Overspenning
ELR - 05	Elektromagnetisk forstyrrelse
ELR - 07	Feilkobling
ELR - 08	Komponenter som ryker gjennom test

3.1.0 Kategorisering

Tabell 3.1.1

ELR - 01	Feil på bestilt komponent	
Sannsynlighet	4	40
Sårbarhet	10	

Konsekvens	Merarbeid, merkostnad, forsinkelser
Tiltak	a) Kontrollere komponent ved mottakelse b) Være bevisst på valgt av leverandør

Tabell 3.1.2

ELR - 02		Bestilling forsvinner eller ødelegges under frakt		
Sannsynlighet		4	40	
Sårbarhet		10		
Konsekvens		Tidsplan forskyves, merkostnader		
Tiltak		a) Sporing av pakke b) Kontrollere pakke ved mottakelse c) Være bevisst på valg av leverandør		

Tabell 3.1.3

ELR - 03		Overspenning		
Sannsynlighet		3	30	
Sårbarhet		10		
Konsekvens	Ødeleggelser, tidsplan forskyves, merkostnader			
Tiltak	a) Kontroll/" overspenningsvern" b) Forsikre at riktig mengde spenning/strøm føres til komponenter			

Tabell 3.1.5

ELR - 05		Elektromagnetisk forstyrrelse		
Sannsynlighet		3	24	
Sårbarhet		8		
Konsekvens		Signalforstyrrelser, systemet virker ikke optimalt		
Tiltak		a) Isolering/filtrering av støy		

Tabell 3.1.7

ELR - 07	Feilkobling		
Sannsynlighet	5	50	
Sårbarhet	10		
Konsekvens			
Tiltak	a) Kontrollering før testing		

Tabell 3.1.8

ELR - 08		Komponenter som ryker i test		
Sannsynlighet		3	30	
Sårbarhet		10		
Konsekvens	Merkostnad, tidsforskyvninger			
Tiltak	a) Bestille backup på forhånd b) Sørge for sikkert testmiljø/område c) Kontroll på inngangsverdier			

Oversikt risiko data

Tabell 3.2

Risk ID	Scenario
DAR - 01	Tidkrevende feil i kode.
DAR - 05	Ikke klarer tidsfrist.

3.2.0 Kategorisering

Tabell 3.2.1

DAR - 01	Tidkrevende feil i kode		
Sannsynlighet	7	56	
Sårbarhet	8		
Konsekvens	Forsinkelser, merarbeid		
Tiltak	a) Planlegge for feilretting, debugging		

Tabell 3.2.5

DAR - 05	Klarer ikke tidsfrist.		
Sannsynlighet	10	50	
Sårbarhet	5		
Konsekvens	Forsinkelse i prosjektet, merarbeid og merkostnad		
Tiltak	a) Sette frister for ferdigstilling med margin til kritisk deadline. b) Prioriterer realistisk slik at de viktigste funksjonene blir fullført.		

Oversikt risiko maskin

Tabell 3.3

Risk ID	Scenario
MAR - 12	For stor belastning
MAR - 15	Problemer med gjennomføring av simulering
MAR - 19	3D - printer utilgjengelig/ute av drift
MAR - 21	Overdimensjonering/vekt

3.3.0 Kategorisering

Tabell 3.3.12

MAR - 12	For stor belastning på prototype og utstyr		
Sannsynlighet	7	42	
Sårbarhet	6		
Konsekvens	Brudd / tretthetsbrudd, merkostnad, ikke presis.		
Tiltak	a) Endre vinkler på utstyr. b) Kontrollere innfestninger. c) Kontrollere inngangsverdier d) Testing e) Dimensjonering		

Tabell 3.3.15

MAR - 15	Problemer med gjennomføring av simulering		Vanskeligheter med gjennomføring av simulering slik at vi ikke får verifikasjonen vi trenger
Sannsynlighet	8		

Sårbarhet	10	80	
Konsekvens	Usikkerhet rundt mål og funksjonalitet, kan føre til unødvendig tids - og ressursbruk		
Tiltak	a) Forenkle modell så godt det lar seg gjøre før simulering b) Be om veiledning c) Begrense kompleksitet og omfang av simulering		

Tabell 3.3.19

MAR - 19	3D - printer utilgjengelig/ute av drift		Vi får ikke printet deler til prototype når vi trenger/planlegger for dette i sprint
Sannsynlighet	10	60	
Sårbarhet	6		
Konsekvens	Ufullstendig produkt, forsinkelser		
Tiltak	a) Sørge for å ha flere bein (les: Printere) og stå på b) God planlegging		

Tabell 3.3.21

MAR - 21	Overdimensjonering		Modell er overdimensjonert og bidrar til uønsket vekt på prototype (maks 5 kg), samt overdimensjonering vil kunne skape problemer for arbeidsutførelse
Sannsynlighet	9	90	
Sårbarhet	10		
Konsekvens	Ufullstendig produkt, merarbeid og forsinkelser		
Tiltak	a) Ha fokus på vekt gjennom hele designprosessen. b) 'Kontrollveie' modell underveis (teoretisk) c) Fysisk veiing og testing		

Oversikt risiko grensesnitt

Tabell 3.4

Risk ID	Scenario
GRR - 01	Inkompatible grensesnitt EE prototype 1
GRR - 02	Kommunikasjonssvikt mellom ulike grensesnitt
GRR - 03	Uforutsett interferens og/eller for høy friksjon under fysisk test
GRR - 04	EE ikke godt nok tilpasset CG, og får ikke utført sin oppgave

3.4.0 Kategorisering

Tabell 3.4.1

GRR - 01		Inkompatible grensesnitt EE prototype 1		Ledd/deler av og til EE passer ikke med hverandre
Sannsynlighet		6	60	
Sårbarhet		10		
Konsekvens				
Tiltak		a) Være bevisst målsetting b) God kommunikasjon mellom disiplinene c) Gjøre beregninger d) Testing		

Tabell 3.4.2

GRR - 02		Kommunikasjonssvikt mellom de ulike grensesnittene		System vil ikke fungere	
Sannsynlighet		10			

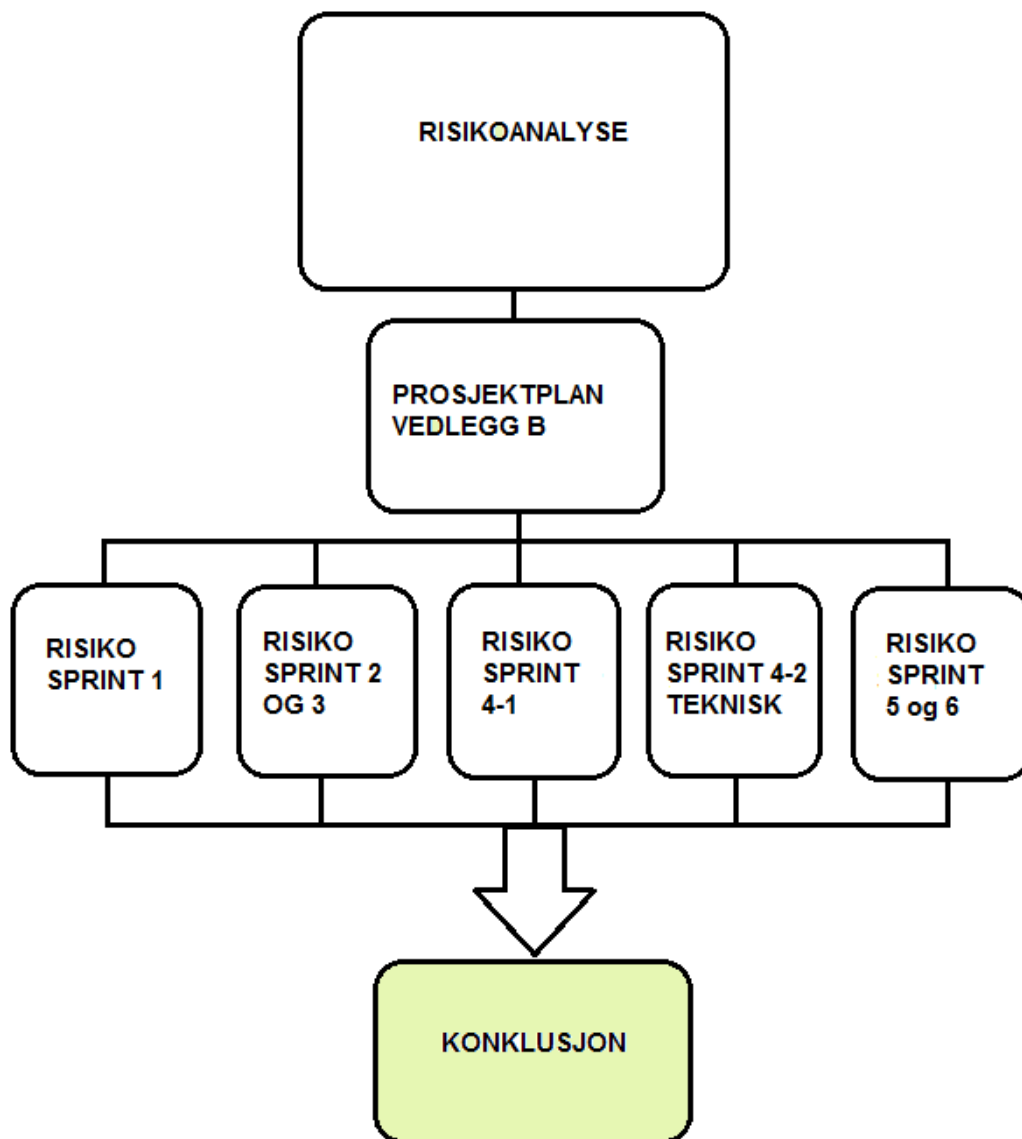
Sårbarhet	10	100	
Konsekvens	Ufullstendig produkt		
Tiltak	a) Kjøre tester b) Eksternt styresystem		

Tabell 3.4.3

GRR - 03		Uforutsett interferens og/eller for høy friksjon under fysisk testing		Problemer med å utføre oppgaver, uønsket slitasje og behov for mer kraft enn først beregnet	
Sannsynlighet		6	60		
Sårbarhet		10			
Konsekvens		Merkostnad, merarbeid og forsinkelser			
Tiltak		a) Være bevisst målsetting. b) Beregne klaringer. c) Studere sammenstilling/visuell teoretisk testing. d) Smøring.			

Tabell 3.4.4

GRR - 04	EE ikke godt nok tilpasset CG, og får ikke utført sin oppgave		Utforming på fingre eller hånd i sin helhet gjør at grepet ikke passer til CG
Sannsynlighet	10	60	
Sårbarhet	6		
Konsekvens	Systemet får ikke utført sin oppgave		
Tiltak	a) Testing b) Utbedring		



Konklusjon

Det er ikke nødvendigvis et mål å unngå all risiko. Målet er å planlegge tiltak for å *redusere* risiko, samt håndtere eventuelle scenarioer som kan oppstå på best mulig måte.

Våre ambisjoner har vært å ha et risikonivå som er akseptabelt for gjennomføring av prosjektet, og som foregående analyse reflekterer.

Risikoanalysen har fortløpende blitt evaluert og oppdatert gjennom prosjektet, da vår prosjektmodell er tuftet på hyppig iterasjon.

Vi konkluderer med at vår risikoanalyse omfatter tilfredsstillende mengde scenarier, og mener at vi kan forsvare et akseptabelt nivå for gjennomføring av prosjektet.

I risikohåndteringen ble det tidlig klart at en altfor omfattende analyse vil kreve for mye ressurser å holde vedlike i et slikt prosjekt. I tillegg til å fange opp de *essensielle* risikoscenarier i prosjektet, skal en slik analyse og tilhørende verktøy være effektive å jobbe med. Underveis i prosjektet har vi fått gode råd og veiledning rundt risikohåndtering fra oppdragsgiver. Vi har gjort noen grep, og snevret inn vårt fokus mot hvilke scenarier vi håndterer. Mitigerende tiltak er å foretrekke fremfor reaktive, og et effektivt håndteringsverktøy er essensielt.

Risikoanalysen peker også i retning av test, og har gitt oss tidlig klare oppfatninger av hva som bør få ekstra oppmerksomhet i denne delen av prosjektet. Risiko er derfor knyttet opp mot vårt testverifikasjonsdokument, og danner grunnlaget for rangering av testklasser i dette.

Referanser

[1] Universal Robots AS. (2015). *UR5 - roboter*.

Tilgjengelig fra:

<http://www.universal-robots.com/no/produkter/ur5-robot/>

[2] [A.K.](#) Svendsen ; E.L. [Roa](#) ; H.B. [Sørum](#) ; S. [Rudin](#). "Automated CROWS Testing" Bachelor Thesis, Fakultet for teknologi og maritime fag, Høgskolen i Buskerud og Vestfold, Kongsberg, 2014.

Tilgjengelig fra:

<http://hdl.handle.net/11250/216910>

[3] Publikasjon fra Geminisenteret, Risiko og sårbarhetsstudier ved NTNU og SINTEF.

"Sikkerhet må skapes og gjenskapes hver dag. Det finnes ingen endelige løsninger" [Online]

Tilgjengelig fra:

<http://docplayer.no/132672-Publikasjon-fra-geminisenteret-risiko-og-sarbarhetsstudier-ved-ntnu-og-sintef.html>

Vedlegg 1

ACTEE RISIKOHÅNDTERING					Meget høy risiko (fokus)		Høy risiko (fokus)		Middels risiko		Tilbak: ref tabeller i sin helhet. Risiko sprint 4-1 teknisk.		A: Høyest vekting, omfattende B: Grundig test for høy risiko C: Moderat vekting				
Dato oppdatert: Versjon: 4-1 av Aina		14.04.2016															
Gjennomgang: 14.04.2016																	
Delikende: 5																	
Id		Risiko		Sannsynlighet		Sårbarhet		R= s's		TSE		Tilbak (preventiv = P, Reaktiv = RE)		Testklasse		Kommentar	
Personrelatert																	
PER - 01		Sykdom/ulykke: Gruppemedlem må trekke seg underveis		7				6		42S		Tabel 2.1: b,c og d					
PER - 02		Dårlig kommunikasjon internt / eksternt		6				10		60S		Tabel 2.1.2: a,b,c,d,e		P			
PER - 04		Interne konflikter i gruppa		3				8		24S		Tabel 2.1.4: a, c d,e		P, RE			
PER - 05		Mangende prosjektstyring, nøkkelperson(er) uteilfr		6				7		42S		Tabel 2.1.5: a,b		P			
PER - 06		Overbelastning/ arbeidsmengde skjeftor delt eller for stor		7				7		49S		Tabel 2.1.6: a,b,c,d		P			
Prosjektrelatert																	
PRR - 03		Tekniske problemer som tap av filer, programvare etc		6				12		72T		Tabel 2.2.3: a		P			
PRR - 04		Forsinkelse i arbeid og ferdigstillinger		8				8		64S		Tabel 2.2.4: a,b,c,d		P			
PRR - 07		Sene endringer		5				8		40S		Tabel 2.2.7: a,b,c		P			
PRR - 09		Avtengjelder mellom disipliner		10				10		100S		Tabel 2.2.9: a,b		P			
PRR - 10		Mangende forklarende teknisk kommunikasjon mellom disipliner		7				6		42T		Tabel 2.2.10: a,b,c		P			
PRR - 11		Mangel på tid kan føre til lite gjennomførte løsninger		7				7		49T		Tabel 2.2.11: a,b,c		P			
PRR - 12		Fokus utenfor User Stories definert i sprint		8				7		56S		Tabel 2.2.12: a,b,c		P			
Elektro																	
ELR - 01		Feil på bestilt komponent		4				10		40E		Tabel 3.1.1: a,b		RE,P		B	
ELR - 02		Bestilling forsinket eller ødelagges under frakt		4				10		40E		Tabel 3.1.2: a,b,c		RE,P			
ELR - 03		Overspenning		3				10		30T		Tabel 3.1.3: a,b		P		B	
ELR - 05		Elektromagnetisk forstyrrelse		3				8		24T		Tabel 3.1.5: a		RE		C	
ELR - 07		Feilklodning		5				10		50T		Tabel 3.1.7: a		P		B	
ELR - 08		Komponenter som ryker i test		3				10		30E		Tabel 3.1.8: a,b,c		P		B	
Data																	
DAR - 01		Tidkrevende feil i kode		7				8		56S		Tabel 3.2.1: a		P		B	
DAR - 02		Mengde kompetanse som må tilegnes blir for stor		6				10		60T		Tabel 3.2.2: a		P			
DAR - 05		Klart ikke tidstest		10				5		50S		Tabel 3.2.5: a,b		P			
Maskin																	
MAR - 12		For stor belastning på prototype (brudd, spenningskonsentrasjon)		7				6		42E		Tabel 3.3.12: a,b,c,d,e		P, RE		B	
MAR - 15		Problemer ved gjennomføring av simulering		8				10		80T		Tabel 3.3.15: a,b,c		P		B	
MAR - 19		3D - printer utilgjengelig/ute av drift		10				6		60S		Tabel 3.3.19: a,b		P			
MAR - 20		Filer for print ikke egnet		8				6		48TS		Tabel 3.3.20: a,b		P		B	
MAR - 21		Overdimensjonering / vekt		8				6		48T		Tabel 3.3.21: a,b		P		B	
Grensesnitt																	
GRR - 01		Inkompatible interfaces prototype		10				10		100ST		Tabel 3.4.1: a,b,c		P		A	
GRR - 02		Kommunikasjonssikt mellom de ulike grensesnittene		10				10		100T		Tabel 3.4.2: a				A	
GRR - 03		Utoilsett interferens og/eller for høy frekvens under fysisk testing		6				10		60ST		Tabel 3.4.3: a,b,c,d		P, RE		B	

Vedlegg 2

ACTEE RISIKOHÅNTERING				Meget høy risiko (fokus)		Høy risiko (fokus)		Middels risiko		Technical Schedule Echonomical		Tilbak: ref tabeller i sin helhet: Risiko sprint 4 - 2.		A: Høyst vektig, omfattende B: Grundig test for høy risiko C: Moderat vektig	
Dato oppdatert:		Version: 4-2		av/Alina		28.04.2016		28.04.2016							
Gjennomgang:						5		Oppdatert evaluering							
Delaktende:															
Id		Risiko		Sannsynlighet		Sårbarhet		R = s*s		TSE		Tilbak (preventivt = P, Reaktivt = RE)		Testklasse	
Personrelatert															
PER - 01 Sykdom/ulykke: Gruppiemedlem må trekke seg underveis				7				6		42 S		Tabell 2.1.1: b,c og d		P	
PER - 02 Dårlig kommunikasjon internt / eksternt				6				10		60 S		Tabell 2.1.2: a,b,c,d,e		P	
PER - 04 Interne konflikter i gruppa				3				8		24 S		Tabell 2.1.4: a, c d,e		P, RE	
PER - 05 Manglende prosjektskyring, nøkkelperson(er) utebilr				6				7		42 S		Tabell 2.1.5: a,b		P	
PER - 06 Overbelastning/ arbeidsmengde skjeftordit eller for stor				7				7		49 S		Tabell 2.1.6: a,b,c,d		P	
PER - 07 Konflikt mellom gruppe og andre involverte parter				8				10		80 S		Tabell 2.1.7: a, b		P, RE	
Prosjektrelatert															
PRR - 03 Tekniske problemer som tap av filer, programvare etc				6				12		72 T		Tabell 2.2.3: a		P	
PRR - 04 Forsinkelse i arbeid og ferdigstilinger				8				8		64 S		Tabell 2.2.4: a,b,c,d		P	
PRR - 07 Sene endringer				5				8		40 S		Tabell 2.2.7: a,b,c		P	
PRR - 09 Avhengigheter mellom disipliner				10				10		100 S		Tabell 2.2.9: a,b		P	
PRR - 10 Manglende forklarende teknisk kommunikasjon mellom disipliner				7				6		42 T		Tabell 2.2.10: a,b,c		P	
PRR - 11 Mangel på tid kan føre til lite gjennomførte løsninger				10				10		100 T		Tabell 2.2.11: a,b,c		P	
PRR - 12 Fokus utenfor User Stories definert i sprint				9				10		90 S		Tabell 2.2.12: a,b,c		P	
Elektro															
ELR - 01 Feil på bestilt komponent				4				10		40 E		Tabell 3.1.1: a,b		RE,P	
ELR - 02 Bestilling forsynner eller ødelegges under frakt				4				10		40 E		Tabell 3.1.2: a,b,c		RE,P	
ELR - 03 Overspenning				3				10		30 T		Tabell 3.1.3: a,b		P	
ELR - 05 Elektromagnetisk forstyrrelse				3				8		24 T		Tabell 3.1.5: a		RE	
ELR - 07 Feilkobling				5				10		50 T		Tabell 3.1.7: a		P	
ELR - 08 Komponenter som ryker i test				3				10		30 E		Tabell 3.1.8: a,b,c		P	
Data															
DAR - 01 Tidkrevende feil i kode				7				8		56 S		Tabell 3.2.1: a		P	
DAR - 05 Klarer ikke tidstfist				10				5		50 S		Tabell 3.2.5: a,b		P	
Maskin															
MAR - 12 For stor belastning på prototype (brudd, spenningskonsentrasjon)				7				6		42 E		Tabell 3.3.12: a,b,c,d,e		P, RE	
MAR - 15 Problemer ved gjennomføring av simulering				8				10		80 T		Tabell 3.3.15: a,b,c		P	
MAR - 19 3D - printer utilgjengelig/ute av drift				10				6		60 S		Tabell 3.3.19: a,b		P	
MAR - 21 Overdimensjonering / vekt				9				10		90 T		Tabell 3.3.21: a,b,c		P,RE	
Grensesnitt															
GRR - 01 Inkompatible grensesnitt prototype				6				10		60 ST		Tabell 3.4.1: a,b,c,d		P	
GRR - 02 Kommunikasjonssvikt mellom de ulike grensesnittene				10				10		100 T		Tabell 3.4.2: a		A	
GRR - 03 Utbrusett interferens og/eller for høy fiksjon under fysisk testing				6				10		60 ST		Tabell 3.4.3: a,b,c,d		P, RE	
GRR - 04 EE ikke godt nok tilpasset CG, og får ikke utført sin oppgave				10				6		60 S		Tabell 3.4.4: a,b		P, RE	

Vedlegg 3

ACTEE RISIKOHÅNTERING										A: Høyest væking, omfattende
Dato oppdatert: Versjon: 5 av Alpha		11.05.2016		Middel høy risiko (fokus)		Høy risiko (fokus)		Middels risiko		B: Grunlig test for høy risiko
Gjennomgang: 12.05.2016		Oppdatert evaluering		Technical Schedule		Economic		Tiltak: ref tabeller i sin helhet: Risiko sprint 5 og 6		C: Moderat væking
Deltekende:										
Id		Risiko	Sannsynlighet	Sårbarhet	R = s*s	TSE	Tiltak (preventiv = P, Reaktiv = RE)		Testklasse	
Personrelatert										
PER - 01 Sykdom/ulykke: Gruppenmedlem må trekke seg underveis			6	5	30 S		Tabel 2.1.1: b,c og d	P		
PER - 02 Dålig kommunikasjon internt / eksternt			6	10	60 S		Tabel 2.1.2: a,b,c,d,e	P		
PER - 04 Interne konflikter i gruppa			3	8	24 S		Tabel 2.1.4: a, c,d,e	P, RE		
PER - 06 Overbelastning/ arbeidsmengde skjevfordelt eller for stor			4	9	36 S		Tabel 2.1.6: a,b,c,d	P		
PER - 07 Konflikt mellom gruppe og andre involverte parter			7	7	49 S		Tabel 2.1.7: a, b	P, RE		
PER - 08 Manglende formidlingssveve skriftlig/muntlig			6	10	60 T		Tabel 2.1.8: a	P		
Prosjektrelatert										
PRR - 03 Tekniske problemer som tap av filer, programvare etc			6	12	72 T		Tabel 2.2.3: a	P		
PRR - 04 Forsinkelse i arbeid og ferdigstillinger			8	5	40 S		Tabel 2.2.4: a,b,c,d	P		
PRR - 09 Avhengigheter mellom disipliner			6	6	36 S		Tabel 2.2.9: a,b	P		
PRR - 12 Fokus utenfor User Stories definert i sprint			8	5	40 S		Tabel 2.2.12: a,b,c	P		
Elektro										
ELR - 03 Overspenning			2	10	20 T		Tabel 3.1.3: a,b	P	B	
ELR - 05 Elektromagnetisk forstyrrelse			3	8	24 T		Tabel 3.1.5: a	RE	C	
ELR - 07 Feilkobling			2	10	20 T		Tabel 3.1.7: a	P	B	
ELR - 08 Komponenter som ryker i test			2	10	20 E		Tabel 3.1.8: a,b,c	P	B	
Data										
DAR - 01 Tidkrevende feil i kode			7	8	56 S		Tabel 3.2.1: a	P	B	
DAR - 05 Klarer ikke tidsfrist			5	5	25 S		Tabel 3.2.5: a,b	P		
Maskin										
MAR - 12 For stor belastning på prototype (brudd, spenningskonsentrasjon)			4	10	40 E		Tabel 3.3.12: a,b,c,d,e	P, RE	B	
Grensesnitt										
GRR - 02 Kommunikasjonssvikt mellom de ulike grensesnittene			8	10	80 T		Tabel 3.4.2: a		A	
GRR - 04 EE ikke godt nok tilpasset CG, og får ikke utført sin oppgave			8	6	60 S		Tabel 3.4.4: a,b	P, RE	A	

A: Høyest vektning, omfattende
B: Grundig test for høy risiko
C: Moderat vektning

7. Testverifikasjon og risiko

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAGSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Testverifikasjon og risiko
VERSJON / DATO	2.0 / 20.05.2016
FORFATTER	Aina Engen Nilsen
SIDEANTALL	57



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



i. Abstrakt

I testspesifikasjonen defineres selve verdien av å bruke user stories (US) og acceptance criterias (AC).

I dette dokumentet gjøres det rede for verifikasjon av oppfylte acceptance criterias generert fra tekniske user stories.

Disse metodene plukker kompleksiteten i prosjektet ned, og gir en nøye gjennomgang av de ulike kravene.

Våre prosesser knyttes hele veien parallelt opp mot risikovurderinger, som sammen med testverifikasjoner danner kvalitetssikring underveis.

ii. Dokumenthistorie

Tabell 1: Dokumenthistorie

v0.1 05.03.16	Monica: Opprettet dokument. Forord og struktur.
v 0.2 07.03.16	Monica: Tillegg
v 0.3 09.03.16	Monica: Tillegg, struktur for el og Solid Works
v 0.4 11.03.16	Aina: Tillegg innledning
v 0.5 12.03.16	Monica: Tillegg av flere verifikasjon på AC. Aina; lager tabeller, utfylling US/AC og risk.
v 0.6 13.03.16	Aina: fyller ut i tabeller.
v 0.7 14.03.16	Aina: fyller ut tabeller, legger inn tekst
v 0.8 15.03.16 v 0.9 16.03.16	Aina: Oppretter v 1.0. finpuss og ferdigstiller Monica: legger til bedre bilder Oppretter versjon 1.0
v1.1 26.04.16	Bård: Lagt til test T-E6, fyringssensor-krets
v1.2 04.05.16	Aina: legger til tabeller, innhold, setter opp del for systemtest
v1.3 09.05.16	Aina: Legger ut utfyllingsklart dokument, føringer foreligger.
v1.4 10.05.16	Espen: Lagt til tester for software D4 – D12 Thomas: Fylt inn software tester. Formatering.
v 1.5 11.05.16	Aina: Fyller inn tester for maskin, starter tilknytning risiko. Retter småfeil, skriftstørrelse m.m.
v1.6 13.05.16	Carl M: Fyller inn test for elektronikk og system
v 1.7 17.05.16	Thomas: Fyller inn test av fyringssensor, og tilhørende kode.
v 2.0 20.05.16	Aina: Knytter mot risiko, Fyller inn siste innhold, FERDIGSTILLER.

ii. Innholdsfortegnelse

i. Abstrakt.....	2
------------------	---

ii. Dokumenthistorie	3
ii. Innholdsfortegnelse.....	3
iii. Tabelloversikt.....	5
iv. Figuroversikt.....	6
v. Forkortelser.....	8
1 Innledning	9
2 Mekanikk.....	11
3 Elektro.....	20
4 Software.....	32
5 System.....	45
6 Konklusjon	56
Referanser	57

iii. Tabelloversikt

Tabell 1:Dokumenthistorie	3
Tabell 2:Testvekting.....	10
Tabell 3:Testklasser	10
Tabell 4:T-M1	11
Tabell 5:T-M2.....	12
Tabell 6:T-M3.....	13
Tabell 7: T-M5.....	14
Tabell 8: T-M6.....	15
Tabell 9:T-M7.....	16
Tabell 10:T-M8.....	17
Tabell 11:T-M9.....	18
Tabell 12:T-E1	20
Tabell 13:T-E3	21
Tabell 14:Kraft nødvendig for knappebetjening	22
Tabell 15:T-E3	23
Tabell 16:T-E4	24
Tabell 17:T-E5	25
Tabell 18:T-E6	26
Tabell 19:T-E7	28
Tabell 20:T-E8	30
Tabell 21:T-E9	31
Tabell 22:T-D1	32
Tabell 23:T-D2	33
Tabell 24:T-D3	34
Tabell 25:T-D4	35
Tabell 26:T-D5	36
Tabell 27:T-D6	37
Tabell 28:T-D7	39
Tabell 29:T-D8	41
Tabell 30:T-D9	41
Tabell 31:T-D10	43
Tabell 32:T-SY1	45
Tabell 33:T-SY2.....	47
Tabell 34:T-SY3.....	50
Tabell 35:T-SY4.....	52
Tabell 36:T-SY5.....	53

iv. Figuroversikt

Figur 1: første prototype	12
Figur 2: hånd/arm grensesnitt	13
Figur 3: rom for komponenter	14
Figur 4: modell av CG	15
Figur 5: EE med «dummy» - CG	16
Figur 6: Gripeverktøyets finger med servoer	16
Figur 7: anordning «håndflate»	17
Figur 8: Komponenthus i pleksiglass monteres på eksisterende kobling for første test.	19
Figur 9: Kobling mot UR5 (grensesnitt) printet i ABS plast	19
Figur 10: Grensesnitt i ABS plast	19
Figur 11: Servomotor	20
Figur 12: Control Grip	21
Figur 13: Måling av kraft på CG	21
Figur 14: Arduino Mega 2560	23
Figur 15: Bluetooth enhet	23
Figur 16: Orcad simulering, Fyringskrets 32V	26
Figur 17: OrCad simulering, Fyringskrets 18V	27
Figur 18: OrCad simulering, Fyringskrets 8V	27
Figur 19: Redesignet krets UR5-Arduino	28
Figur 20: Arduino AT mega 2560	30
Figur 21: Test på WS	31
Figur 22: resetsensor	35
Figur 23: Fra sensor til loggfil	35
Figur 24: Kode for å skrive til logg	36
Figur 25: Loggen under kjøring av test	37
Figur 26: Våpentest	38
Figur 27: Sending av instruksjoner	39
Figur 28: Test editor	40
Figur 29: Lese data fra robot	42
Figur 30: Funksjoner ved mottak av strenger	42
Figur 31: Kode for å skrive til og resette log.txt	43
Figur 32: Koden som lytter etter endringer i loggfilen, og leser loggen til GUI	44
Figur 33: Modifisering slik at ikke ytreledd interfererer med mellomledd	46
Figur 34: Test av Servomotorene mot CG	46
Figur 35: Aller første test av EE 3. mai	47
Figur 36: FEM analyse, Von Mises stress	48
Figur 37: Servomotor	48
Figur 38: Finger med servoer montert	48
Figur 39: Koblingsskjema for bevegelse til servomotorene	49
Figur 40: Ny fingertupp kan utføre å løfte vippebryter	50
Figur 41: Fingertupp vipper indre bryter.	51

Figur 42:Fingertupp lukker vippebryter.	51
Figur 43: Indre bryter lukker seg sammen med ytre vippebryter.	52
Figur 44: Ny fingertupp til EE fungerer som den skal og klarer å trykke inn knapper på DCP	53
Figur 45: EE har fått på sugru og fått kabler gjemt.	54
Figur 46: EE griper rundt CG og trykker palm og trigger knapp.	55

v. Forkortelser

KPS	Kongsberg Protech Systems
HSN	Høgskolen i Sørøst-Norge
CG	Control Grip, styringsjoystick for våpenstasjon
DCP	Deportable Control Panel
EE	End Effector, gripeverktøy
UR5 [5]	Universal Robots 5 robotarm
US	User Stories
AC	Acceptance Criterias
CROWS	Common Remotely Operated Weapon Station
FEM	Final Element Method (analyse)
WS	Weapon Station
LSB	Least significant bit
MSB	Most significant bit
GND	Ground
GPIO	General purpose input output
FPGA	Field-programmable gate array
TP	Twisted pair
TCP/IP	Transmission Control Protocol/Internet Protocol

1 Innledning

Mål med test

Målet med testingen som gjennomføres er ikke bare å verifisere og validere spesifikasjoner og dokumentasjon. Like viktig er det at testingen defineres som en prosess som gjør det mulig å identifisere og teste risikomomenter slik at løsningen som implementeres tilfredsstillende oppdragsgivers krav (formulert som US).

Omfanget av testingen som skal gjennomføres bestemmes av størrelse på og innhold i sluttprodukt, men det er viktig at alle elementer ved testgjennomføringen vurderes og dokumenteres.

Hensikten med testing er å sikre at:

- Systemet oppfyller alle funksjonelle og ikke - funksjonelle krav (ytelse, kapasitet, stabilitet, sikkerhet, brukervennlighet, pålitelighet m.m.).
- Systemet oppfyller godkjenningsskriterier som fremkommer av avtaledokumenter.
- Feil identifiseres for deretter å sørge for at de blir rettet/håndtert før produksjonssetting.

Færre feil i produksjon betyr reduserte vedlikeholdskostnader og vi oppnår mer fornøyde interessenter og sluttbrukere. Kvalitet må bygges inn fra starten av ettersom kvalitet primært ikke kan testes inn i en leveranse. En metodisk og strukturert testgjennomføring vil allikevel bidra til å heve kvaliteten på leveransen.

Testprosessen vil normalt være tilpasset en smidig utviklingsmodell med 2 til 4-ukers sprinter. Ved endring av driftsmiljø vil det ikke nødvendigvis være slik, og testing vil bli tilpasset dette. For å være sikker på at en test som gjennomføres faktisk har en verdi, må testen designes slik at den tester de egenskaper og deler av testobjektet som representerer en risiko hvis de ikke fungerer tilfredsstillende i produksjon.

Utførelse

Selve testingen av de *tekniske* US vil foregå i følgende steg:

1. Startfase (før test): Hva må være klart før systemet kan testes? Hva trenger vi? i hvilket miljø skal det testes?
2. Testfase (under test): Hvordan tester vi best? Hva må være igangsatt (aktiviteter) for at vi skal få testet/ aktivisert systemet?
3. Slutfase (etter test): Hva forventer vi av resultat? Virker systemet på alle nivåer? Hvor rask reaksjonstid? Loggføring? Krefter?

Håndtering av test

I vår oppgave har vi begrensende elementer i form av tid og kompetanse. Vi finner det derfor naturlig å holde fokus for testing rettet mot de tekniske risikoscenarier som vil ha *størst* betydning for vårt resultat dersom de skulle inntreffe. Vi vil bruke vår risikoanalyse til å velge ut prioritert testdekning og omfang, og som tidligere nevnt vil grensesnitt ha brorparten av vår oppmerksomhet.

Ut ifra vår risikomatrise, fikk de ulike scenarier en rangering (tabell 1). Det er denne rangeringen vi nå knytter til testingen.

Tabell 2: Testvekting

Sannsynlighet	Sårbarhet			
	Ubetydelig	Mindre	Moderat	Stor
Sannsynlig	C	B	B	A
Moderat		C	B	A
Usannsynlig			C	B
Sjelden			C	B

Tabell 3: Testklasser

Rangering	Vekting og forklaring
C	Testes med lavest vekting, det vil si minst omfattende test. Allikevel vil endel elementer merket gult måtte testes, da fargen indikerer en moderat risiko.
B	Testes med normal til høy vekting. Risiko i fargen rød forventes å <i>kunne</i> være omfattende, så testprosessen må gjenspeile dette.
A	Tester med høyest vekting. I klasse A vil testene være mest omfattende, under denne fargen ligger risiko med høyest farepotensiale for systemet. Testklasse A tillegges mest tid.

Vi vil ha hovedfokus på de scenarier som har fått farge i rødskalaen, men enkelte av de gule vil også måtte testes for.

2 Mekanikk

Første fase av prosjektet var i stor grad kartlegging av interessenter, krav, kriterier for å oppfylle kravene, fundament for forebygging av risiko, og planlegging. Vi jobbet likevel fra starten med å utforske eksisterende teknologier på vår problemstilling. Derfor er det hensiktsmessig å ta enkelte user stories som omhandler EE fra sprint 1 også. Verifikasjonen av acceptance criterias er ordnet i kronologisk rekkefølge.

I vår oppgave har vi som nevnt begrensende elementer i form av tid og kompetanse. Vi finner det derfor naturlig å holde fokus for testing rettet mot de *tekniske* risikoscenarier som vil ha *størst* betydning for vårt resultat dersom de skulle inntreffe.

Tabell 1: T-M1

Test ID	Klasse	User story	Acceptance criterias	Risiko ID
T-M1	B	US-010701	AC-010701-01: Prototype Godkjent av Product Owner. AC-010701-02: Prototype er ferdig produsert. AC-010701-03: Alle mål og dimensjoner er notert for videre bruk.	MAR-02
				MAR-03
				MAR-07

Risiko tilknyttet Prosjektplan Vedlegg B versjon 1.0 [3]

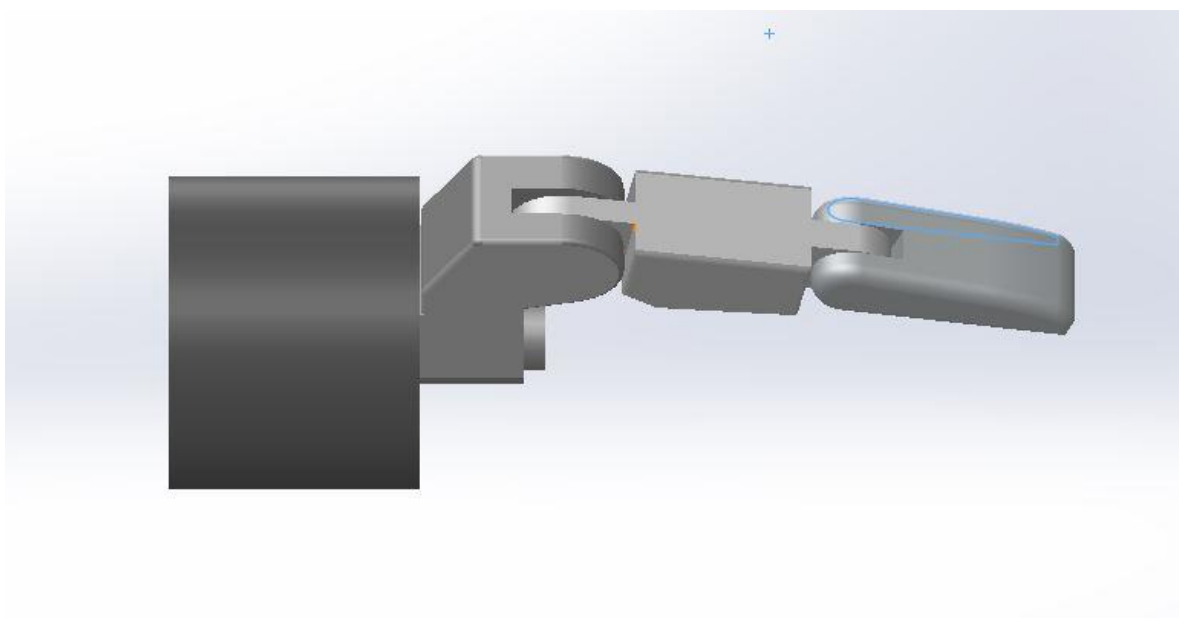
1. Er tatt utgangspunkt i en ferdig robot hånd (etter avtale), førsteutkast modifisert for eget formål. Modellen er tilpasset i Solid Works, filer/ arbeid lagres flere steder for sikkerhet. Målene er satt til omtrent ønsket størrelse, og tar utgangspunkt i mål tilgjengelig fra CG. Mål skal brukes videre, men vil modifiseres ytterligere lenger ut i prosessen.
2. Første utkast er 3D-printet i et nylonmateriale, med en fast finger for illustrasjon av idé/tanke videre.
3. Modellen er godkjent av PO og fremvist på 1. presentasjon. (Se figur 1)

Tabell 5: T-M2

Test ID	Test klasse	User story	Acceptance criterias	Risiko ID
T-M2	B	US-051301	AC-051301-01: Første Prototype er utformet med en "pekefinger"	MAR-02
				MAR-07

Risiko tilknyttet Prosjektplan Vedlegg B versjon 1.0 [3]

1. Mål opp mot fungerende eksisterende hånd, samt CG er kartlagt og tegnet i Solid Works. Oppbygging av fingeren legger grunnlaget for videre arbeid av modellen, og er tilpasset modellens krav til bruksområde.
2. Modellen er printet, mål er kontrollert.
3. Første prototype er fremvist og godkjent av PO.



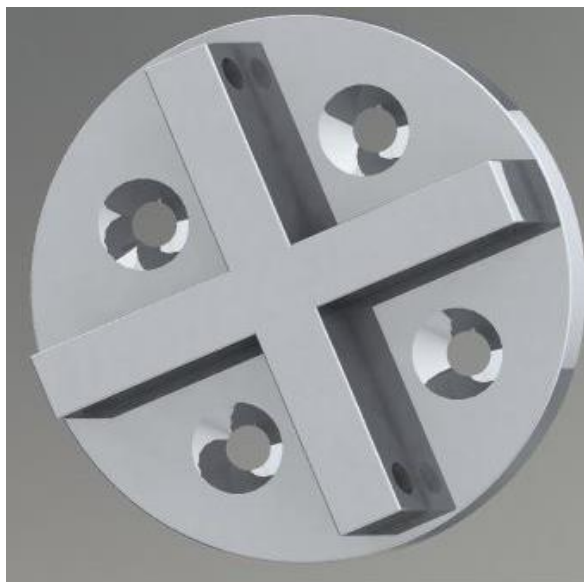
Figur 1: første prototype

Tabell 6: T-M3

Test ID	Klasse	User story	Acceptance criteria	Risiko ID
T-M3	B	US-043003 US-043004	AC-043003-01: Design av grensesnitt er tilpasset monteringshull på UR5. AC-043004-01: Det begrunnes kort i tekst som føres inn i teknologidokumentet.	MAR-07
				MAR-12

Risiko tilknyttet Risikoanalyse sprint 2/3 (versjon 1.3) [3]

1.I følge brukermanual for UR5, kommer det frem at dette er en maskin som er lett integrerbar for brukerne. Det er da lite sannsynlig at slike ting ikke er tatt høyde for, man vil få en stabil nok innfestning til tiltenkt bruk.



Figur 1: hånd/arm grensesnitt

2.Det er modellert en kopling med hull til innfesting for 4 stk. M6 skruer. Det vurderes eventuelt å legge inn gummipakning mellom UR5 og grensesnittet.

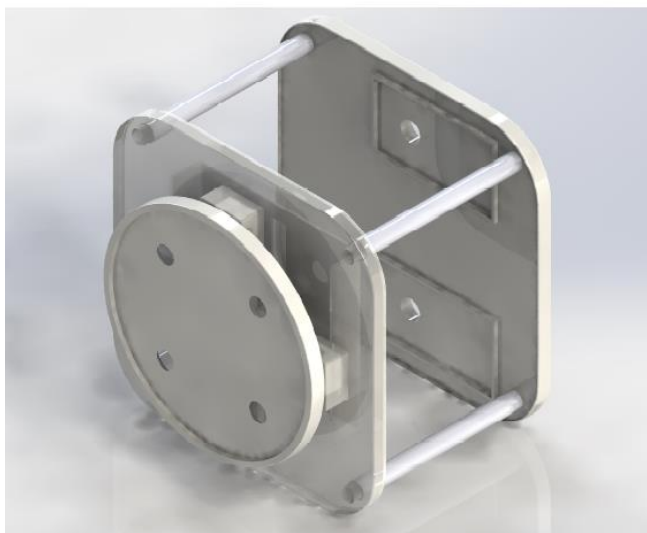
3. Modellert grensesnitt er kontrollmålt opp mot UR5 grensesnitt, faglig vurdering for løsninger begrunnes i teknologidokument.

Tabell 2: T-M5

Test ID	Klasse	User story	Acceptance criterias	Risiko ID
T-M5	C	US-050401	AC-050401-01: Luken er designet	ELR-08 ELR-07 ELR-03
				MAR-12

Risiko tilknyttet Risikoanalyse sprint 2/3 (versjon 1.3) [3]

1. Nødvendig plass for komponenter er beregnet, samt behov for enkel tilgang er prioritert. Vi har sett på flere løsninger enn formulert i US.
2. Det er designet et åpent rom for komponenter, og tilrettelagt for festing av servomotorer og Arduino. Rommet består av tak og gulv med 4 stk. sylindere mellom. Denne løsningen gjør at det er lett å komme til komponentene, det er god lufting ved varmeutvikling fra motorer, og det er mulig å justere volumet ved regulere lengden på sylindere.
3. Designet er teoretisk godkjent av ansvarlige ved elektro og maskin, og vedtatt.



Figur 2: rom for komponenter

Tabell 3: T-M6

Test ID	Klasse	User story	Acceptance criterias	Risiko ID
T-M6	C	US-081001	AC-081001-01: Modell tilnærmet CG er modellert opp i Solid Works.	MAR-15

Risiko tilknyttet Risikoanalyse sprint 4-1 teknisk (versjon 1.4) [3]

En dummy er modellert opp, uten finesse. Modellen er ikke optimalt bygget, men vil kunne fungere som en test -CG. Bevegelse/begrensninger i CG og knapper kan omgjøres/tillegges etter behov. Mål fra 2D CG er brukt der de har vært tilgjengelige. EE sammenstilt med denne vil gi oss en visuell verifikasjon på om målene påsatt kan fungere med den egentlige CG.



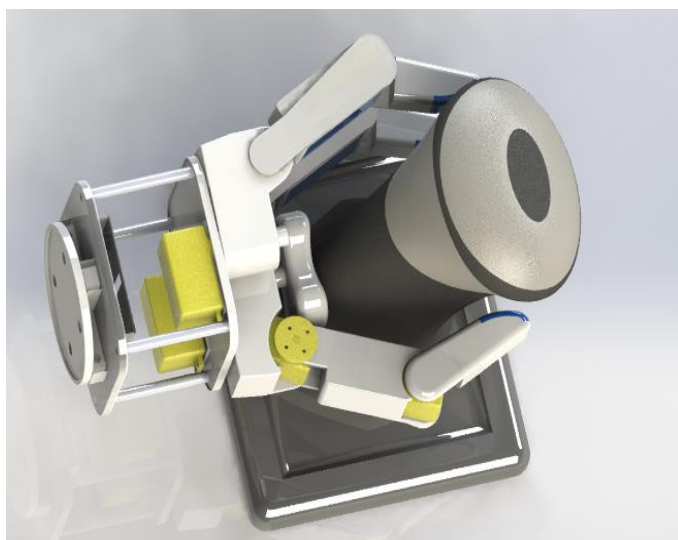
Figur 3: modell av CG

Tabell 9: T-M7

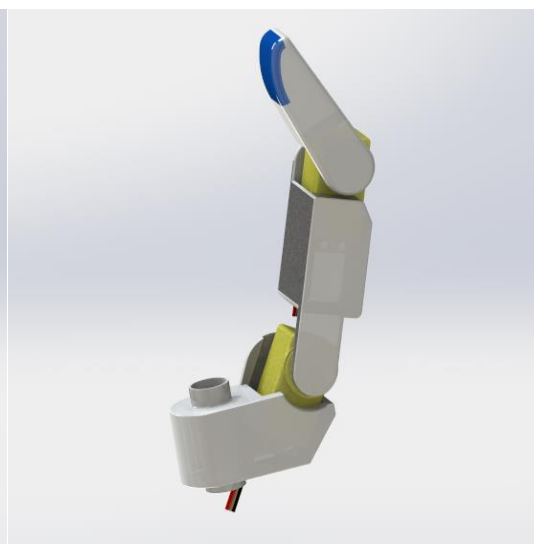
Test ID	Klasse	User story	Acceptance criterias	Risiko ID
T-M7	B	US-04400201	AC-04400201-01: 3D modell detaljert med klart definert funksjon for betjening av Palm AC-04400201-02: Modellen er designet med nødvendige volum for elektronikk og komponenter	PRR-07 PRR-10
				PRR-09 PRR-11 MAR-18

Risiko tilknyttet Risikoanalyse sprint 4-1 teknisk (versjon 1.4) [3]

1. Teknisk informasjon fra ferdig produkt foreligger, det innhentes ytterligere informasjon ved pugh - matriser for de forskjellige komponenter og konsepter. Mål hentes inn fra CG.
2. Det er modellert opp mulige løsning i Solid Works, etter konseptutvelgelse for funksjonalitet og styring.
3. Modeller av komponenter er modellert opp i 1:1 og forsøkt sammenstilt med tilnærmet CG.



Figur 4: EE med «dummy» - CG



Figur 5: Gripeverktøyetts finger med servoer

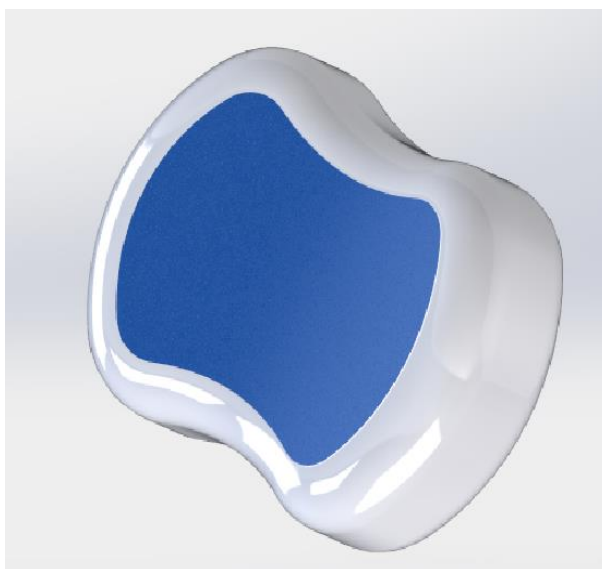
Tabell 10: T-M8

Test ID	Klasse	User story	Acceptance criterias	Risiko ID
T-M8	B	US-044004	AC-044004-01: EE utstyres med "formbar" anordning for best friksjon i grep AC-044004-02: EE 's anordning i håndflaten gir stødig grep om CG ved fysisk test	MAR-15 MAR-12

Risiko tilknyttet Risikoanalyse sprint 4-1 teknisk (versjon 1.4) [3]

1. For best mulig kontroll av CG, skal vi unngå at grepet til EE glipper. Det er modellert opp en "dummy CG" i den hensikt å gi et bilde på foreløpig tilpasning.
2. Det er tegnet en anordning til "håndflaten" med den hensikt å støtte grepet om CG og forhindre glipp. Anordningen skal utstyres med friksjonsbelegg, og utføring med evne til tilforming. I tillegg er det planlagt for sensor i anordningen.
3. Anordning er satt på modellen i Solid Works og studert i sammenstilling.

Kommentar: Denne delen har blitt nedprioritert da vi ønsket å teste prototypen før anordningen ble fastmontert. Det ble planlagt å foreløpig dekke et eventuelt behov for utbygging av «håndflate» med en midlertidig formbar anordning.



Figur 7: anordning «håndflate»

Tabell 11: T-M9

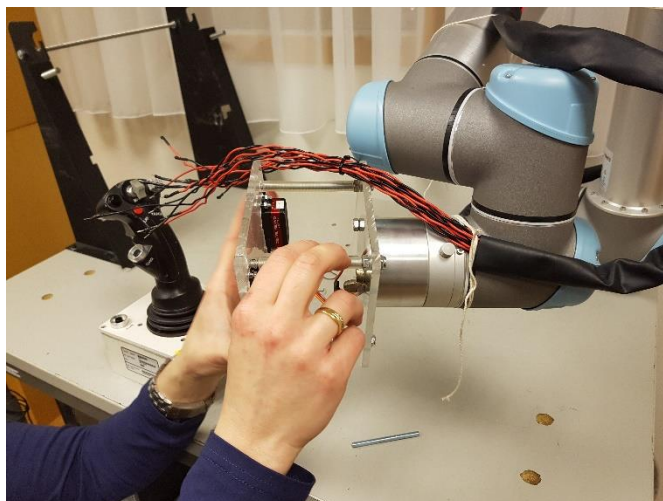
Test ID	Klasse	User story	Acceptance criterias	Risiko ID
T-M9	A	US-043001 US-043002	AC-043001-01: Interface er designet AC-043001-02: Interface er godkjent av PO AC-043001-04: Interface er printet AC-043001-03: Interface er fysisk testet koblet på UR5 AC-043002-01: Koblingen er printet AC-043002-02: Koblingen er fysisk testet	PRR-03 PRR-04 MAR-19
				PRR-11 MAR-21

Risiko tilknyttet Risikoanalyse sprint 4-2 teknisk (versjon 1.6) [3]

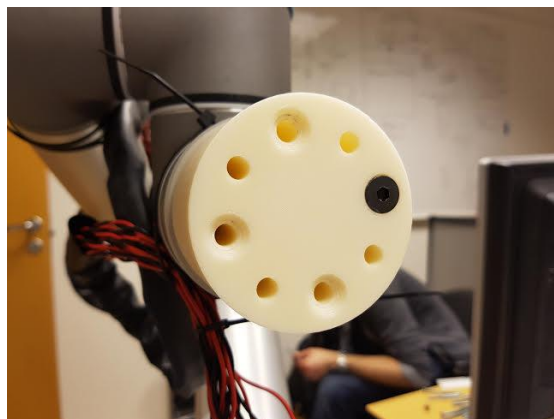
EE skal kobles til UR5 via sitt komponent hus (grensesnitt) og en separat kobling. Hus er designet i tidligere sprint, og er først testet i 6 mm pleksiglass. Herfra har vi avdekket og tillagt ønskede detaljer, før vi har printet huset i ABS plast. Koblingen er grensesnittet mellom komponenthuset og UR5. Denne festes med 4 M6 skruer direkte i UR5. Komponenthuset festes til koblingen med 4 M6 skruer plassert i samme sirkulære mål som innfestning til UR5 Pleksiglass – huset er montert på eksisterende kobling for første test. Huset fungerer som planlagt, vi har derfor beholdt alle mål i vår nye print. ABS er et mye bedre materiale å arbeide med dersom vi har behov for små endringer, samt at vi ønsker en visuell helhet på vårt produkt. Vi har derfor bestemt oss for å printe dette også.

Diskusjon/kommentar:

Vi var ikke godt nok forberedt på bl.a. helligdag og inneklemt fridag i forhold til print denne gangen, og fikk litt mindre tid enn ønskelig for en A - test som følgefeil. Avhengigheter mellom disiplinene gjør seg også gjeldende, da programmering av bevegelser ikke bør skje før alt er montert slik det skal. Til tross for dette har vi fått testet at systemet fungerer, og vil ta opp igjen tråden etter avlevert dokumentasjon.



Figur 6: Komponent hus i plexiglass monteres på eksisterende kobling for første test.



Figur 7: Kobling mot UR5 (grensesnitt) printet i ABS plast



Figur 10: Grensesnitt i ABS plast

3 Elektro

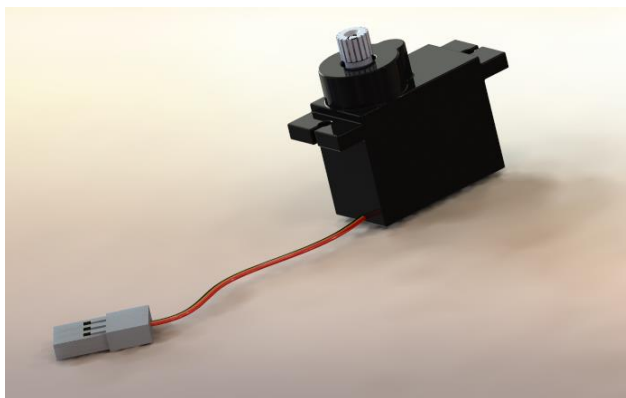
Tabell 12: T-E1

Test ID	Klasse	User story	Acceptance criteria	Risiko ID
T-E1	B	US-051401	AC-051401-01: Utføres pugh matrise av motorer. AC-051401-02: Utføres pugh matrise av mekanisk valg. AC-051401-03: Utføres pugh matrise av valg av design til verktøy.	PRR-09 PRR-11 MAR-18 PRR-07 PRR-10 ELR-01 ELR-04

Risiko tilknyttet Risikoanalyse sprint 2/3 (versjon 1.3) [3]

1. Det er innhentet større mengder data på forskjellige løsninger for både mekanikk og komponenter for styring. Det er bestemt kriterier i forhold til ønsket funksjon, og vektet disse.
2. Pugh - matrisene fylles ut og gir oss de beste forslag etter våre kriterier og vektning på disse.
3. På bakgrunn av Pugh - matrisene har vi kommet frem til mulig konsept på både styring og mekanisk design, men i har ikke låst oss til kun en løsning riktig ennå.

Henviser til Pugh - matriser i teknisk dokument for EE.



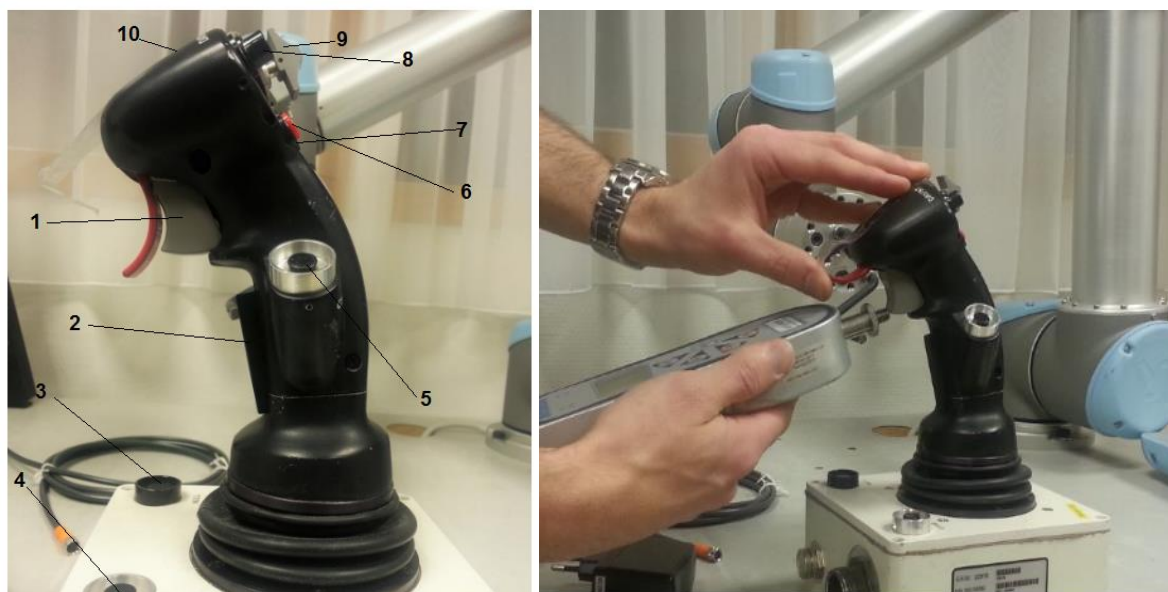
Figur 11: Servomotor

Tabell 13: T-E3

Test ID	Klasse	User story	Acceptance criterias	Risiko ID
T-E2	B	US-050103	AC-050103-01: Komponenter som tilfører kraft, skal resultere i en styrke på minst 25N i angrepspunktet AC-050103-02: Det skal vises til utregninger som legges i teknologidokument	PRR-03
				PRR-07 PRR-10 ELR-01 ELR-04 ELR-08

Risiko tilknyttet Risikoanalyse sprint 2/3 (versjon 1.3) [3]

Via pugh matrisens verdivurdering av krav, har konseptet landet på en Power HD-1207 TG servomotor. Motoren drar 8kg-cm (78N-cm) noe som vil være nok for konseptet. Ved 25N i angrepspunktet er dette satt med god margin fra tidligere. Beregningen vil si at fra eventuelt motor så vil det være kg-cm, for eksempel en enhet som gir 100Ncmcm kan gi 25N-cm ved et 4 cm emne.



Figur 12: Control Grip

Figur 8: Måling av kraft på CG

Det er foretatt en måletest med en Mecmesin AFG (Advanced force gauge) på CG og DCP, og listet verdiene i tabell 13 på neste side. Maks kraft nødvendig er 10N.

Tabell 14: Kraft nødvendig for knappebetjening

Knapp CG	Bevegelse	Kraft nødvendig (N)
1	Klem	10
2	Klem	8
3	Trykk	8
4	Trykk	8
5	Trykk	8
6	Trykk	8
7	Trykk	8
8	Trykk	8
9	Forskyv	6
10	Forskyv	8
11	Trykk	8
Flipp DCP	Flipp	5
Knapp DCP	Trykk	10

Tabell 15: T-E3

Test ID	Klasse	User story	Acceptance criterias	Risiko ID
T-E3	B	US-046302	AC-046302-01: Kode er laget på Arduino for håndtering av trådløs kommunikasjon. AC-046302-02: Kode er testet så langt det lar seg gjøre før loggsystemet er på plass.	PRR-03 ELR-05 DAR-01
				PRR-10 ELR-01 ELR-08

Risiko tilknyttet Risikoanalyse sprint 2/3 (versjon 1.3) [3]

Koden ble skrevet i Arduino IDE [1], og testet på en Arduino MEGA 2560 sammen med en hc-06 Bluetooth modul. I setup - delen av koden ble det lagt til at den skal starte seriell kommunikasjon på seriell kanal 1, og hastigheten på denne bestemt.

Testet med å sende en string via Serial1 interface til tilkoblet Bluetooth enhet på henholdsvis pin 10 og 11. Ved å bruke Putty koblet vi oss til en virtuell Seriell kommunikasjons port opprettet av Bluetooth kortet på datamaskinen vi testet mot. Vi så her at det vi sendte fra Arduino ble mottatt.



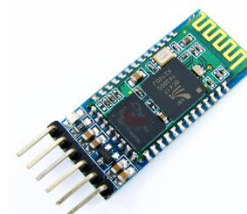
Figur 14: Arduino Mega 2560

Vi testet også simultan tilkobling, både via USB, og Bluetooth. Der vi kommuniserte mellom disse konsoll applikasjonene, via Arduino kortet. Dette fungerte også fint.

Den ble også testet sammen med interrupt - signalet beskrevet i US-046104.

For å teste at den takler å stå en stund, lot vi den stå i 3 timer, og sjekket om den fremdeles var i stand til å kommunisere uten innblanding. Dette fungerte også fint.

Kode er testet så langt det lar seg gjøre før loggsystemet er på plass. Siden det ikke er noe loggsystem på plass, eller bestemt noe spesifikt om kommunikasjonen mellom disse, så er det ikke definert akkurat hva som skal sendes. Det å endre på string som sendes er en rask ting å gjøre når føringer for dette er bestemt.



Figur 9: Bluetooth enhet

Tabell 16: T-E4

Test ID	Klasse	User story	Acceptance criterias	Risiko ID
T-E4	B	US-046104	AC-046104-01: Kode skrevet slik at ved merket avfiring vil det umiddelbart bli sendt beskjed til testmaskin om dette.	ELR-05
				ELR-07

Risiko tilknyttet Risikoanalyse sprint 2/3 (versjon 1.3) [3]

Koden ble skrevet i Arduino IDE [1], og testet på en Arduino MEGA 2560.

Koden ble skrevet slik, ved Rising egde på et signal inn på PIN 3 på Arduino, trigges en interrupt. Interrupten sender en string via USB Serial interface til tilkoblet maskin. Ved å bruke Arduino IDE sin innebygde konsoll kan man se output på skjerm, man kan også bruke andre konsoll verktøy for å lese disse signalene, for eksempel Putty.

For å simulere høyt signal, ble det koblet en liten krets med en trykkbryter og en 220Ohms motstand. Motstanden ble koblet mellom input Pin og jord, trykkbryter mellom +5V og Pin. På denne måten hindrer man at Arduino tolker noen signaler feil, noe den kan gjøre om man lar en Pin stå åpen.

Når vi slutter kretsen ved å trykke inn knappen, fikk vi umiddelbart resultat. Ønsket string dukket opp i konsollen.

Vi kan med dette si at koden fungerer som forventet.

Det ble også gjort en test over tid, vi lot den stå tilkoblet i 2 timer for å sjekke om den ennå fungerte, noe den gjorde.

Tabell 17: T-E5

Test ID	Klasse	User story	Acceptance criterias	Risiko ID
T-E5	B	US-046202	AC-046202-01: Kode er laget på Arduino for håndtering av kommunikasjon til maskin. AC-046202-02: Kode er testet så langt det lar seg gjøre før loggsystemet er på plass.	DAR-01 PRR-03
				PRR-10 ELR-07 ELR-08

Risiko tilknyttet Risikoanalyse sprint 2/3 (versjon 1.3) [3]

Koden ble skrevet i Arduino IDE [1], og testet på en Arduino MEGA 2560.

I setup delen av koden ble det lagt til at den skal starte seriell kommunikasjon, og hastigheten på denne bestemt.

Testet med å sende en string via USB Serial interface til tilkoblet maskin. Ved å bruke Arduino IDE sin innebygde konsoll kan man se output på skjerm, man kan også bruke andre konsoll verktøy for å lese disse signalene, for eksempel Putty.

Den ble også testet sammen med interrupt signalet beskrevet i US-046104.

Kode er testet så langt det lar seg gjøre før loggsystemet er på plass. Siden det ikke er noe loggsystem på plass, eller bestemt noe spesifikt om kommunikasjonen mellom disse, så er det ikke definert akkurat hva som skal sendes. Det å endre på denne strengen er en rask ting å gjøre når føringer for dette er bestemt.

Tabell 18: T-E6

Test ID	Klasse	User story	Acceptance criterias	Risiko ID
T-E6	B	US-046103	AC-046103-01: Ideen til krets er notert ned. AC-046103-02: Konseptet er simulert i for eksempel OrCad for å teste om det er mulig. AC-046103-03: Konseptet er testet i praksis.	PRR-09
				PRR-03 PRR-04 ELR-07 ELR-08
				ELR-05

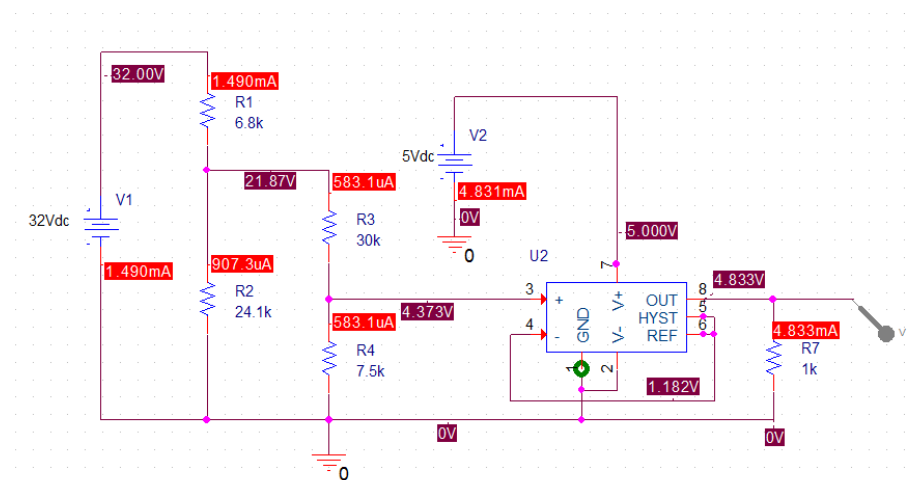
Risiko tilknyttet Risikoanalyse sprint 4-1 teknisk (versjon 1.4) [3]

Kretsen har noen utfordringer da det ikke holder med kun en spenningsdeler for å få ønsket effekt. En enkel spenningsdeler vil gi oss et output på mellom 5V, og 2.8V, da inngangsspenningen på WS (weapon station) bestemmer avfyringssignalet.

Da Arduino ikke registrerer et signal på 2.8V som høyt, risikerer vi da at avfyringer ikke blir registrert. For å forhindre dette har vi brukt en komparator for å «hjelpe» signalet til Arduino.

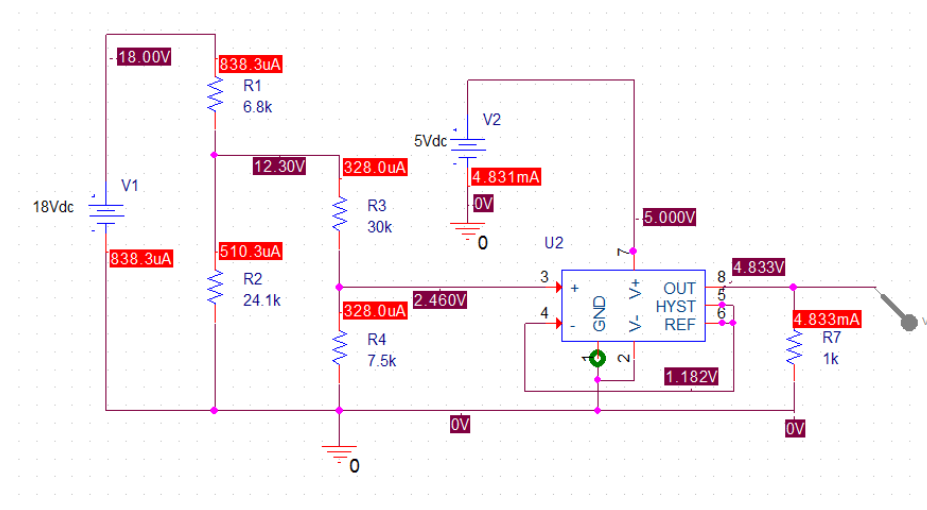
Ved å sette inngang 2 på komparator til 1.2V, vi et hvert signal på inngang 1 som overstiger dette få et utsignal på ca. 5V.

Simulering ble gjort i Orcad for å bekrefte dette. Først med maks inngangsspenning.



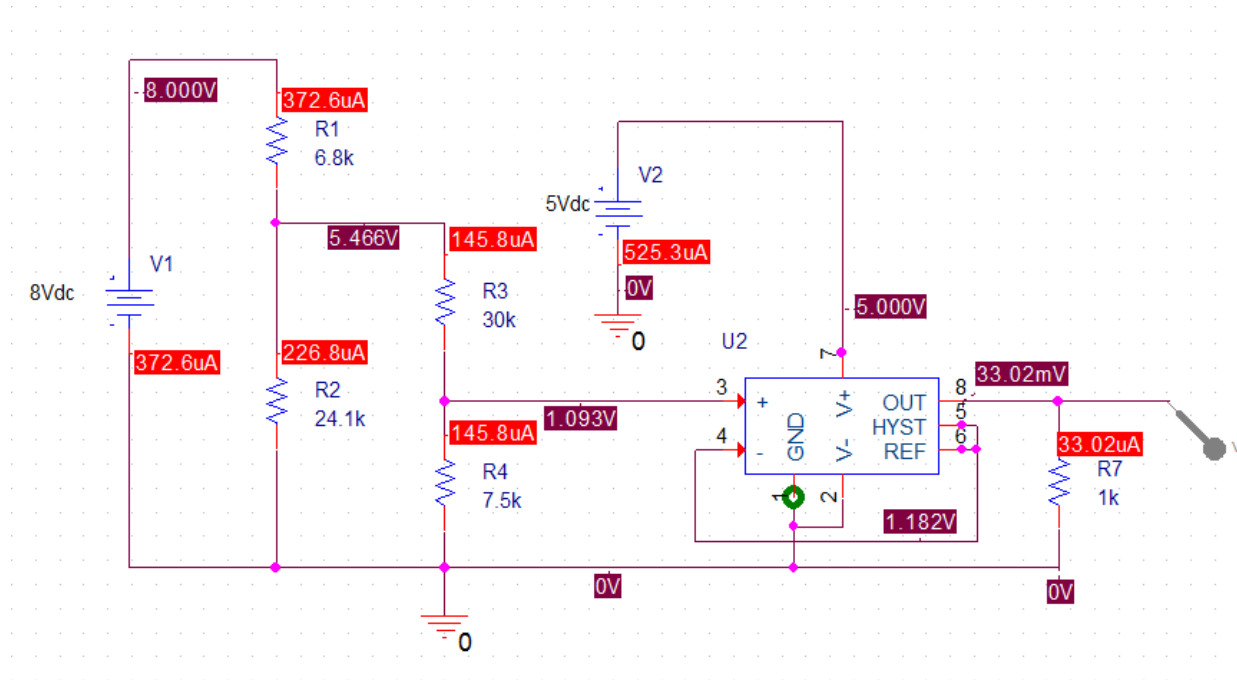
Figur 10: Orcad simulering, Fyringskrets 32V

Deretter med minimum inngangsspenning.



Figur 11: OrCad simulering, Fyringskrets 18V

«Threshold» ble også testet og det ble konkludert med at minste mulige inngangsspenning som vil resultere i at en avfiring blir registrert er 9V, ved 8V vil inngang 1 være så liten at den «registers» som lavere enn Inngang 2, og et lavt signal vil bli sendt ut av komparatoren.



Figur 18: OrCad simulering, Fyringskrets 8V

Tabell 19: T-E7

Test ID	Klasse	User story	Acceptance criterias	Risiko ID
T-E7	B	US-043101	AC-043101-01: Statemaskinen er kodet AC-043101-02: Arduino tar imot instruksjer fra UR5 (test) AC-043101-03: Dokumentasjon av AC-043101-01/02.	PRR-03 ELR-07 DAR-01
				ELR-05

Risiko tilknyttet Risikoanalyse sprint 4-2 teknisk (versjon 1.6) [3]

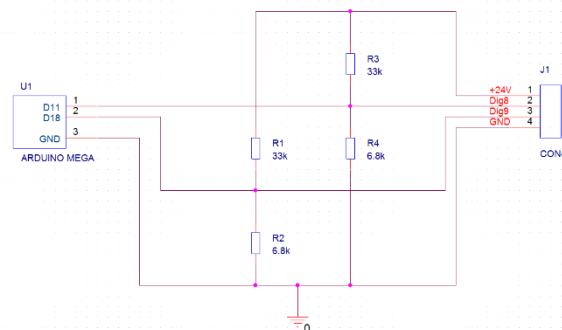
Etter å ha sett nærmere på kommunikasjonen UR5 til EE gikk vi bort fra konseptet med statemaskin, og over til en litt annen løsning, da vi så at «state machine» ville bli en veldig komplisert løsning for brukeren. Ved å bruke Arduino sin «Interrupt» benytter vi den ene digitale utgangen som en trigger, og leser verdien fra den andre digitale utgangen 4 ganger. Dette settes så sammen til en 4-bit binær verdi, noe som gir oss 16 mulige tilstander.

Interrupten er satt til å trigge hver gang den ene digitale utgangen skifter, det vil si går fra lav til høy eller motsatt. Koden på UR5 vil da i korte trekk bli, sett verdien du ønsker på LSB, skift trigger til høy. Sett verdi på neste bit, skift trigger til lav. Sett verdi på tredje bit, skift trigger til høy. Sett verdien du ønsker MSB skal ha, skift trigger til lav.

Arduinokoden lagre verdiene fortløpende, og når den har fått en verdi på alle 4 bits lager den et nummer av disse, som blir benyttet i en «switch» case til å velge hvilke bevegelse EE skal gjøre, før den resettes og er klar for å motta en nye 4 bits.

De første testene fungerte dårlig pga. mye støy å falske signaler på «trigger» pin. Dette var forårsaket av dårlig krets mellom UR5 konnektor og mikrokontroller. Kretsen manglet kobling mot GND når UR5 output var «FALSE» og fikk da mange falske utslag.

Etter en re-design av kretsen ble det gjort en ny test, denne hadde et mye bedre resultat, men dessverre ikke helt perfekt. Innimellom kunne vi oppleve falske utslag, men det var betydelig redusert fra tidligere. Vår oppfatning av årsak til dette var at kretsen var koblet opp på et «breadboard», og led av dårlig terminering på lederne.



Figur 19: Redesignet krets UR5-Arduino

Krets ble loddet og test igjen med et langt mer tilfredsstillende resultat. Støyen vi opplevde tidligere er borte. Dessverre så oppdaget vi noen problemer med timing, og at våre tidligere kodeforsøk på å redusere forstyrrelser, nå skapte feilsituasjoner for oss. Etter en liten omskriving av kode fikk vi redusert også dette, men vi står igjen med en kode som er sårbar for feil, om det skulle oppstå. Vi gjennomførte en tidstest etter dette, og hånden sto i 45 minutter og kjørte, vekslende mellom åpen hånd og lukket hånd styrt via UR5 uten at det oppsto problemer.

Tabell 20: T-E8

Test ID	Klasse	User story	Acceptance criterias	Risiko ID
T-E8	B	US-051201	AC-051201-01: Anskaffelse av Arduino ATmega 2560 AC-051201-02: Test av Arduino ATmega 2560	PRR-03 ELR-07 ELR-08
				ELR-05

Risiko tilknyttet Risikoanalyse sprint 4-2 teknisk (versjon 1.6) [3]

For å kunne motta signaler fra UR5 og gi signaler til servomotorene bruker vi en Arduino ATmega 2560 som skal utføre dette.

Arduino ATmega 2560 er anskaffet og intrigert i systemet og testet med alle servomotorer og UR5.



Figur 20: Arduino AT mega 2560

Tabell 21: T-E9

Test ID	Klasse	User story	Acceptance criterias	Risiko ID
T-E9	B	US-046401	AC-046401-01: Info er fremskaffet for å få det til AC-046401-02: Ideen til krets er notert ned, og kodet AC-046401-03: Konseptet er testet i praksis	PRR-03 ELR-07 ELR-08 DAR-01
				ELR-05

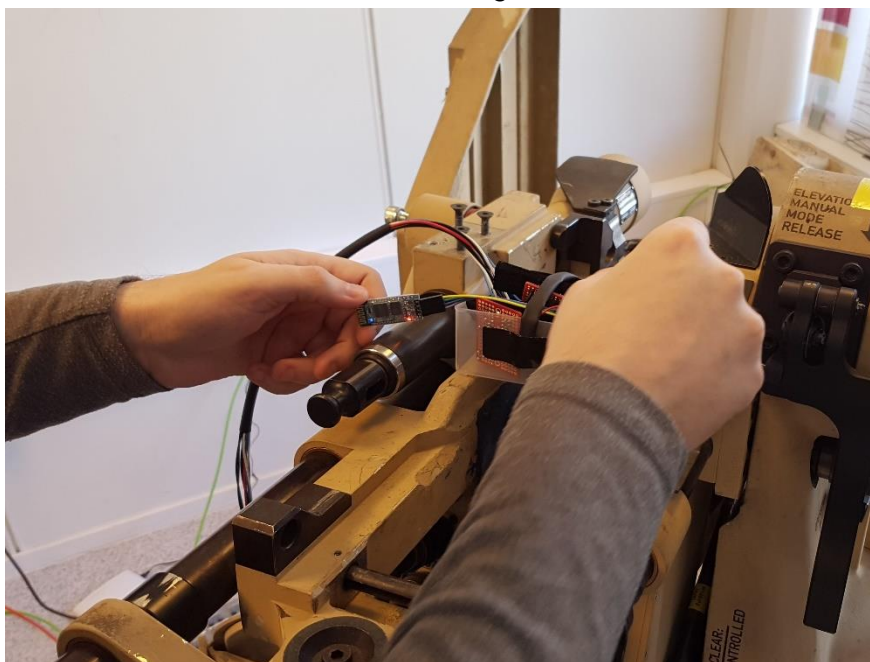
Risiko tilknyttet Risikoanalyse sprint 5 og 6 (versjon 1.7) [3]

Informasjonen vi fikk ang våpen - ID viste at WS leser status fra 4 GPIO - pinner på en FPGA for å bestemme hvilken våpen - ID som skal settes. Pinnene er koblet med «pull-up» motstander som gjør at de normalt står høye. Kode på WS tolker høyt signal på pin som en logisk lav, eller en 0. Om pin kobles mot GND vil kode på WS tolke den som en logisk høy eller 1. De fire pinnene leses og utgjør tilsammen et 4-bits binært tall mellom 0 og 15.

Vi koblet derfor 4 digitale fra Arduino til de respektive pinnene i connectoren som samsvarte med GPIO - pinnene på WS, lagde kode i Arduino IDE som tok imot verdier mellom 0-F, gjorde disse om til binær 0000-1111 og sendte de invertert ut på de digitale portene.

Testing ble gjennomført direkte på WS, der vi kan lese fra displayet på DCP hvilken våpen - ID den mottar.

Testene viste at krets og kode fungerte perfekt og at vi fritt kunne skifte våpen - ID så mange ganger vi ønsket.



Figur 12: Test på WS

WS vil ved oppstart oppgi Våpen ID 1111(0000 når invertert), eller F.

4 Software

Tabell 22: T-D1

Test ID	Klasse	User story	Acceptance criteria	Risiko ID
T-D1	B	US-040101	AC-040101-01 Software kan sette opp tilkobling med robotarm AC-040101-02 Det er programmert en listener loop på robotarm AC-040101-03 Det kjører en testsekvens når tilkobling skjer for å vise at tilkobling skjer.	PRR-03 DAR-01
				ELR-07

Risiko tilknyttet Risikoanalyse sprint 2/3 (versjon 1.3) [3]

1. Vi kobler opp systemet, med TP kabler og en ruter. Leser ut IP-adressen til robotarm og legger den inn i software. Når vi trykker på "connect" blir det forsøkt å skape en tilkobling med robotarmen. På robotarmen må det kjøre en løkke som lytter etter tilkobling fra IP- adressen til datamaskinen testprogrammet kjører på, på port 30002. Formålet med testen var å isolere deler av kildekoden fra ACT systemet slik at vi kan koble opp direkte til robotarmen uten å koble til database og våpenstasjon.
2. Når vi trykker på "connect" i testprogrammet. Responderer UR5 med å gå over i en annen del av løkken. En fargeboks i testprogrammet ved siden av "connect" knappen skifter farge fra rød til grønn.
3. Dette er testet visuelt, ved at robotarmen bevegede seg etter instruksjon fra testprogrammet.

Tabell 23: T-D2

Test ID	Klasse	User story	Acceptance criteria	Risiko ID
T-D2	C	US-040701	AC-040701-01 Software leser inn en XML fil og lagrer den i et string array	PRR-03 DAR-01

Risiko tilknyttet Risikoanalyse sprint 4-1 teknisk (versjon 1.4) [3]

Denne user storien omhandler programmets mulighet til å lese en fil av typen XML fra disk og prosessere denne på en hensiktsmessig måte slik at den kan brukes i programmet. Den valgte løsningen ble å bruke en «openFileDialog» for å åpne en fil fra disk og sende lokasjonen på denne til en funksjon som leser filen ved hjelp av de spesifiserte XML taggene. Dette gjøres ved å bruke XElement klassen i C#, denne har funksjoner som kan brukes sammen med løkker for å gå igjennom XML filen og få ut den informasjonen vi trenger. Dette ble så lagret i et «jagged array» som vi kan bruke senere i programmet.

Selve testingen av denne ble utført i to steg:

1. Det ble opprettet utskrifter til konsollen i koden hvor det blir skrevet til «arrayet» hvor vi får en utskrift av hva som skrives til de enkelte plasseringene i «arrayet» og disse ble verifisert ved at en programmerer sjekket disse opp mot forventet resultat.
2. Det ble forsøkt kjørt en sekvens på UR 5 robotarmen hvor de enkelte «array» elementene sendes til robotarmen som en streng, disse ble forstått og kjørt og regnes da som verifisert.

Tabell 24: T-D3

Test ID	Klasse	User story	Acceptance criteria	Risiko ID
T-D3	C	US-045101	AC-045101-01: Det er mulig for software og ta imot informasjon fra UR5 AC-045101-02: Informasjon fra UR5 lagres i software. AC-045101-03 : Det er tilrettelagt at dette kan sendes videre til logg	PRR-03 DAR-01

Risiko tilknyttet Risikoanalyse sprint 4-1 teknisk (versjon 1.4) [3]

Denne ble løst ved at det ble opprettet en lyttefunksjon i Robot - klassen. Denne kjøres som en tråd som kontinuerlig sjekker etter aktivitet på et «networkstream object» av klassen TcpClient. Dette vil si at den sjekker om det sendes noe fra klienten, som her er UR5 robotarmen. Denne funksjonen setter verdier eller kjører andre funksjoner på bakgrunn av hva disse beskjedene er. Informasjonen som mottas blir ikke lagret i software, men det utføres isteden funksjoner ut i fra hvilken streng som sendes fra armen,

AC-045101-01: Software kan ta imot informasjon fra UR5. Dette er testet ved at det kjøres en løkke på robotarmen starter etter den har koblet seg til ved hjelp av TCP/IP. Denne sender en "Oki" streng for å vise at den fortsatt er tilkoblet, og denne blir mottatt og testet ved å skrive til konsollen. Det er også lagt inn funksjonalitet for å lytte etter strengene "Connected" som setter en boolsk variable til true, og en "OK" som gjør at vi kan gå videre med testen.

AC-045101-02: Denne ble ikke gjort på den måten som er beskrevet i denne ACen men ble løst ved hjelp av funksjonskall og ved å sette variabler.

AC-045101-03: For å sende noe videre til log kan vi bare kalle på toLog funksjonen med den mottatte strengen som argument.

Tabell 25: T-D4

Test ID	Klasse	User story	Acceptance criteria	Risiko ID
T-D4	C	US-060501	AC-060501-01: Det er opprettet en funksjon som kan sende informasjon til sensor AC-060501-02: Funksjonen er testet og det er dokumentert i testverifikasjons dokumentet	PRR-03 DAR-01

Risiko tilknyttet Risikoanalyse sprint 5 og 6 (versjon 1.7) [3]

I sensorklassen er det laget en funksjon som kan sende en string til sensoren. Sensoren er laget så den resettes når den mottar en «R». Funksjonen bruker seriellport objektet og funksjonen Write til å sende over en R'en.

```
public void resetSensor()
{
    blueTooth.Write("R");
}
```

Figur 13: resetsensor

Sensoren sender tilbake en 0 å verifiserer med dette at fyringstelleren er resatt. Vi vi bekrefter hendelsen i loggen ved å håndtere 0 en på følgende måte:

```
if (indata[0] == '0')
{
    InData = "Sensor has been reset, counting from 0";
}
```

Figur 14: Fra sensor til loggfil

InData skrives så til loggfilen.

Tabell 26: T-D5

Test ID	Klasse	User story	Acceptance criteria	Risiko ID
T-D5	B	US-042003	AC-042003-01: Det er opprettet funksjonalitet for å skrive disse hendelsene til loggen. AC-042003-02: Brukeren kan se forventet output i loggen i programmet.	PRR-03 DAR-01
				PRR-09

Risiko tilknyttet Risikoanalyse sprint 5 og 6 (versjon 1.7) [3]

Ved opprettelse av log funksjonalitet ble det opprettet en «logEntry» funksjon. Den kan kalles fra andre steder i koden med en streng som argument. Dette blir skrevet til loggen som vises til brukeren kontinuerlig.

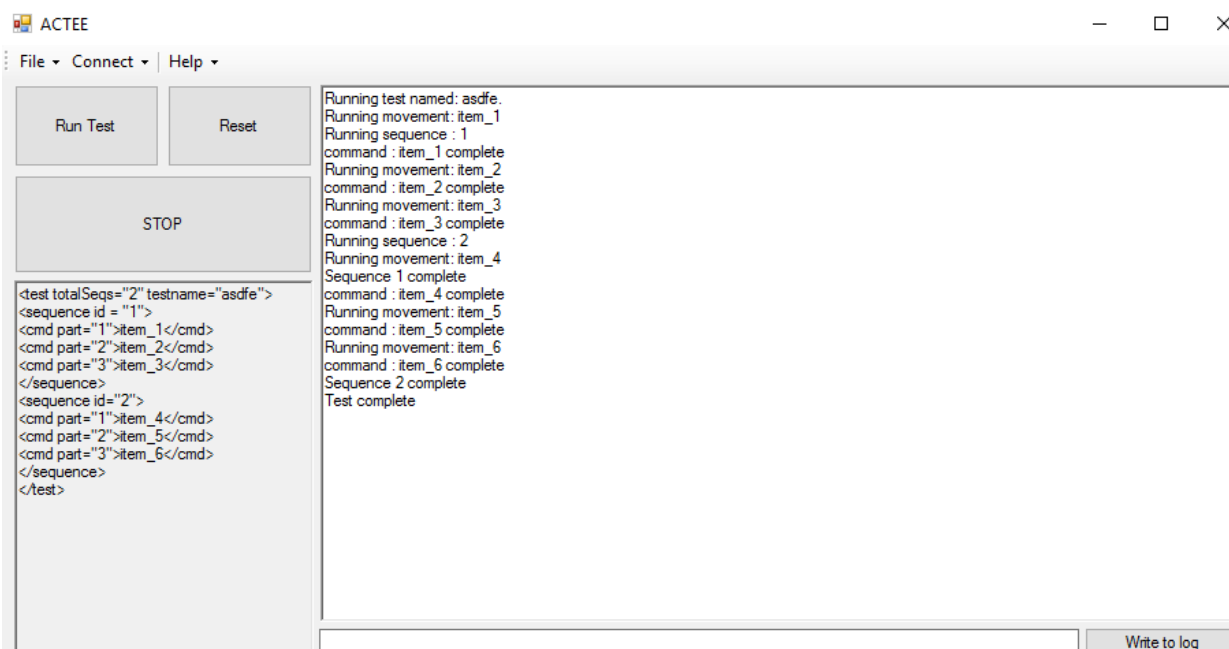
AC-042003-01: I løkken som kjører gjennom sekvensen som brukeren velger blir det kjørt kall til «logEntry» funksjonen som skriver det som er nødvendig til loggen.

```
private void testRunner()
{
    //write the testname to the log
    log.logEntry("Running test named: " + testName + ".");

    for (int i = 0; i < robotCommandArray.Length; i++)
    {
        //write current sequence number to log
        log.logEntry("Running sequence : " + (i+1) );
    }
}
```

Figur 24: Kode for å skrive til logg

AC-042003-02: Ved å kjøre en testsekvens vil brukeren få opp informasjon om hvilken sekvens, bevegelse og test som kjøres.



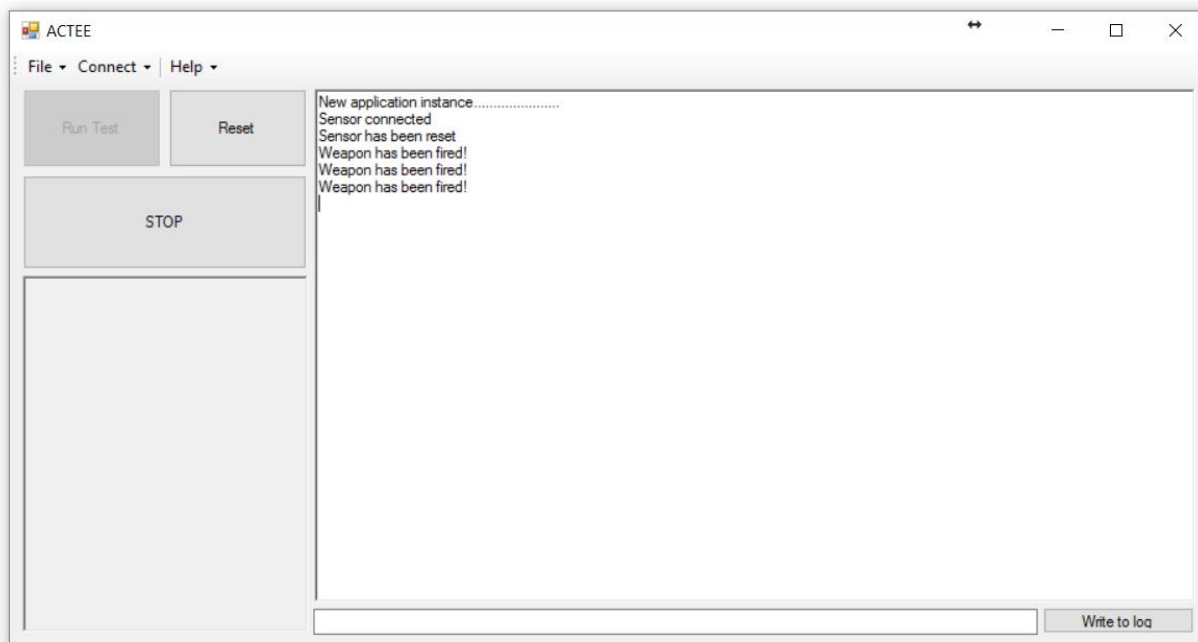
Figur 15: Loggen under kjøring av test

Tabell 27: T-D6

Test ID	Klasse	User story	Acceptance criteria	Risiko ID
T-D6	B	US-040601	AC-040601-01: Software tar imot en variabel som bestemmer om fyring skjer eller ikke AC-040601-02: Software kan vise om fyring skjer i GUI	PRR-03 DAR-01
				PRR-09

Risiko tilknyttet Risikoanalyse sprint 5 og 6 (versjon 1.7) [3]

AC-040601-01: For hver fyring registrert på våpenstasjonen sendes en F fra sensoren til testmaskinen. F'en sendes som en streng med Serial kommunikasjon over Bluetooth. F'en fanges opp og oversettes av programvaren. Etter oppstart og tilkobling til sensoren på våpenstasjonen, skrives hendelser rett i loggen. Etter tre avfyringer ser loggen slik ut:



Figur 16: Våpentest

Dette er ønsket utfall og begge AC er verifisert.

Tabell 28: T-D7

Test ID	Klasse	User story	Acceptance criteria	Risiko ID
T-D7	B	US-040301	AC-040301-01: Software kan sende innlastet XML fil med instruksjoner til ur 5 robotarm og de blir forstått og fulgt. AC-040301-02: Det er lagt til funksjonalitet i programmet slik at brukeren kan lage egne testsekvenser ut ifra de mulige funksjonene på DCP og CG AC-040301-03: Det er opprettet en funksjon for hver bevegelsessekvens som er mulig på UR 5	PRR-03 DAR-01
				PRR-09

Risiko tilknyttet Risikoanalyse sprint 5 og 6 (versjon 1.7) [3]

AC-040301-01: Software sender instruksjonene som ligger i innlastet XML fil med instruksjoner til UR 5 robotarm ved at det itereres igjennom XML filen ved hjelp av XElement funksjonaliteten i C# og noen løkker. Det sendes en instruksjon, så venter programmet på en OK streng tilbake fra robotarmen før den går videre.

```
//send message to robot with the command we want it to execute
robot.sendMessageRobot(robotCommandArray[i][j]);

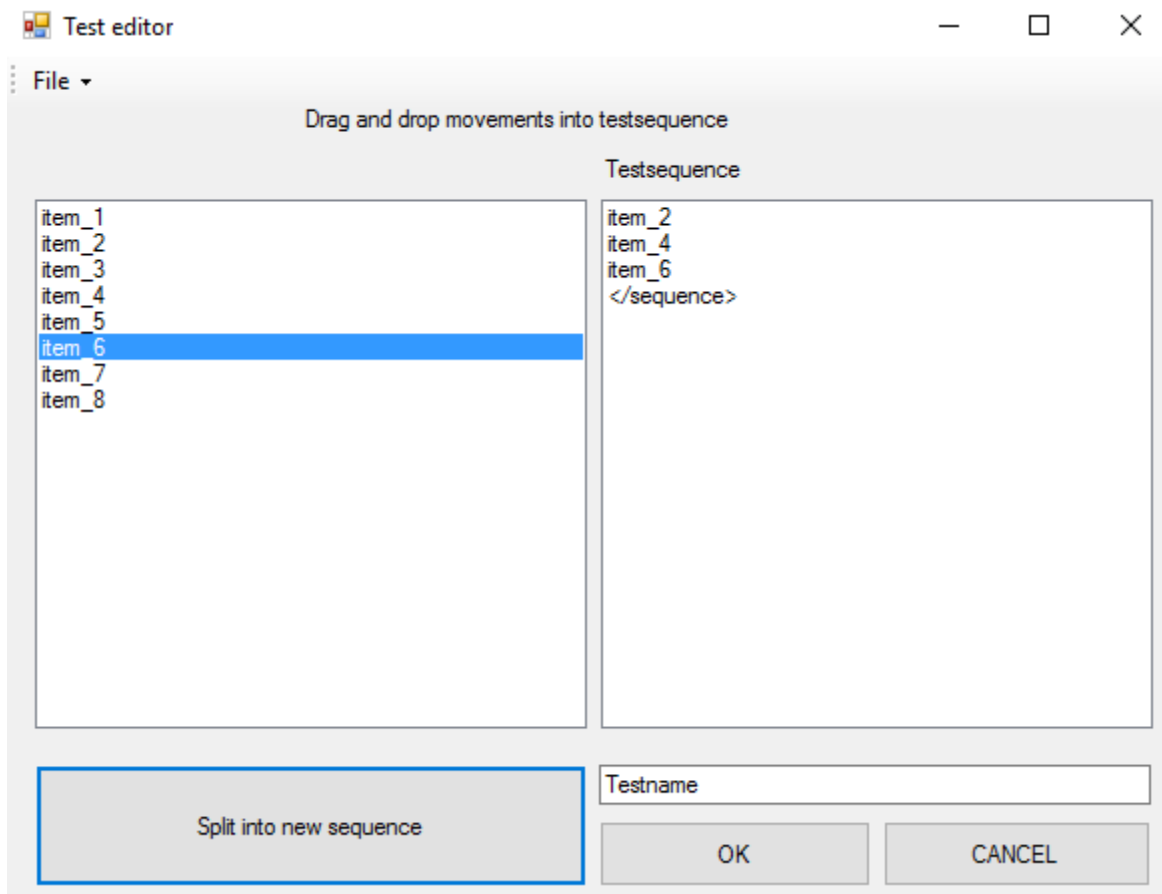
//the thread will spin until the nextCommand is set to true
System.Threading.SpinWait.SpinUntil(() => robot.nextCommand);
```

Figur 27: Sending av instruksjoner

Dette er de to linjene som direkte sender en streng til roboten og venter på å få lov til å sende neste streng.

Denne er testet ved at vi har kjørt bevegelsessekvenser på armen fra software med visuell godkjenning.

AC-040301-02: Det er lagt til funksjonalitet i programmet hvor brukeren kan lage egne sekvenser. Dette gjøres ved at brukeren får mulighet til å «drag and droppe» fra to liste bokser og kan lagre denne sekvensen som en XML fil som programmet kan kjøre senere.



Figur 28: Test editor

Disse listene fungerer som de skal og har blitt testet ved å lagre sekvensene som lages, og kjøre den resulterende XML filen senere.

Listene er på nåværende tidspunkt populært med eksempel navn da det ikke var tid igjen i Sprint 4 til å utvikle programmet videre etter gripeverktøyet var ferdig montert. All funksjonaliteten ligger til rette og det eneste som må gjøres er å endre navnet på elementene i listen til venstre til navnet på bevegelsene som ligger lagret i programmet vårt på UR 5 når vi får opprettet disse.

AC-040301-03: Denne ble løst ved at vi sender en streng istedenfor å opprette egne funksjoner. Funksjoner ville hatt bedre muligheter til å gjøre verifikasjon på hver enkelt bevegelse vi kaller på, men dette ble ikke mulig å gjennomføre da det ble for lite tid igjen etter fullføring av gripeverktøyet. Ingen test gjort.

Tabell 29: T-D8

Test ID	Klasse	User story	Acceptance criteria	Risiko ID
T-D8	C	US-045201	AC-045201-01: «Socket listener loop» med bevegelsessekvenser som inkluderer gripeverktøy er opprettet	PRR-03 DAR-01

Risiko tilknyttet Risikoanalyse sprint 4-2 teknisk (versjon 1.6) [3]

Denne AC er i to steg:

1. Det er opprettet ett program på UR 5 som først prøver å koble til et nettverkspunkt definert med IP adresse og portnummer.

Dette er testet ved å forsøke å opprette en tilkobling til testprogrammet, hvis det er tilkobling så skal programmet på UR 5 sende strengen «Connected» til testprogrammet. Dette mottas av programmet.

2. Bevegelsessekvenser som inkluderer gripeverktøyet er ikke opprettet på grunn av tidsmangel.

Tabell 30: T-D9

Test ID	Klasse	User story	Acceptance criteria	Risiko ID
T-D9	C	US-060201	AC-060201: Software kan motta nødvendig data fra UR 5	PRR-03 DAR-01

Risiko tilknyttet Risikoanalyse sprint 4-2 teknisk (versjon 1.6) [3]

Denne er løst ved at det sendes strenger fra UR 5 på punkter i programmet hvor det er hensiktsmessig å kommunisere med testprogrammet. Den sier ifra når den er «Connected», og den sender en «OK» streng når den er ferdig med en valgt bevegelse.

I testprogrammet kjøres det en tråd som lytter etter informasjon fra klienten. Dette gjøres ved hjelp av TCP Client klassen i C#.


```
private void receiveDataRobot()  
{  
  
    while (true)  
    {  
        stream = client.GetStream();  
        //if the networkstream supports reading, do  
        if (stream.CanRead)  
        {  
            byte[] myReadBuffer = new byte[1024];  
            StringBuilder myCompleteMessage = new StringBuilder();  

```

Figur 29: Lese data fra robot.

Det opprettes ett «NetworkStream» objekt av client objektet, og vi lager en streng av data vi mottar fra UR 5 ved hjelp av «StringBuilder».

```
//robot sends ok message after completing a movement  
if (myCompleteMessage.ToString() == "OK")  
{  
    //if robot send OK, go to next movement  
    nextCommand = true;  
}  
//robot sends connected when TCP/IP connection is complete  
if (myCompleteMessage.ToString() == "Connected")  
{  
    //variable telling the GUI to show connection status  
    Connected = true;  
}
```

Figur 17: Funksjoner ved mottak av strenger

Hvis de mottatte beskjedene er «OK» eller «Connected» setter vi disse variable til true.

Dette er testet ved at vi printet de mottatte beskjedene til konsollen og det som mottas er det som blir sendt.

Tabell 31: T-D10

Test ID	Klasse	User story	Acceptance criteria	Risiko ID
T-D10	C	US-042001 US-042002	AC-042001-01: Loggsystem er opprettet og testet AC-042002-01: Det er satt opp en funksjon som skriver til fil	PRR-03 DAR-01

Risiko tilknyttet Risikoanalyse sprint 4-2 teknisk (versjon 1.6) [3]

Loggsystemet fungerer slik at det skriver alt som skal i loggen til en midlertidig loggfil som heter log.txt. Hver gang log.txt endrer seg skrives den til tekstboksen i brukergrensesnittet. Dette gjøres ved bruk av en «eventlistener» som overvåker filen. Om brukeren ønsker å ta vare på loggen kan den lagres som noe annet med «Save log» under «File» menyen.

```

public const string LogFile = @"log.txt";
//adding strings to the queue, to ensure that the file is available
9 references | Thomas Wegener, 8 days ago | 1 author, 1 change
public void logEntry(string msg)
{
    // To enqueue the write, we use the built in threadpool.
    ThreadPool.QueueUserWorkItem(WriteToFile, msg);
}
// the lock
private static object writeLock = new object();

1 reference | Thomas Wegener, 6 days ago | 1 author, 3 changes
public static void WriteToFile(object msg)
{
    lock (writeLock)
    {
        //using the lock object we now can ensure that we are the only thread accessing the file
        if ((string)msg == "reset") //the reset string flushes the logfile.
        {
            StreamWriter strm = File.CreateText(LogFile);
            strm.Flush();
            strm.Close();
        }
        else
        {
            using (var writer = File.AppendText(LogFile)) //adds a string to a new line in the log
            {
                writer.WriteLine((string)msg); //writes the string.
            }
        }
    }
}

```

Figur 18: Kode for å skrive til og resette log.txt

Kort fortalt kan flere tråder og funksjoner sende «strings» til log med «logEntry». Den putter disse i en kø og venter på tilgang til log.txt. Så låses filen for andre mens innholdet i køen skrives til loggfilen og programvinduet oppdateres.


```
public void updateView()
{
    //We make a watcher keep an eye out for changes in the temporary logfile,
    _watcher = new FileSystemWatcher(Application.StartupPath, Path.GetFileName(Log.LogFile));
    //The watcher looks for writes or rewrites make to the file.
    _watcher.NotifyFilter = NotifyFilters.LastAccess | NotifyFilters.LastWrite
    | NotifyFilters.FileName | NotifyFilters.DirectoryName;

    //When the file is changed we try to read it into txtLogOutput (the GUI)
    _watcher.Changed += (s, e) =>
    {
        try
        {
            //Using filestream handles read and write accsett for the file. We already write to it from the log class.
            using (FileStream fileStream = new FileStream(
                Path.GetFileName(Log.LogFile),
                FileMode.Open,
                FileAccess.Read,
                FileShare.ReadWrite))
            { //using streamreader allows us to grab the content of the file,
                using (StreamReader streamReader = new StreamReader(fileStream))
                {
                    //To sync the output with the view thread we invoke on readtoend, and "forcefeeds" logText to gui with a lambda expression.
                    var logText = streamReader.ReadToEnd();
                    if (!txtLogOutput.IsDisposed)
                        Invoke(new Action(() => txtLogOutput.Text = logText ));
                }
            }
        }
        catch (InvalidOperationException) //If it for some reason doesn't work it opens a messagebox.
        {
            //Show a message box with the exception message if a exception happens
            MessageBox.Show("Error loading log file.");
        }
    };
    _watcher.EnableRaisingEvents = true;
}
//Function for writing to the log, this gets called wherever there is changes in log.txt
```

Figur 19: Koden som lytter etter endringer i loggfilen, og leser loggen til GUI

Om reset skrives alene til loggen som en separat string så overskrives loggfilen. Nødvendig testing er utført med loggfilen og testprogrammet.

5 System

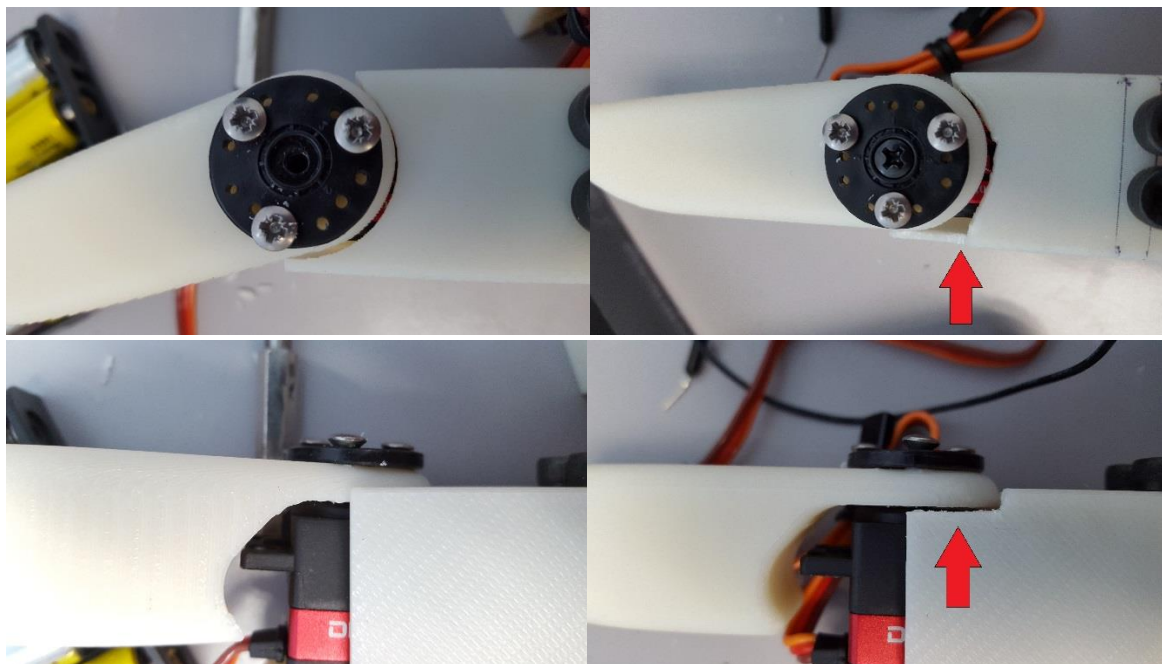
I denne delen av dokumentet tar vi for oss testing av sub - systemer og systemet i sin helhet.

Tabell 32: T-SY1

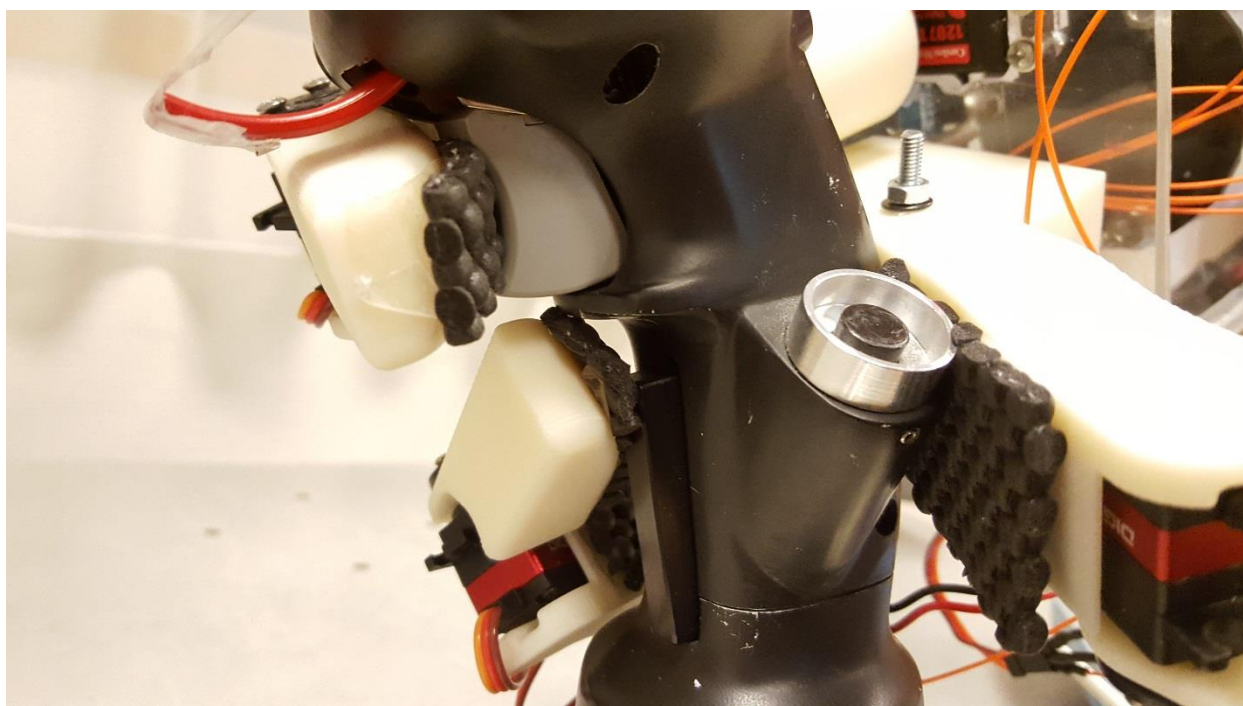
Test ID	Klasse	User story	Acceptance criterias	Risiko ID
T-SY1	A	US-010701	AC-010701-03: Første del av EE design printes AC-010701-04: Montering og integrering av servomotorer sammen med printet EE. AC-010701-05: Dokumentering av AC-010701-01/02 AC-010701-06: Test av system sammen (EE og servomotorer) AC-010701-07: Dokumentasjon av AC-010701-04.	PER-02 PER-06 PRR-03 PRR-04 PRR-12 ELR-07 ELR-08 MAR-19 MAR-20
				PER-01 PRR-07 PRR-10 ELR-01

Risiko tilknyttet Risikoanalyse sprint 4-1 teknisk (versjon 1.4) [3]

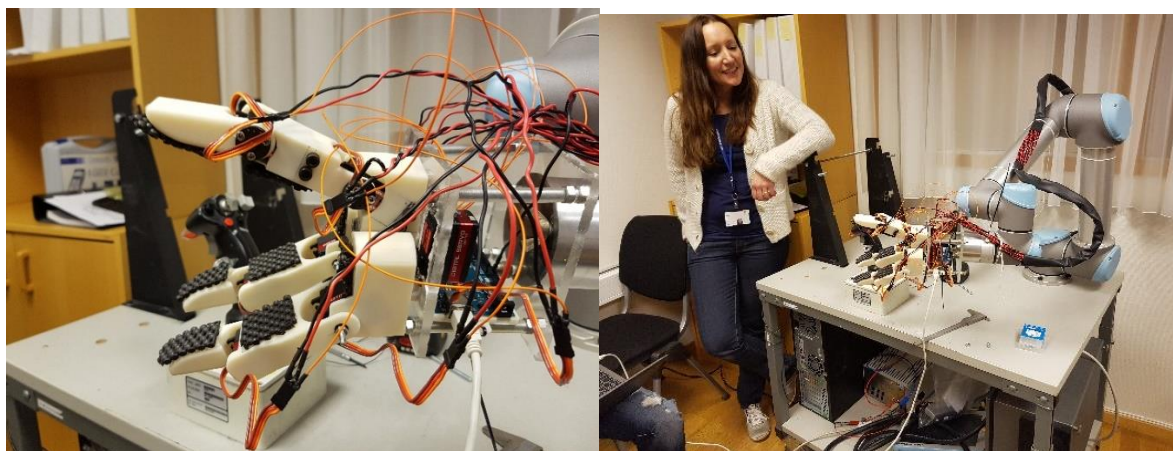
1. Første utkast er 3D – printet i ABS plast, og er grunnlaget for vårt videre arbeid. Utgangspunktet vårt er ytterligere tilpasset, og første *funksjonelle* prototype er 3D – printet. Delene er montert med servomotorene, og er klare for testing, og er testkjørt separat før montering på systemet. Vi forventer ytterligere modifisering før siste presentasjon. Dokumentasjon i form av tekst, bilder og video fortløpende underveis.
 2. Det skal testes for compatible grensesnitt i systemet. For å få til dette på best mulig måte, tester vi sub – systemer først. Fingerledd og servoer fungerer slik vi har sett for oss, med riktig størrelse på nødvendig kraft.
 3. Montering EE: Mye småtilpasninger; skruer måtte tilpasses, boret til de fleste hull. Det er brukt mye tid på dette som i utgangspunktet kunne vært printet klart eller planlagt bedre for. Dette viker klart fra først tiltenkt prosess, og vi tar med oss lærdom videre.
- Vel montert fungerer EE som tenkt, men små tilpasninger viser seg fordelaktig:
- En smal fingertupp tiltenkt knapper - laget
 - Smalere ledd da vi endret innfestning til servo, planlagt men ikke printet
 - En buet tommelfinger for bedre støttegrep rundt CG – planlagt men ikke printet
 - Tommelfinger noe kortere med kun ett ledd, og smal tupp for enkel betjening av knapper. Er planlagt men ikke printet.



Figur 20: Modifisering slik at ikke ytreledd interfererer med mellomledd.



Figur 34: Test av Servomotorene mot CG



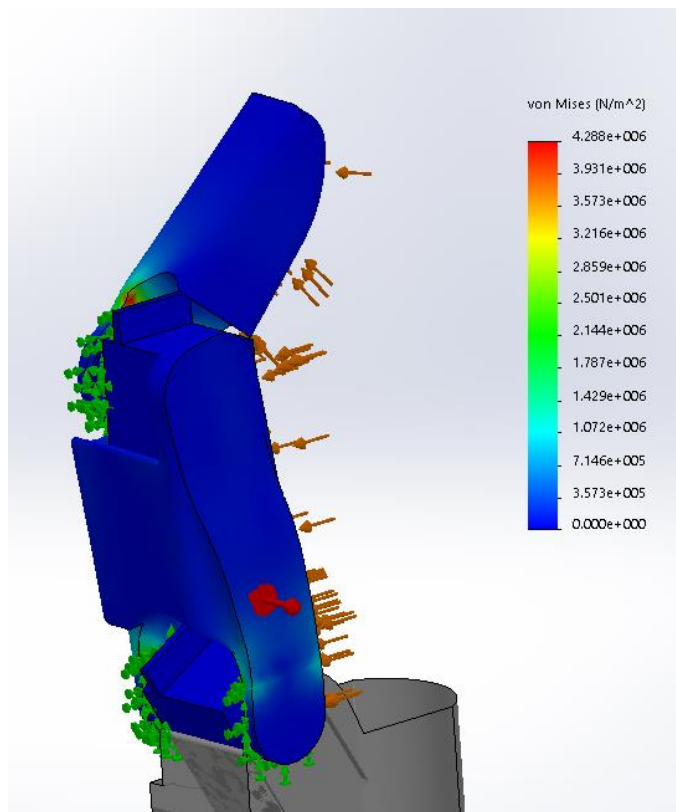
Figur 21: Aller første test av EE 3. mai

Tabell 33: T-SY2

Test ID	Klasse	User story	Acceptance criterias	Risiko ID
T-SY2	A	US-044002 US-044005	AC-044002-01: Beregninger er gjort slik at det skal er nok styrke i utforming av EE AC-044002-02: Servomotorer er programmert til å utføre sekvensene for å aktivere palm - knapp. AC-044002-03: Dokumentasjon av AC-044002-01/02. AC-044005-01: Beregninger er gjort slik at det skal er nok styrke i utforming av EE AC-044005-02: Servomotorer er programmert til å utføre sekvensene for å aktivere triggerknapp. AC-044005-03: Dokumentasjon av AC-044005-01/02.	PRR-09
				PRR-03
				PRR-04
				ELR-07
				ELR-08
				MAR-15
				PRR-07

Risiko tilknyttet Risikoanalyse sprint 4 -2 teknisk (versjon 1.6) [3]

Styrkeberegninger er foretatt i FEM, resultater herfra foreligger i dokument 6 – teknologidokument EE.



Figur 22: FEM analyse, Von Mises stress



Figur 23: Servomotor

Figur 24: Finger med servoer montert

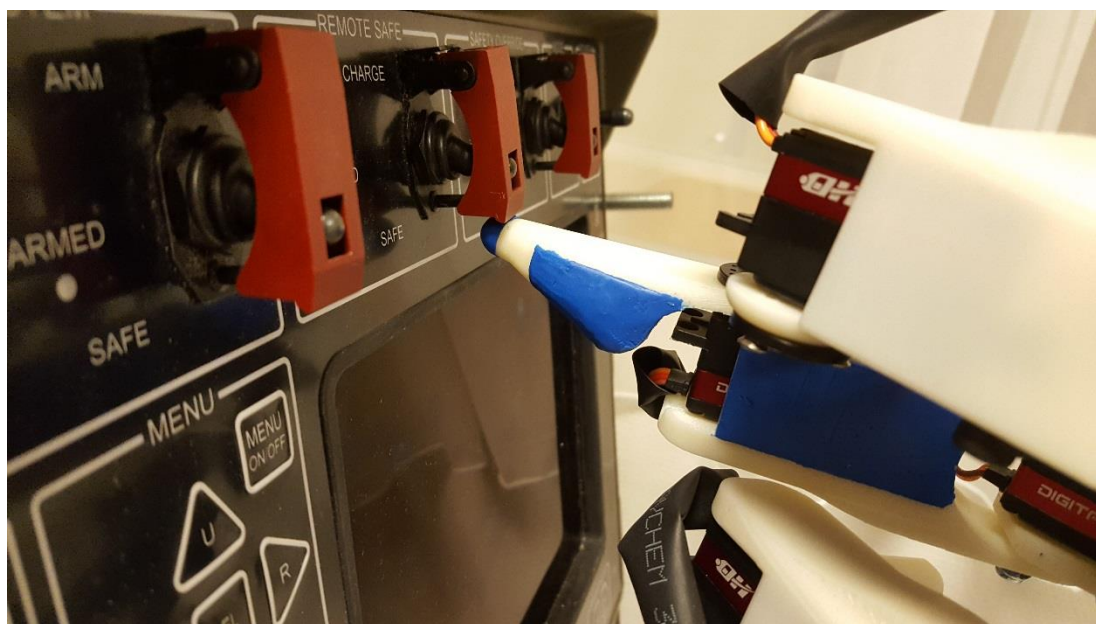


Tabell 34: T-SY3

Test ID	Klasse	User story	Acceptance criterias	Risiko ID
T-SY3	A	US-044102	AC-044102-01: Beregninger er gjort slik at det skal er nok styrke i utforming av EE AC-044102-02: Programmering av EE skal være ferdig slik at Software kan programmere UR5 til å gjøre sekvens. AC-044102-03: Software programmerer sekvens for bevegelse av UR5 slik at EE utfører test for betjening av vippebrytere til DCP AC-044102-04: Dokumentasjon av AC044102-01/02/03	PRR-09
				PRR-03 PRR-04 ELR-07 ELR-08 MAR-19
				PRR-07 PRR-10 ELR-01

Risiko tilknyttet Risikoanalyse sprint 4 -2 teknisk (versjon 1.6) [3]

Kommentar: Denne testen er kun kjørt et fåtall ganger, med positivt resultat. Vi vil fortsette testingen noe i sprinten etter innlevert dokumentasjon, hvor vi forbereder siste presentasjon.



Figur 26: Ny fingertupp kan utføre å løfte vippebryter



Figur 41: Fingertupp vipper indre bryter.



Figur 42: Fingertupp lukker vippebryter.



Figur 27: Indre bryter lukker seg sammen med ytre vippebryter.

Tabell 35: T-SY4

Test ID	Klasse	User story	Acceptance criterias	Risiko ID
T-SY4	A	US-044103	AC-044103-01: Beregninger er gjort slik at det skal er nok styrke i utforming av EE AC-044103-02: Programmering av EE skal være ferdig slik at Software kan programmere UR5 til å gjøre sekvens. AC-044103-03: Software programmerer sekvens for bevegelse av UR5 slik at EE utfører test for betjening av Knapper til DCP AC-044103-04: Dokumentasjon av AC044103-01/02/03	PRR-09
				PRR-03 PRR-04 ELR-07 ELR-08 MAR-19
				PRR-07 PRR-10 ELR-01

Risiko tilknyttet Risikoanalyse sprint 4 -2 teknisk (versjon 1.6) [3]

Kommentar: Denne testen er kun kjørt et fåtall ganger, med positivt resultat. Vi vil fortsette testingen noe i sprinten etter innlevert dokumentasjon, hvor vi forbereder siste presentasjon.



Figur 28: Ny fingertupp til EE fungerer som den skal og klarer å trykke inn knapper på DCP

Tabell 36: T-SY5

Test ID	Klasse	User story	Acceptance criterias	Risiko ID
T-SY6	A	US-010101	AC-010101-01: EE skal kunne holde inn palm knapp, trykke trigger knapp, og bevege cg, samt kunne vippe bryter og trykke på knapper til DCP. Posisjon skal kartlegges og dokumenteres. AC-010101-02: Dokumenter AC-010101-01.	PRR-09 MAR-21
				PRR-03 PRR-04 ELR-01 ELR-07 ELR-08 DAR-01 DAR-05 MAR-12
				PRR-10

Risiko tilknyttet Risikoanalyse sprint 4 -2 teknisk (versjon 1.6) [3]

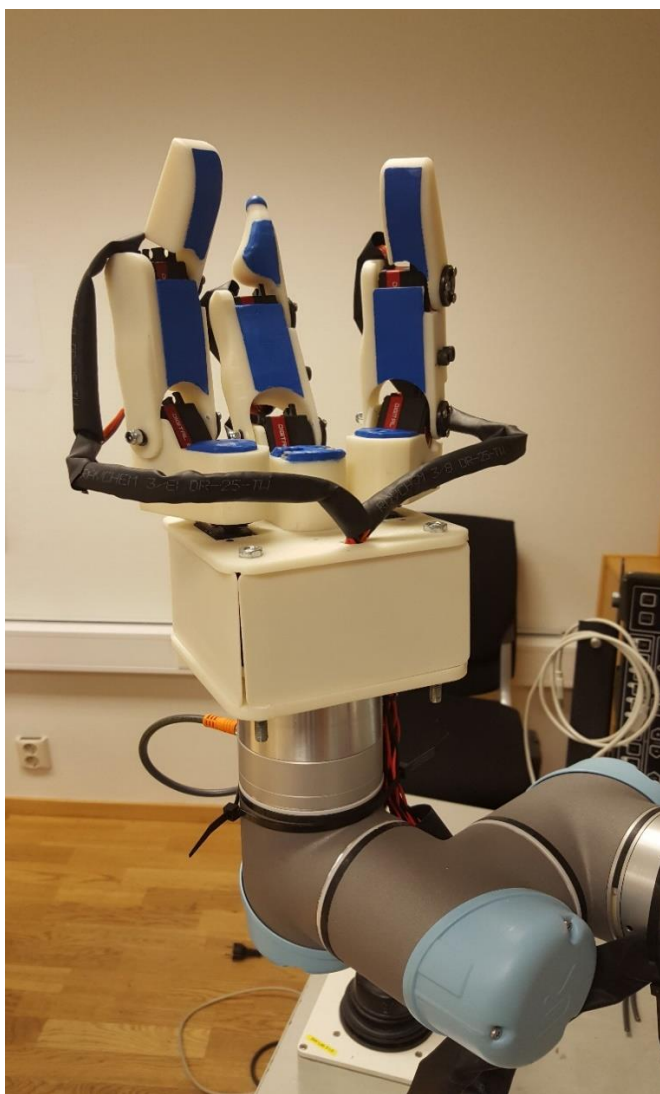
Ved test av EE mot CG og DCP ble det gjort en del finish på EE slik at det ikke skulle være noen monteringer/forandringer på EE i etterkant av lagrede sekvensene og posisjonering.

EE fikk et belegg med Sugru slik at det så litt mer elegant ut og ga et litt bedre grep enn rein overflate med PCB

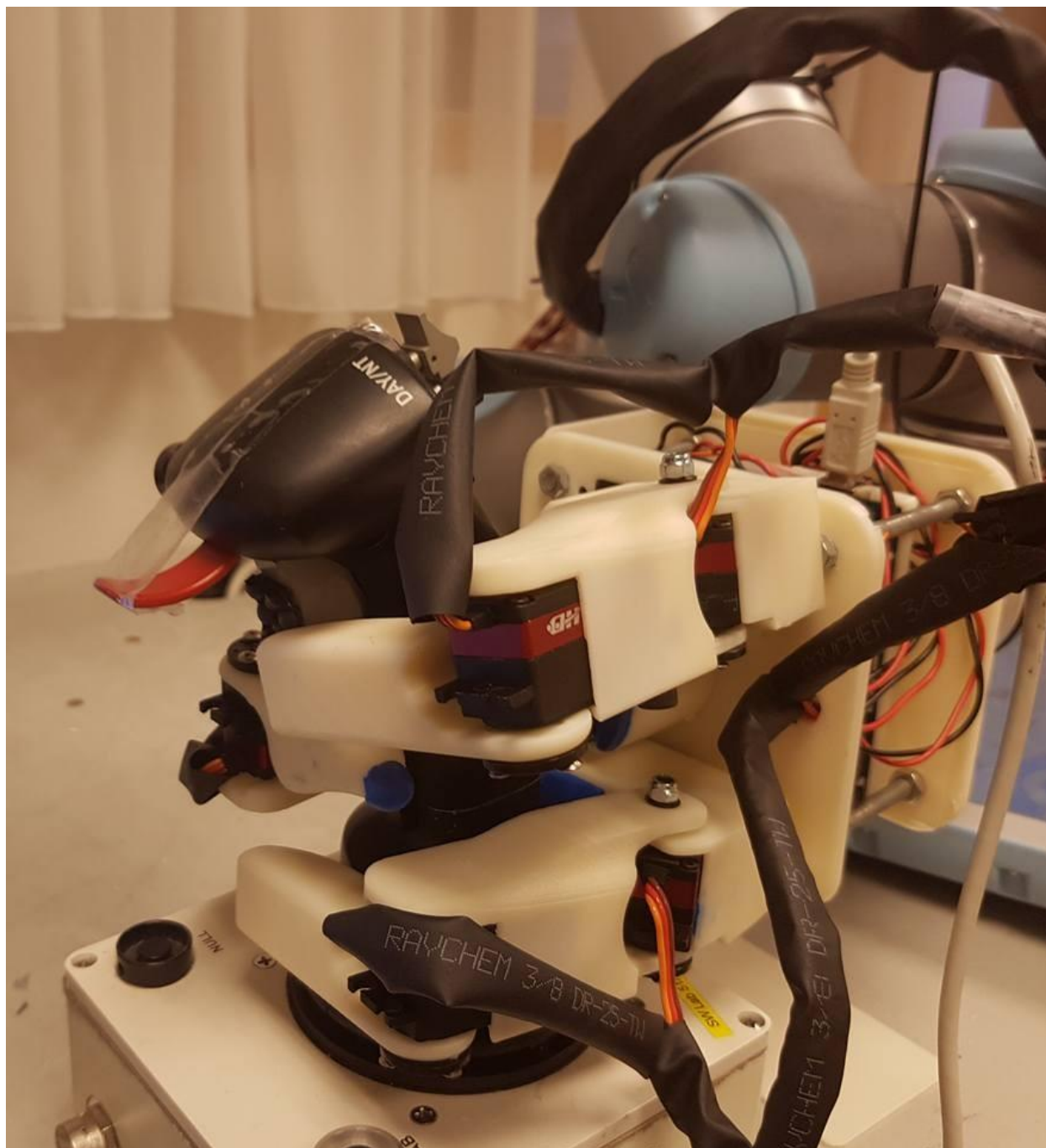
Det ble gjort en jobb med å kunne skjule kabler og en smalere fingertupp ble montert på for trykking på DCP.

EE kan gripe rundt CG og holde inne trigger knapp og palm knapp.

EE interferer med UR5 i kabinettet og vi trenger å kunne rotere CG slik at UR5 får en bedre posisjon på CG.



Figur 29: EE har fått på sugru og fått kabler gjemt.



Figur 30: EE griper rundt CG og trykker palm og trigger knapp.

6 Konklusjon

Etter endt teknisk fase sitter vi igjen med en god del testresultater, og mye har gått som forventet. Vi ser at mer tid til test ville vært en klar fordel, og planlegger å forlenge testingen noe etter dokumentasjonen er innlevert. I perioden mellom innlevering og siste presentasjon ser vi for oss at test og forberedelse lar seg kombinere.

Systemtestene er de testene med høyest risiko og dermed vektning. Det er disse testene vi har et ønske om å ta et steg videre.

Referanser

[1] Arduino. 2016. *Arduino Software (IDE)*.

Tilgjengelig: <https://www.arduino.cc/en/Guide/Environment>

Sist sjekket: 16.03.2016

[2] *Universal Robots user manual*

Tilgjengelig: [Universal Robots](#)

Sist sjekket: 16.03.2016

[3] **Risikoanalyse Versjon 2.0 (18.05.16)**

[4] Universal Robots AS. (2015). *UR5 - roboter*.

Tilgjengelig fra:

<http://www.universal-robots.com/no/produkter/ur5-robot/>

ACTEE User Manual

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROJECTNAME	Automated CROWS testing end effector
EMPLOYER	Kongsberg Protech Systems
INSTITUTION	University College of SouthEast Norway
DOCUMENT NAME	ACTEE User Manual
DOCUMENT STATUS	v 1.0/ 19. May 2016
AUTHOR	Thomas Wegener
PAGECOUNT	42



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



i. Abstract

This document is a user manual for the gripping tool and the control software made the bachelor group ACTEE. This document will show the reader how to assemble and operate the gripping tool, and how to operate the software.

ii. Table of contents

i.	Abstract	2
ii.	Table of contents	3
iii.	Table of tables	4
iv.	Table of figures.....	4
v.	Document change report.....	5
vi.	Acronyms	5
1	Introduction	6
1.1	Intended for.....	6
1.2	Purpose	6
1.3	How to use this document.....	6
1.4	Problem reporting	6
2	System overview.....	7
2.1	Hardware dependencies	7
2.2	Software dependencies	7
3	End Effector Assembly Guide	7
3.1	Part Overview	7
3.2	Assembly instructions finger	8
3.3	Assembly instructions cabinet.....	18
3.4	Mounting fingers to the cabinet	21
3.5	Mounting End Effector to the UR5 Robot arm	23
3.6	Attaching cables	25
4	Software Tutorial.....	29
4.1	Software installation.....	29
4.2	Software removal	32
4.3	Client overview	32
4.4	Connecting to the UR5.....	33
4.5	Connecting to the Weapon Station Sensor	35
4.6	Running a test.....	36
4.7	Creating a test using the Test editor	38

4.8 Saving the log file	41
Sources.....	42

iii. Table of tables

Table 1 Document change report	5
Table 2 - Acronyms	6
Table 3 - Part overview	7

iv. Table of figures

Figure 1 – Step 1 in assembly of the finger	9
Figure 2 – Step 2	10
Figure 3 – Step 3	11
Figure 4 - Arduino ATmega	12
Figure 5 - USB Cable	12
Figure 6 – Uploading the script to Arduino	13
Figure 7 - Signal Cables.....	13
Figure 8 - Arduino ATmega	14
Figure 9 - Servomotor	14
Figure 10 - Serial Monitor	15
Figure 11 - Finger servo assembly	15
Figure 12 - Servomotor in central joint	16
Figure 13: Step 10	17
Figure 14 Cabinet panel top	18
Figure 15 – Centration of the servos	19
Figure 16: Tightening thread bolts.....	20
Figure 17: Attaching the lower cabinet panel.....	21
Figure 18: Attaching servo wheel	22
Figure 19: Attaching finger to the servo on top cabinet panel	22
Figure 20: Attaching middle finger.....	23
Figure 21: Attaching interface to UR 5	24
Figure 22: Attaching thread bolts to interface	24
Figure 23: Tightening EE to interface	25
Figure 24: Circuit diagram	26
Figure 25: Power cables	27
Figure 26: Signal Cables.....	27
Figure 27: Power cables	28
Figure 28: Dividing cables back	29

Figure 29: Dividing cables front.....	29
Figure 30 - Locate the installation file	30
Figure 31 - Software Installation.....	31
Figure 32 - Destination directory	31
Figure 33 - Client overview.....	32
Figure 34: Socket open UR 5 program.....	34
Figure 35: Robot connection status.....	35
Figure 36: Wss connection.....	36
Figure 37: Load test file.....	37
Figure 38: Test information	37
Figure 39: Test output	38
Figure 40: Launching the test editor	39
Figure 41: Test editor GUI.....	40
Figure 42: Building a sequence	41

v. Document change report

Table 1 Document change report

v 0.1 10.05.16	Thomas: Document Outlines
v 0.2 11.05.16	Thomas: Writing of Chapter 1 and 2.
v 0.3 12.05.16	Thomas: Writing of Chapter 4.
v 0.4 14.05.16	Espen: Adding to chapter 4.
v 0.5 15.05.16	Carl Martin: Adding chapter 3.
v 1.0 19.05.16	Espen: Proofreading and completion of the document.

vi. Acronyms

Table 2 - Acronyms

Acronyms	Explanation
ACT	Automated CROWS Testing
ACTEE	Automated CROWS Testing End Effector
EE	End Effector, gripping tool
CROWS	Common Remotely Operated Weapon Station
UR5	Universal Robots 5 robot arm

1 Introduction

1.1 Intended for

This document is intended for the following users of the ACTEE system:

- The system administrators
- The project administrators
- The test operators
- The hardware providers and mechanics

This document applies to the ACTEE system, build 1.0. Delivered in May 2016.

1.2 Purpose

The purpose of this document is to assist users in assembly, installation and operation of the ACTEE system.

1.3 How to use this document

- Chapter 2 gives an overview of the complete ACTEE system.
- Chapter 3 contains a guide for mechanics. Describing parts, cables and assembly in detail.
- Chapter 4 contains a software usage tutorial, which enable users to get started quickly.

1.4 Problem reporting

Since the ACTEE team will be dissolved after completion of the ACTEE project, the issue of problem reporting is left to Kongsberg Protech Systems.

2 System overview

2.1 Hardware dependencies

- **UR5 robot** - The system requires a Universal Robots model 5 robot. The robot needs to be running a listener loop and containing a set of pre-programmed movements.
- **Test machine** - The computer running the software.
- **Local area network** - A network connection between computer and robot.
- **Weapon station sensor** - To receive information about events on the weapon station the system comes with a sensor.
- **Bluetooth connection** - If the sensor is to be connected, Bluetooth communication is required.

2.2 Software dependencies

- **Windows 7 or newer** - Software is compiled for windows.
- **Bluetooth driver** - An assigned port for sensor connection is required by the software.

3 End Effector Assembly Guide

3.1 Part Overview

This part list is used throughout this chapter, please refer back to this table when a specific part is referenced.

The parts used to assembly the End Effector:

Table 3 - Part overview

Parts:	Quantity:	Part Number:	Description
Base movable	2	1A,1B	
Base solid	1	2	
Central joints	3	3A,3B,3C	
End joints	2	4A,4B	
Fingertip	1	5	
M3 Screws	24	6	
Cabinet panel (top)	1	7	
Cabinet panel (lower)	1	8	

Side panel (long)	2	9A,9B	
Side panel (short)	2	10A,10B	
M3 Bolt	3	11	
M3 Nuts	3	12	
M3 Washer	3	13	
Power HS-1207TG Servomotors	8	14(A,B,C,D,E,F,G,H)	
M5 Bolts	12	15	
Servo axle wheels	8	16	
M3 bolt	8	17	short
Arduino cabinet	2	18	
Arduino ATmega 2560	2	19	
USB cable	1	20	
Strips	1	21	
Signal cables	1	22(A,B,C,D,E,F,G,H)	
Power cables	2	23	
M6 Thread bolt	4	24	
M6 Prevailing torque nuts	8	25	
M6 Washers	8	26	
M6 Nuts	8	27	
M3 bolt	2	28	long
M3 washer	2	29	
M4 bolt	2	30	
M4 washer	2	31	
M4 prevailing torque nuts	2	32	

3.2 Assembly instructions finger

Step 1.

Start with the solid **[table 3, part 2A]** base and **[table 3, part 3A]** the central joint, you will need a M3 bolt **[table 3, part 11]**, a M3 washer **[table 3, part 12]** and a M3 nut **[table 3, part 13]**. First assemble the bolt trough as shown in figure 1, and lock it with the washer and nut

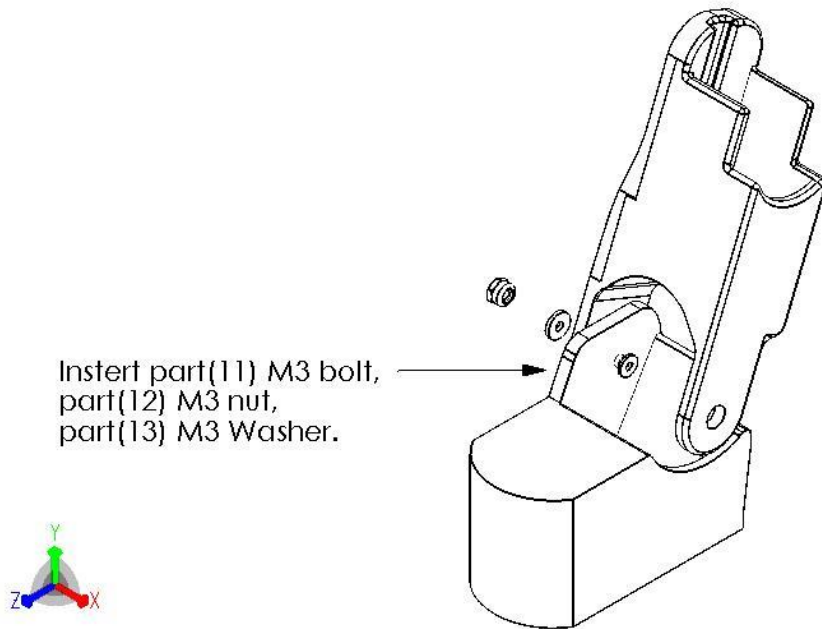


Figure 1 – Step 1 in assembly of the finger

Step 2.

Lay intermediaries ahead as shown in figure 2. This part is a little tricky, you will need to bend the section shown in figure 2 a little to the side to insert the **[table 3, part 14B]** servomotor with the servo axel pointing in the direction as shown in figure 2.



Bend carefully this section away from the servo while you insert the servo until the axel match the central joint hole.

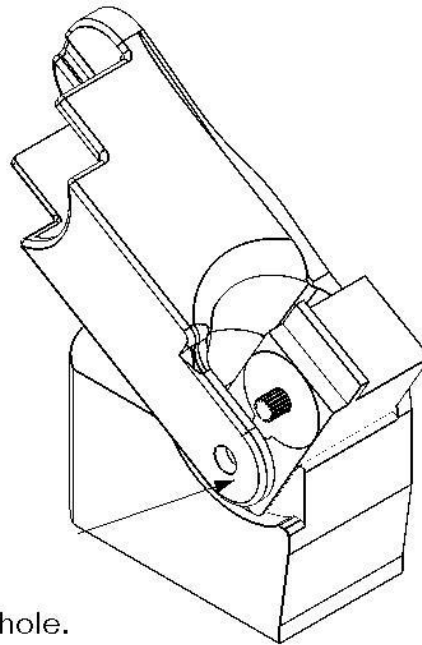


Figure 2 – Step 2

Step 3.

Tighten the servomotor with 2 5M bolts [table 3, part 15] as shown in figure 3.

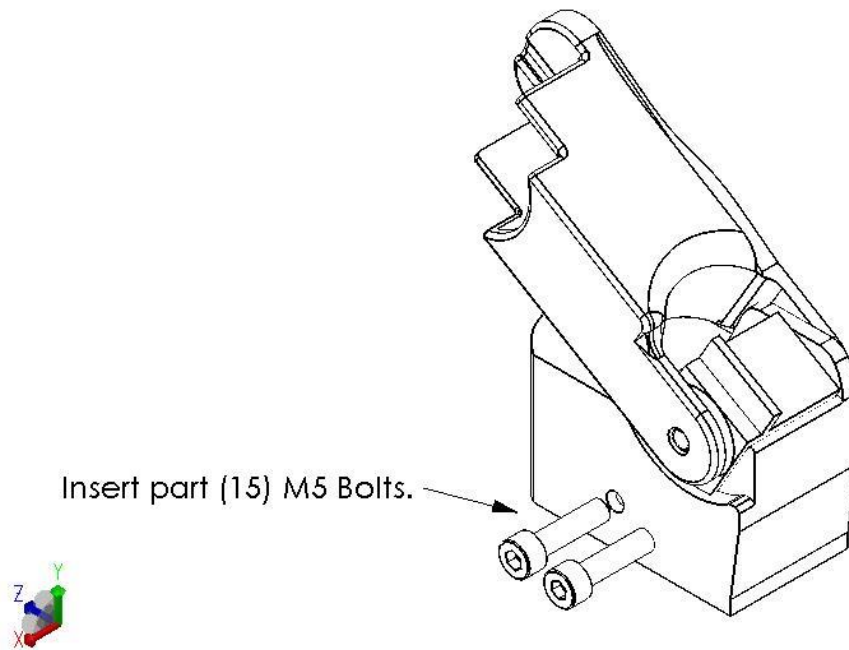


Figure 3 – Step 3

Step 4

Before the **[table 3, part 16]** servo wheel is tightened to the central joint and the servo axel, you have to put the servo in the right position and you will need to connect the Arduino to the servomotor and set it to 150 degrees before installing the servo wheel.

Install the Arduino 1.6.9 (or newer) software from <https://www.arduino.cc/en/Main/Software>

Step 5

Connect the **[table 3, part 19]** Arduino ATmega 2560 microcontroller with **[table 3, part 20]** USB cable to your computer.

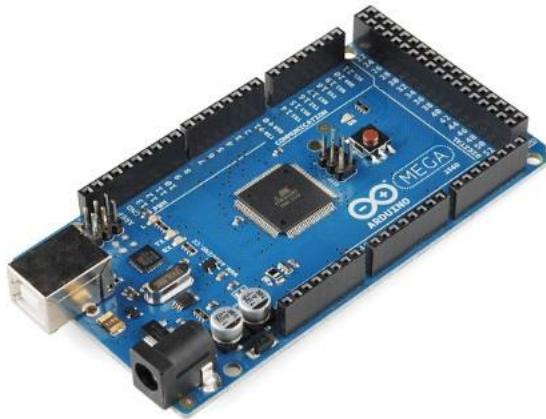


Figure 4 - Arduino ATmega



Figure 5 - USB Cable

Download the Arduino script here:

http://www.mediafire.com/download/7waavvifxwcug6k/Arduino_kode.zip and upload it to the microcontroller.



Figure 6 – Uploading the script to Arduino

Step 6.

To set the position for the servo, you will only use three of the **[table 3, part 22 A, B, C]** signal cables, as power cable, ground cable and signal cable in this step.

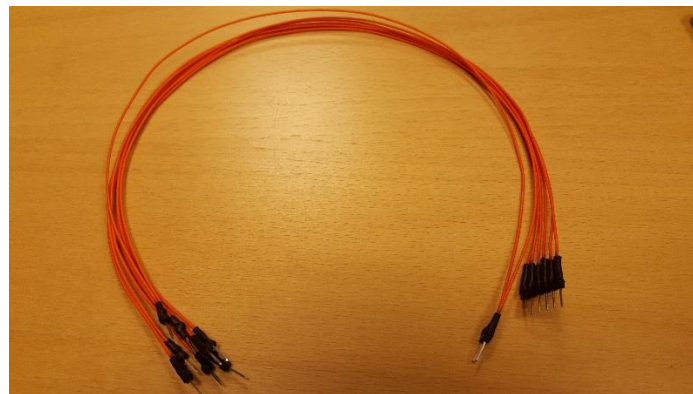


Figure 7 - Signal Cables

Connect the three signal cables to the pins (PWM 2) (GND) (5V) as shown in figure 8.

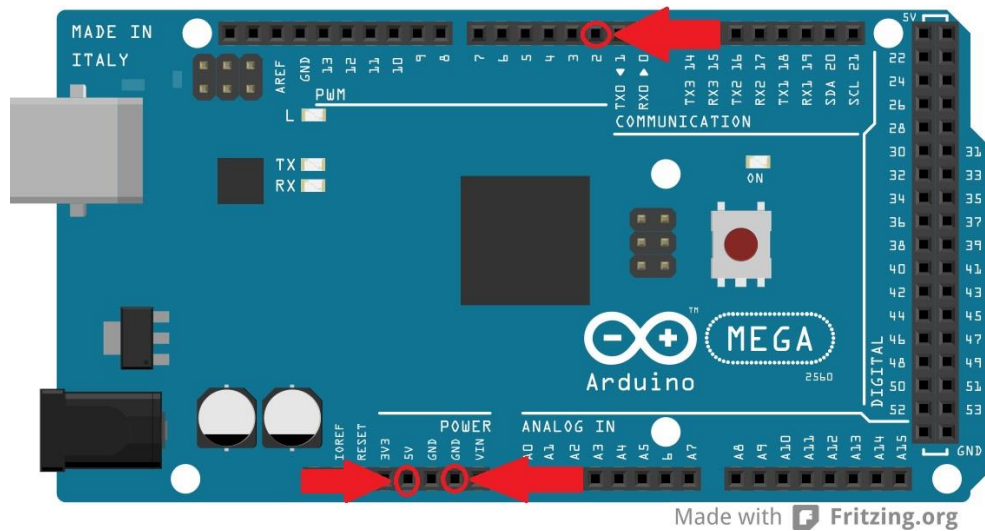


Figure 8 - Arduino ATmega

Connect the signal cable that is connected to the PWM 2 pin on the Arduino to the orange cable on the Servomotor.

Connect the signal cable connected to the GND pin on the Arduino to the brown cable on the servomotor.

Connect the signal cable connected to 5V pin from the Arduino to the red cable.



Figure 9 - Servomotor

In the Arduino software you will now open the Serial Monitor.

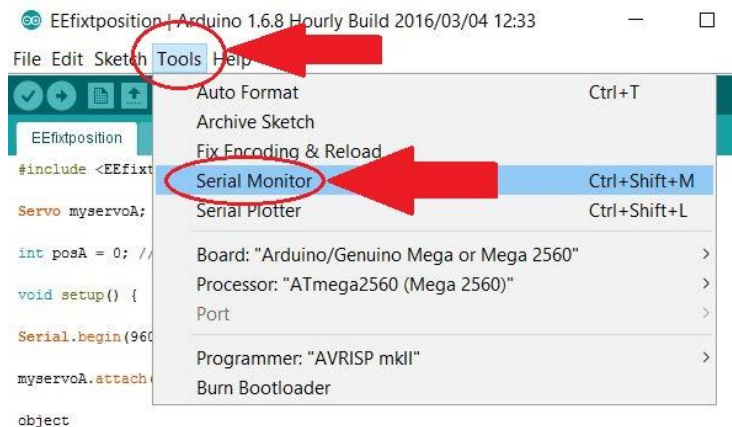


Figure 10 - Serial Monitor

When the Serial Monitor window appears, write comment "A150" in the command line and press Enter. The Servo will now move to the right position, now you can disconnect the signal cables from the servomotor before proceeding to the next assembly step.

Step 7.

Attach the servo wheel to the servo axel, making sure that the centre joint is placed in a 90 degree angle to the servomotor as shown in figure 11. Tighten the **[table 3, part 6]** M3 screws trough the servo wheel to the centre joint, and lock the servo wheel to the servo axel with a **[table 3, part 17]** M3 bolt.

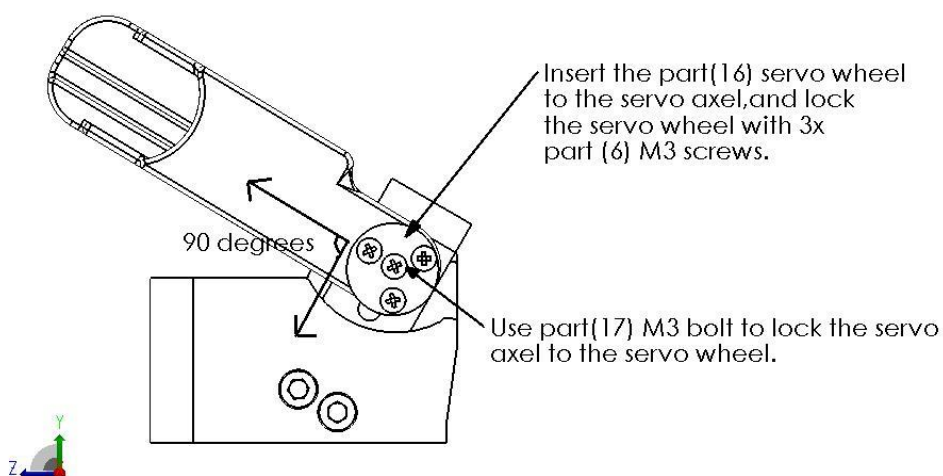


Figure 11 - Finger servo assembly

Step 8.

Insert **[table 3, part 14B]** servomotor in the central joints as shown in figure 12 with the axel pointing in the direction shown and tighten 2x M5 bolts **[table 3, part 15]**.

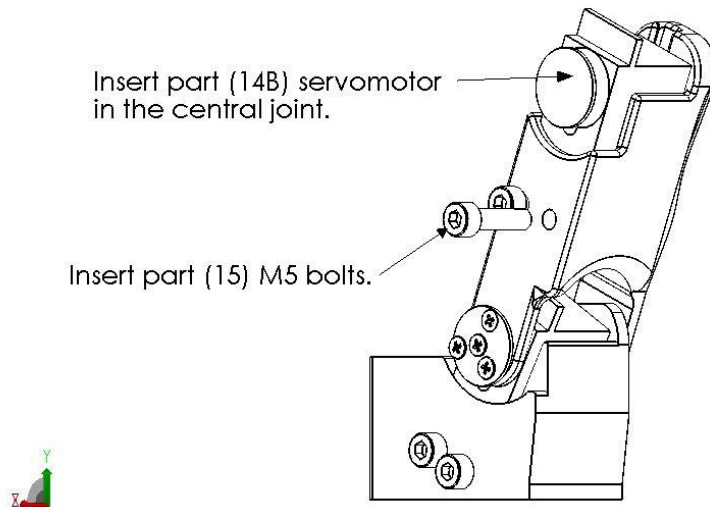


Figure 12 - Servomotor in central joint

Step 9.

You will need to repeat the Step 6 to program the position of the **[table 3, part 14B]** servomotor to be 150 degrees before starting the next step.

Step 10.

When the servomotor **[table 3, part 14B]** is set, use the fingertip **[table 3, part 5]**, and insert the servo wheel **[table 3, part 16]** then tighten the servo wheel with 3x M3 screws **[table 3, part 6]** trough the servo wheel to the fingertip.

Attach and lock the servo wheel which on the fingertip on the servo axel with M3 bolt **[table 3, part 17]** making sure the fingertip is 90 degrees of the servomotor as shown in figure 13
The “pointing finger” is now finish assembled.

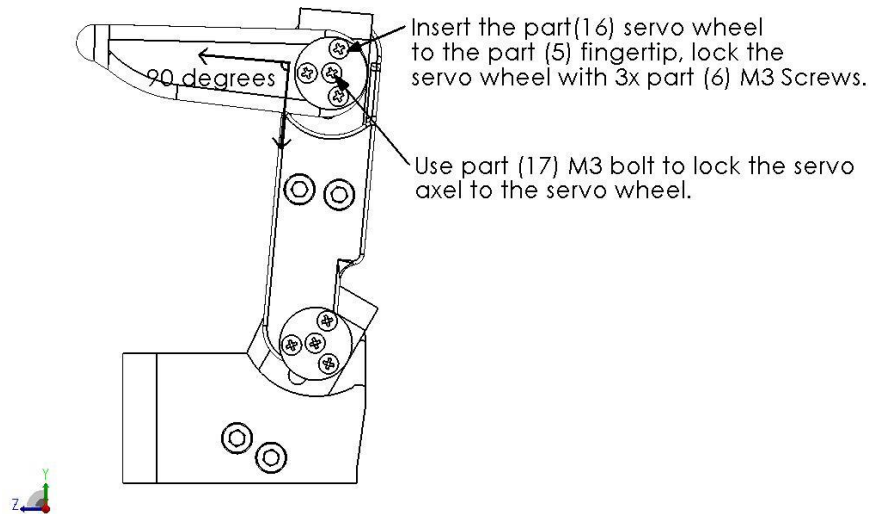


Figure 13: Step 10

Step 11.

You will now make the two other fingers, repeating steps 1 to 10 for both of them.

For the second finger replace the parts in the steps with:

- Servomotor [table 3, part 14A] with [table 3, part 14C]
- Servomotor [table 3, part 14B] with [table 3, part 14D]
- Fingertip [table 3, part 5] with end joint [table 3, part 4A]

For the third finger replace the parts in the steps with:

- Servomotor [table 3, part 14A] with [table 3, part 14E]
- Servomotor [table 3, part 14B] with [table 3, part 14F]
- Fingertip [table 3, part 5] with end joint [table 3, part 4B]

You will now have 3 separate fingers assembled.

3.3 Assembly instructions cabinet

Step 1.

Insert [table 3, part 14G] and [table 3, part 14H] servomotors into the cabinet panel [table 3, part 7].

Make sure that the servo axel is place at the centre of the cabinet panel as shown in figure 15, then tighten the servomotors with 8x [table 3, part 25] M5 bolts through the servomotors to the cabinet panel.

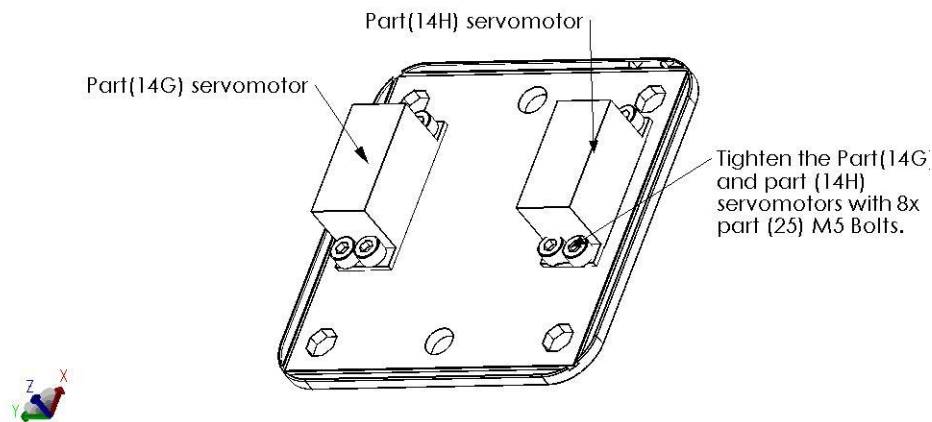


Figure 14 Cabinet panel top

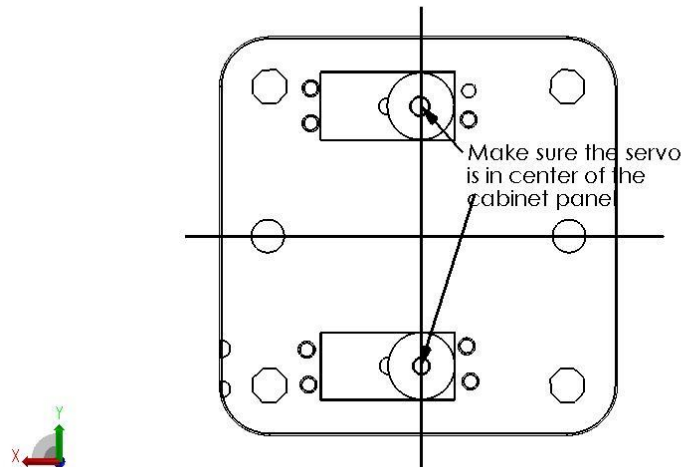


Figure 15 – Centration of the servos

Step 2.

Use 4x **[table 3, part 24]** thread bolts, 4x **[table 3, part 25]** prevailing nuts, 4x **[table 3, part 26]** washers, 4x **[table 3, part 27]** nuts, and tighten the thread bolts to the cabinet panel as shown in figure 16.

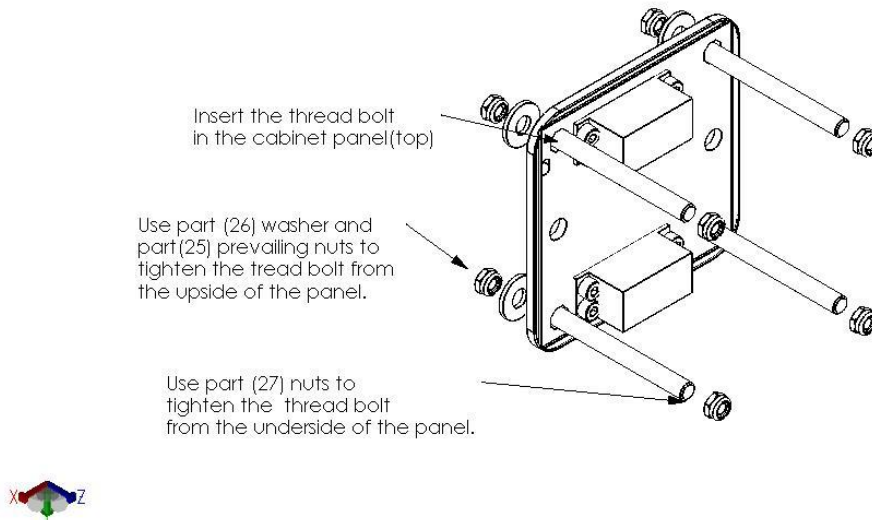


Figure 16: Tightening thread bolts

Step 3.

Insert 4x **[table 3, part 27]** nuts to the tread bolts, put on cabinet panel **[table 3, part 8]**, tighten the lower panel with 4x **[table 3, part 26]** washer, 4x **[table 3, part 25]** prevailing nuts.

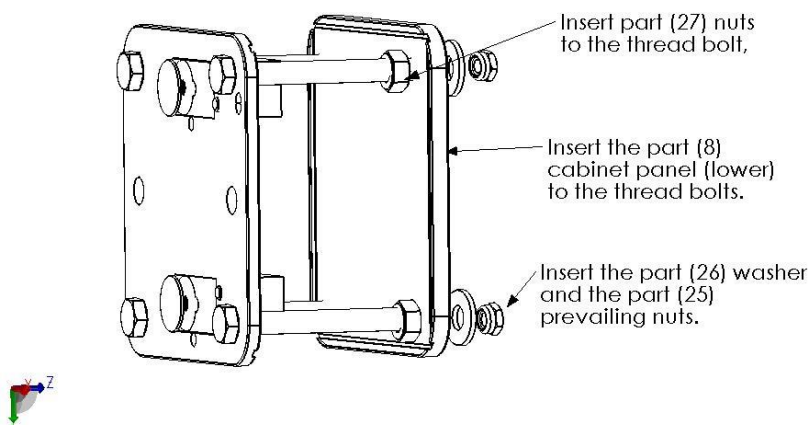


Figure 17: Attaching the lower cabinet panel

3.4 Mounting fingers to the cabinet

Step 1.

Before the attaching the fingers to the servomotors on the top panel, you have to follow the same steps for **[table 3, part 14G]** and **[table 3, part 14H]** as you did in chapter 3.2 step 5 programming the position to the servo axel.

Set **[table 3, part 14G]** to «A150» in the Arduino command serial monitor
Set **[table 3, part 14H]** to «A35» in the Arduino command.

Step 2.

Use servo wheel **[table 3, part 16]** and tighten it under the second assembled finger on the moveable base **[table 3, part 1A]** with 3x **[table 3, part 6]** M3 screws.

Repeat this step with third assembled finger movable base **[table 3, part 1B]**.

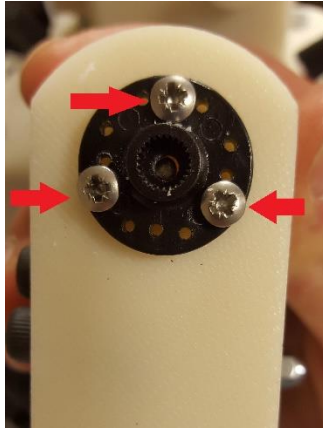


Figure 18: Attaching servo wheel

Step 3.

Attach the servo wheel which tighten to the second finger to the servo motor[**table 3, part 14G**], making sure that the base is placed in a straight line in regard to the servo motor before attaching in to the servo axel as shown in Figure 18. Tighten the finger with M3 bolt [**table 3, part 28**] and M3 washer [**table 3, part 29**] to the servo axel.

Repeat this step for the third finger on servo motor [**table 3, part 14H**].



Figure 19: Attaching finger to the servo on top cabinet panel

Step 4.

Attach the first finger to the top panel as shown in figure 19 through the solid base with 2x M4 bolts [table 3, part 30] and 2x M4 washer [table 3, part 31], tighten the M4 bolts from the inside of the cabinet with 2x prevailing nuts [table 3, part 32] as shown in figure 20.

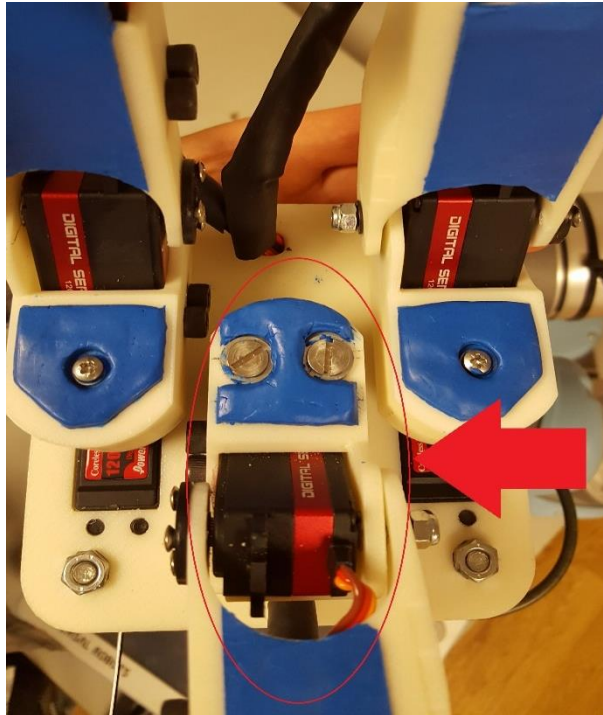


Figure 20: Attaching middle finger

3.5 Mounting End Effector to the UR5 Robot arm

Step 1.

Take the UR5 interface [table 3, part 33] and tighten it to the UR5 with 4x M6 bolts [table 3, part 34].

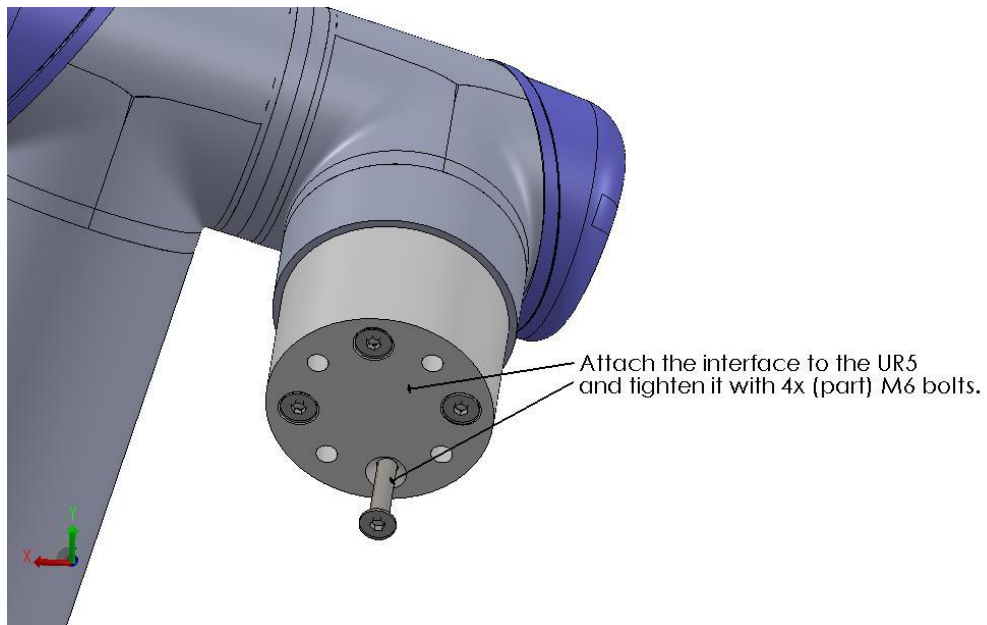


Figure 21: Attaching interface to UR 5

Step 2.

Tighten 4x thread bolts [table 3, part 35] to the interface.

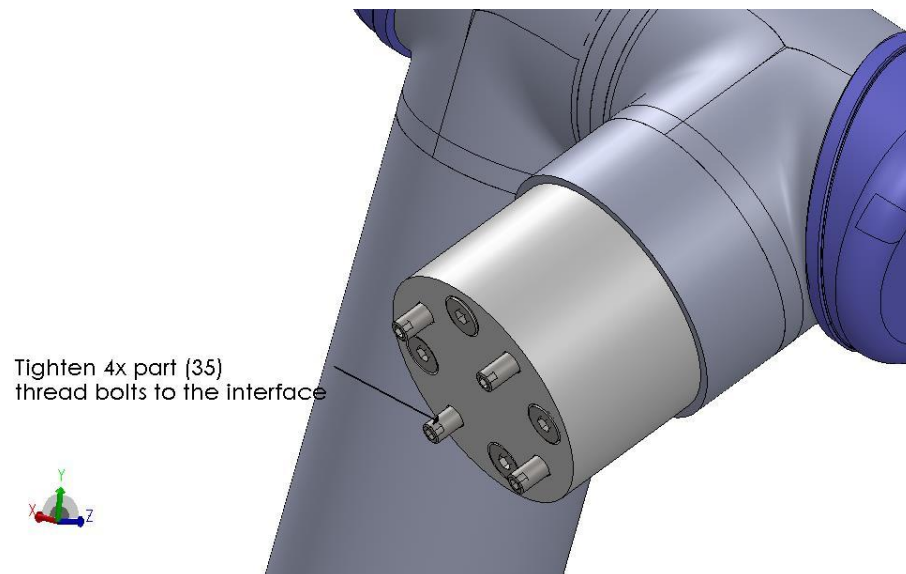


Figure 22: Attaching thread bolts to interface

Step 3.

Attach the End effector assembly with the lower end of the cabinet panel, to the thread bolts on the interface of the UR5 through the 4 holes on the lower panel, and tighten it with 4x wing nuts [table 3, part 36].

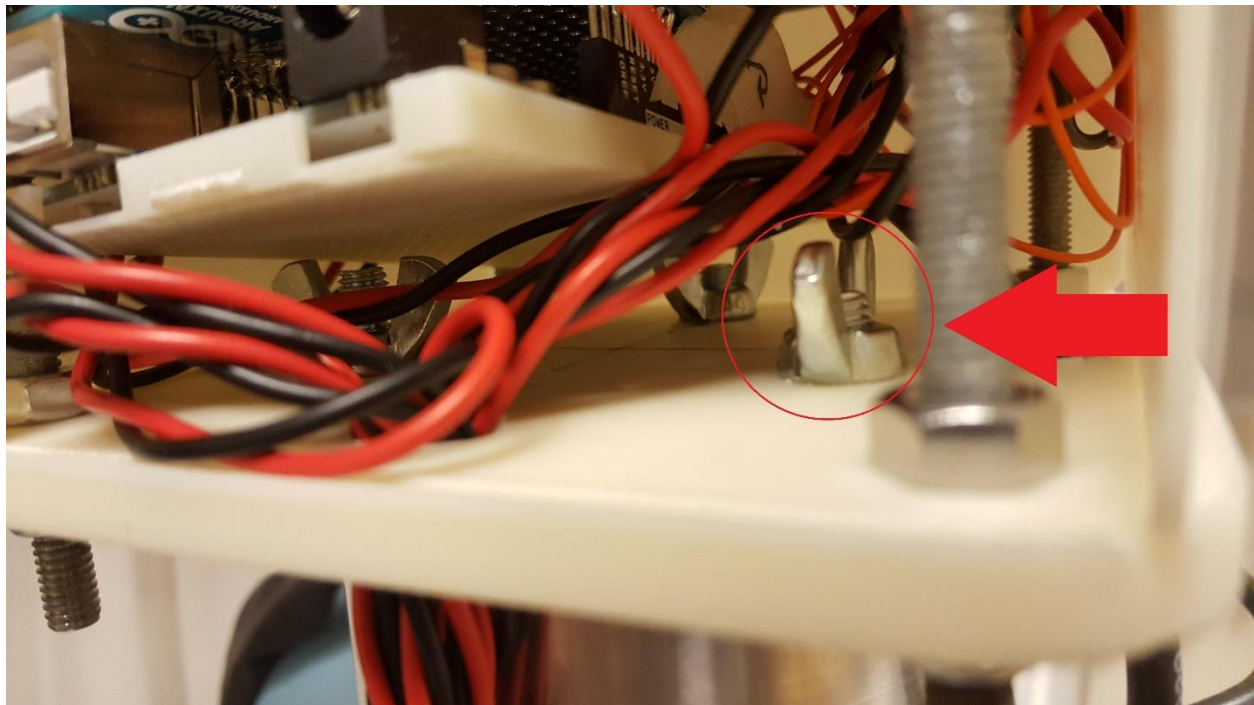


Figure 23: Tightening EE to interface

3.6 Attaching cables

For this step you will need to do some cable management, and follow the circuit diagram shown in figure 24.

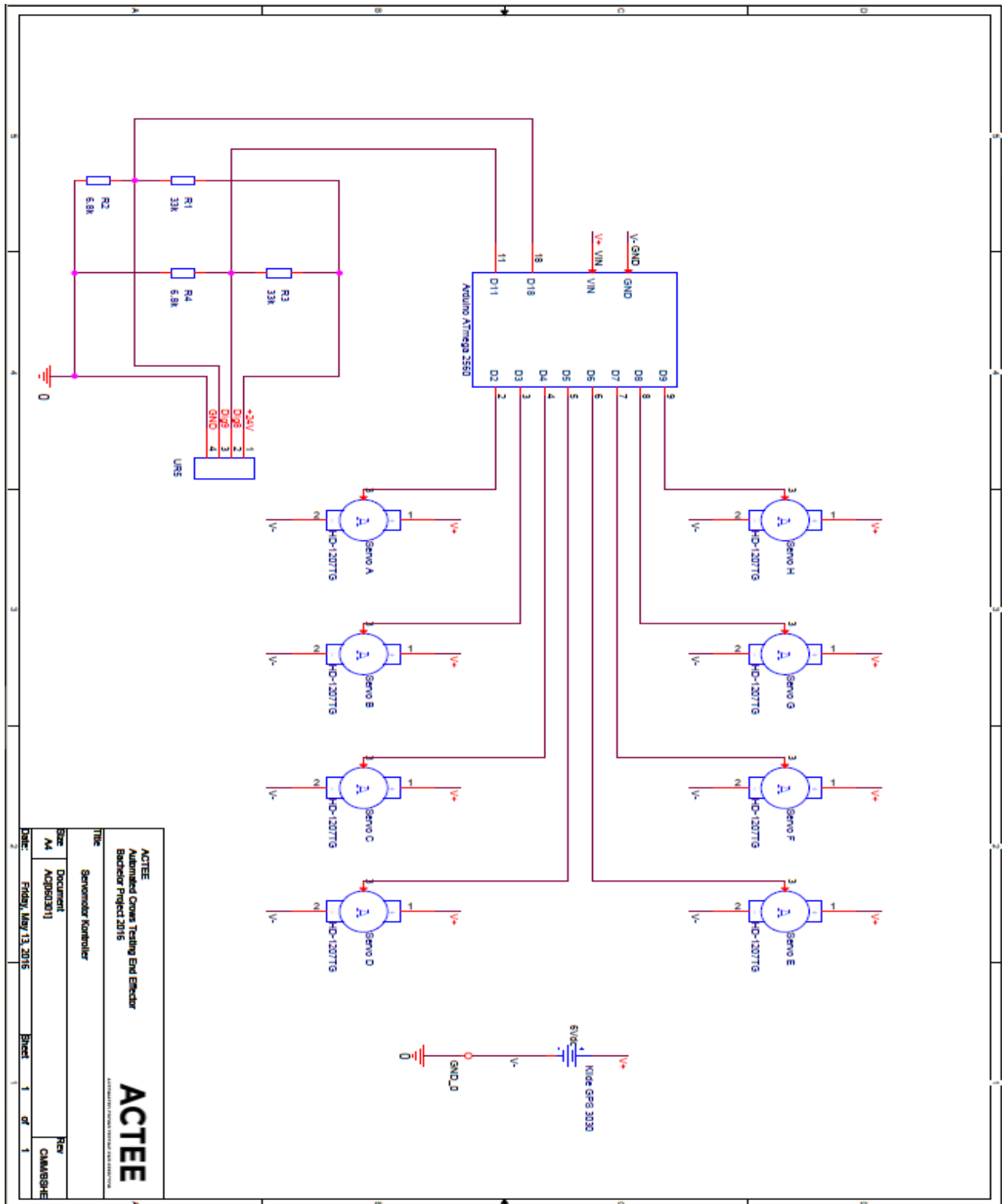


Figure 24: Circuit diagram

You will need a power supply that can deliver enough power to the servomotors. Power supply used for developing the product was a CPX200 DUAL 35V 10A power supply. The End Effector system requires 6V, 9A supply.

Power cables **[table 3, part 23]** are shown in figure 25.



Figure 25: Power cables

Signal cables **[table 3, part 22 A, B, C, D, E, F, G, and H]** are shown in figure 26.

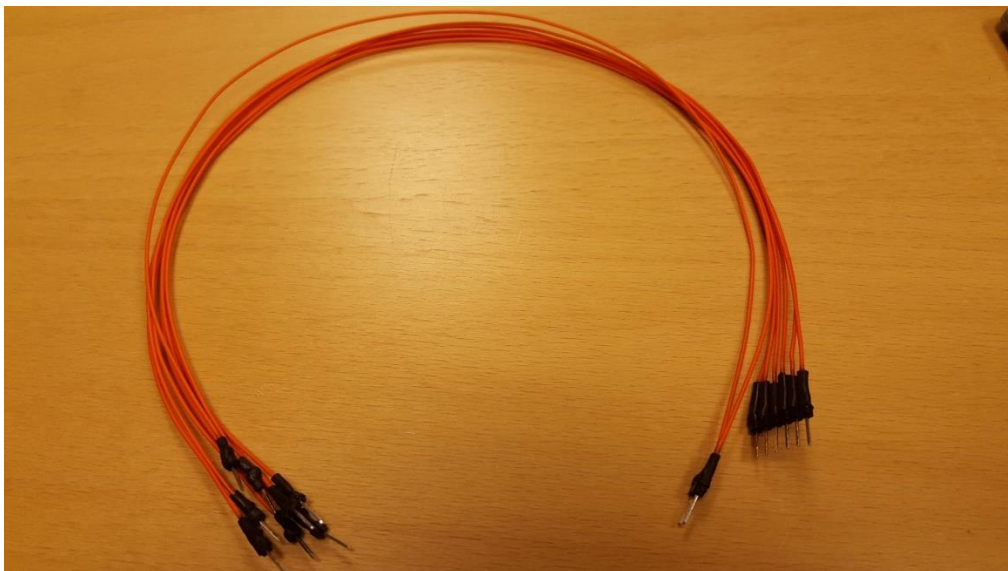


Figure 26: Signal Cables

There is a cable hole on the lower cabinet panel where the power cable can be inserted.

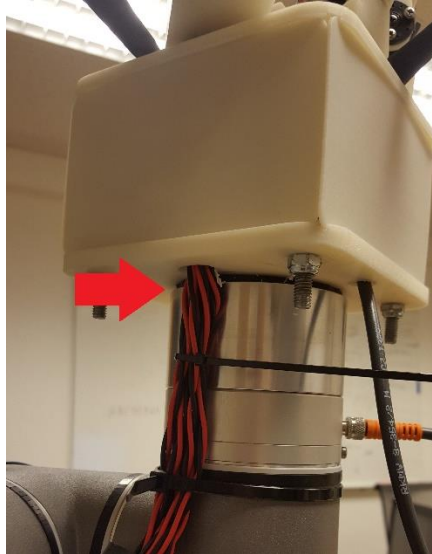


Figure 27: Power cables

On the top cabinet panel there are holes for the cables to go to the different fingers, be aware to divide the cables so that there is enough ground and power cables for each of the servo motors. Follow the circuit diagram to attach right cable to right place.

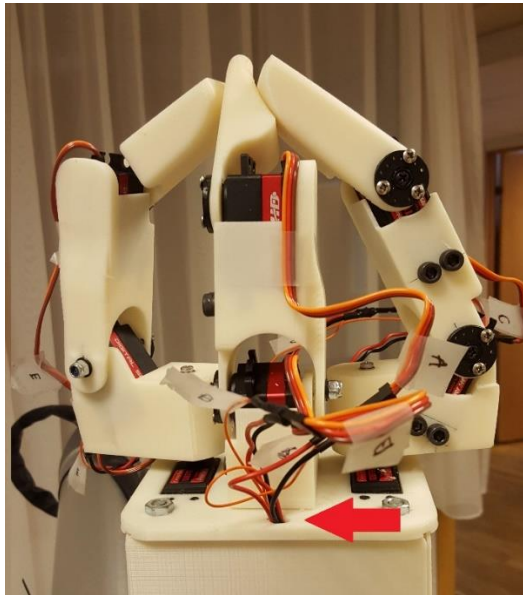


Figure 28: Dividing cables back

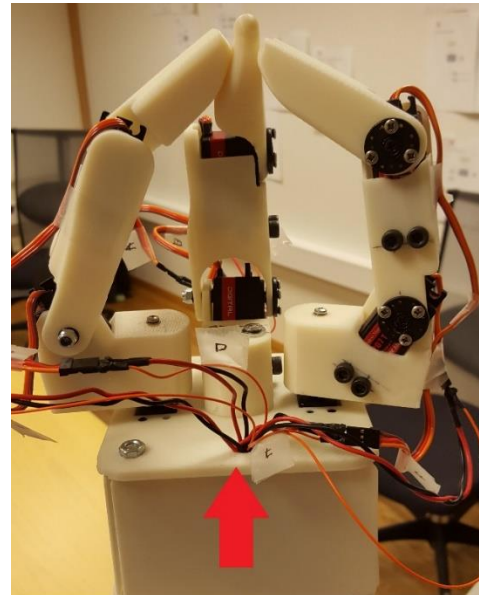


Figure 29: Dividing cables front

4 Software Tutorial

4.1 Software installation

The software is compiled for windows 7 and newer and can be run directly, however the installation file is an executable archive containing a temporary log file that will be stored in the same folder as the executable. First locate and run the installation file.

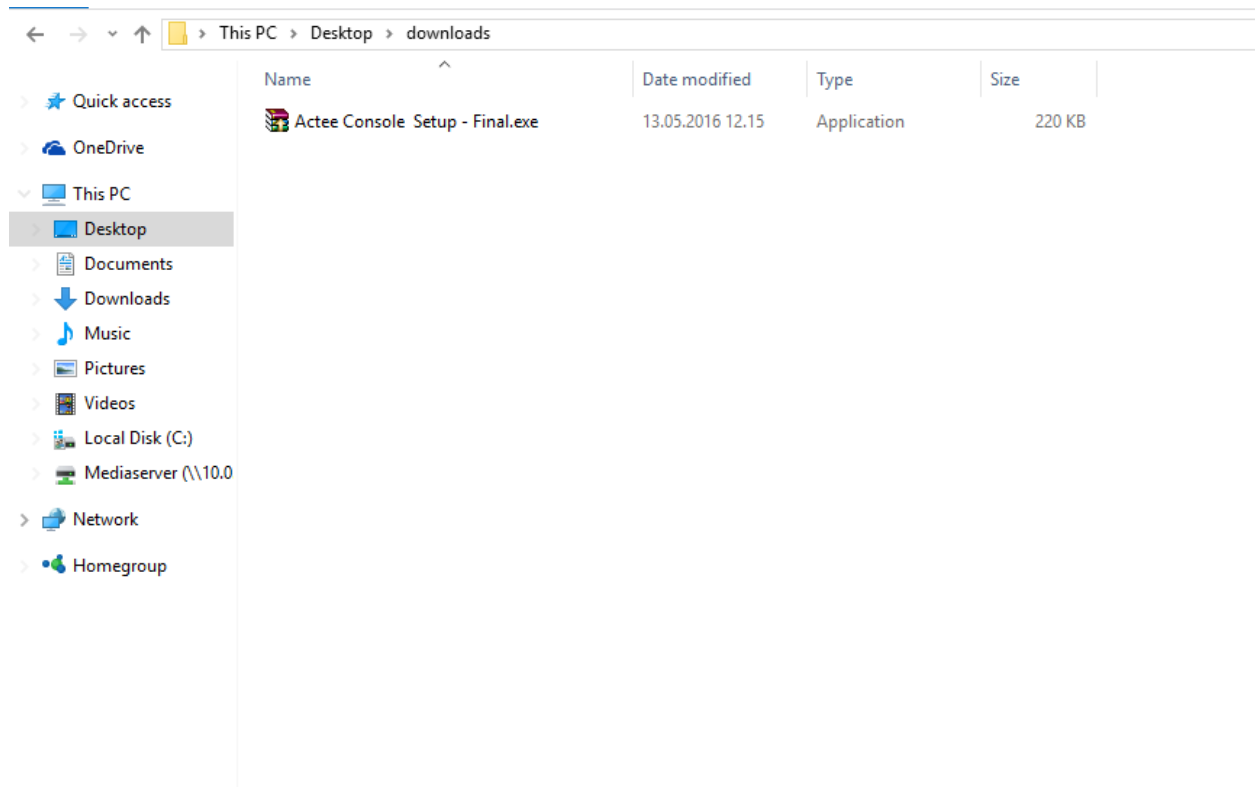


Figure 30 - Locate the installation file

The installation file is set to install the software to c:\ACTEE\ if you want to change that you must edit the extract destination.

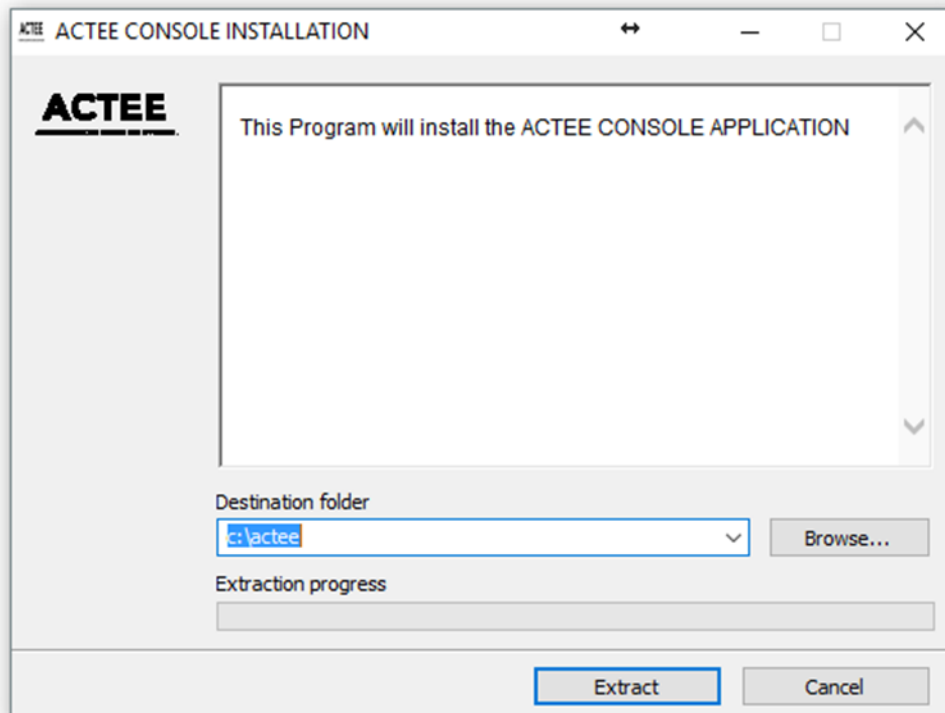


Figure 31 - Software Installation

After the installation go to the destination folder to run the application.

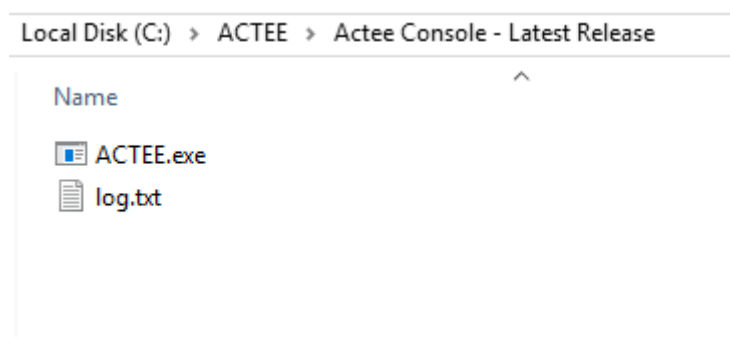


Figure 32 - Destination directory

You are now ready to run the application

4.2 Software removal

The software can be removed simply by deleting the program folder. No further uninstallation needed.

4.3 Client overview

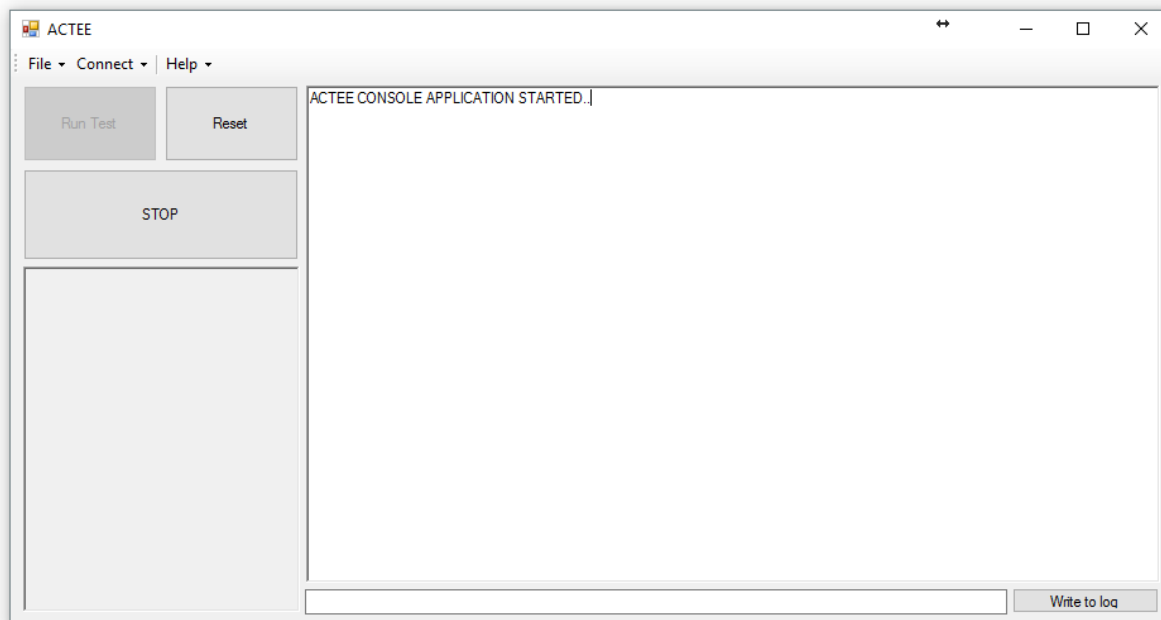


Figure 33 - Client overview

The client's toolbar contains File, Connect and Help.

File contains **Clear log**, if you want to reset the log, without resetting the entire client application. **Save log**, to save the content of the big white textbox to a txt file. **Edit test**, allows the user to define their own test sequences. **Load test**, to load an earlier saved test for a XML file. And **Exit**, to exit the application.

Connect contains **Connect to UR5** see 4.4 Connecting to the UR5, and **Connect to WSS**, see 4.5 Connecting to the WSS

Help contains **Quick Help**, a short instruction. **Online Manual**, A link to an online version of this document. And **About**.

Just below the toolbar menu bar the buttons Run Test, Reset and Stop are located.

For **Run Test** to become enabled a test must be loaded and the system must be connected to the UR5. When these conditions are met, Run Test starts the test sequence.

Reset resets the entire application, including the log file.

Stop stops the test. This allows the user to stop the test and, still be able to save the log. If the user wants to continue running tests he or she has to reset the application by pressing reset and reconnect to the ur5. (4.4)

The Write to log button on the bottom right. Allows the user to make comments in the log. It simply adds the content of the textbox to the right into the log window.

4.4 Connecting to the UR5

To connect to the UR5 robot a local area network must be in place. Both UR5 robot and the system running the test software has to be connected, and its advised to have both units receive their IP-address from the switch, or router. The TP connection on the UR 5 robot is beneath the control box.

Start the UR 5 by pressing the power button on the connected tablet.

After initializing the robot choose “load a program”, find the “ACTEE.urp” file and select open.

Check the IP of the test machine, and write the address in the “Socket_open” line in the code on the UR 5.

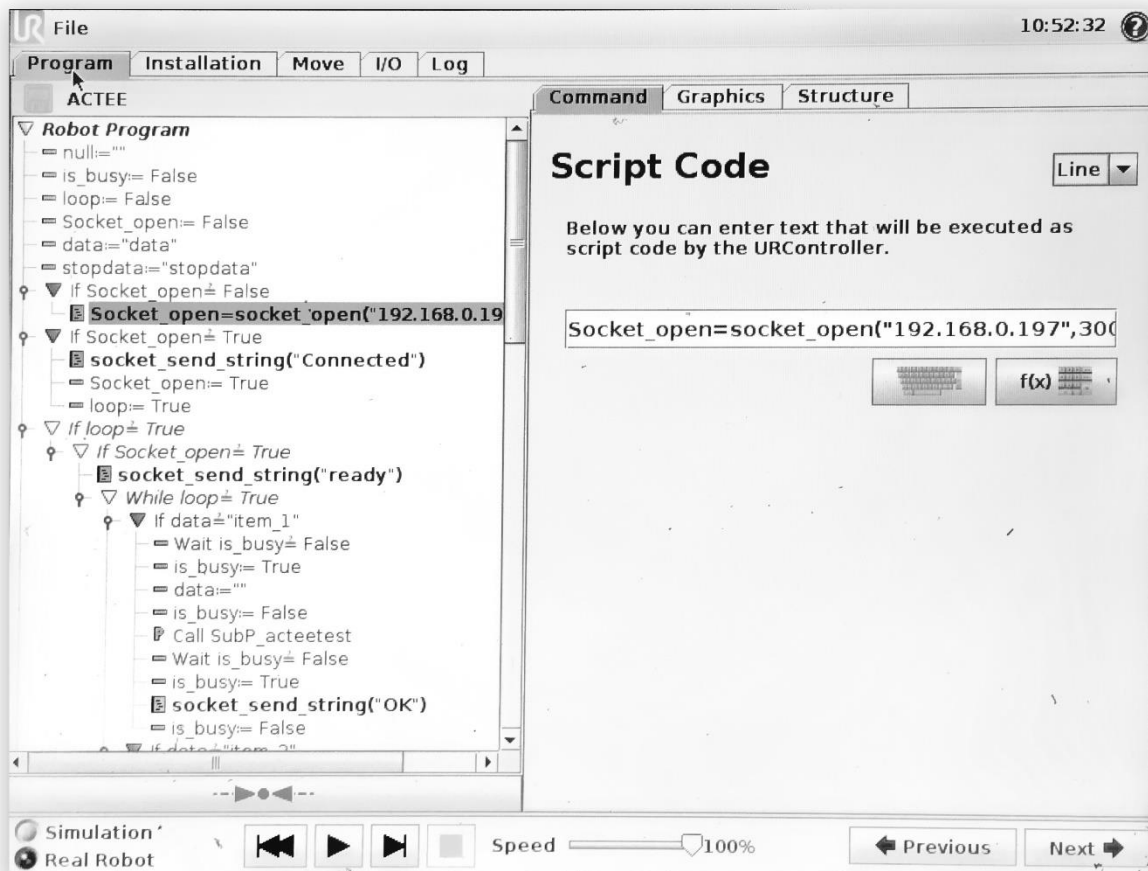


Figure 34: Socket open UR 5 program

Now press the play button to start the program on the UR 5, it will start a loop that tries to connect to the given IP address.

In the test software click the Connect tab, and press Connect to UR 5.

Wait for the "The robot is connected" message to show up in the log, if it doesn't. See troubleshooting.

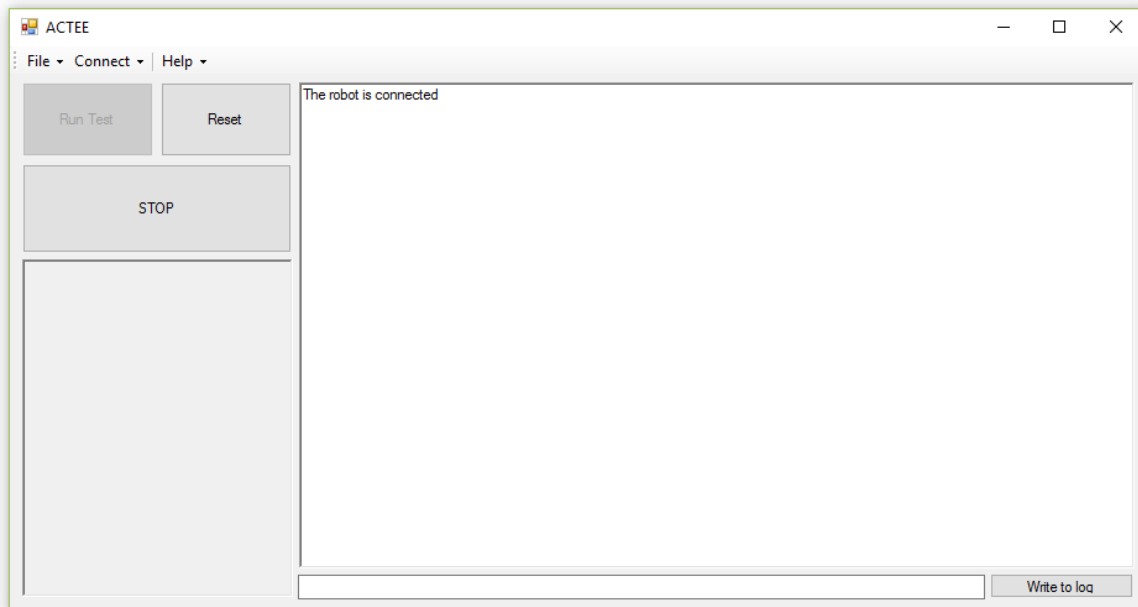


Figure 35: Robot connection status

Now the connection is established and we can load and run a test sequence.

4.5 Connecting to the Weapon Station Sensor

To connect to the weapon station sensor you have to connect to the Bluetooth unit in the Bluetooth settings on the computer.

It's normally found in Control panel\Devices\Bluetooth

Make sure the sensor is on and search for new devices. When BTCOM appears choose connect and enter 1234 when prompted for a key.

Scroll to the bottom and select more Bluetooth options to receive a list of connected com ports, make note of the outgoing port number for BTCOM. You need this in the next step.

Go back to the Actee console application, select connect form the menu bar and Connect to WSS select the port which the device is connected and press connect. When the sensor connection is established, the user is informed trough the log output.

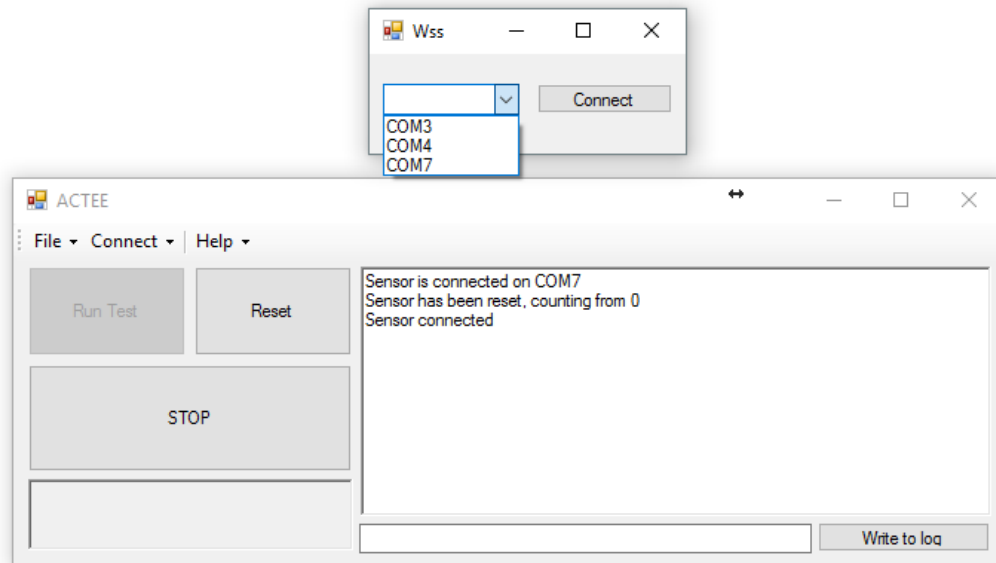


Figure 36: Wss connection

4.6 Running a test

The software needs to be connected to the robot to run a sequence, see chapter 4.4 for help.

Click file; load test file. A dialog box will open, navigate to the wanted test file and press open.

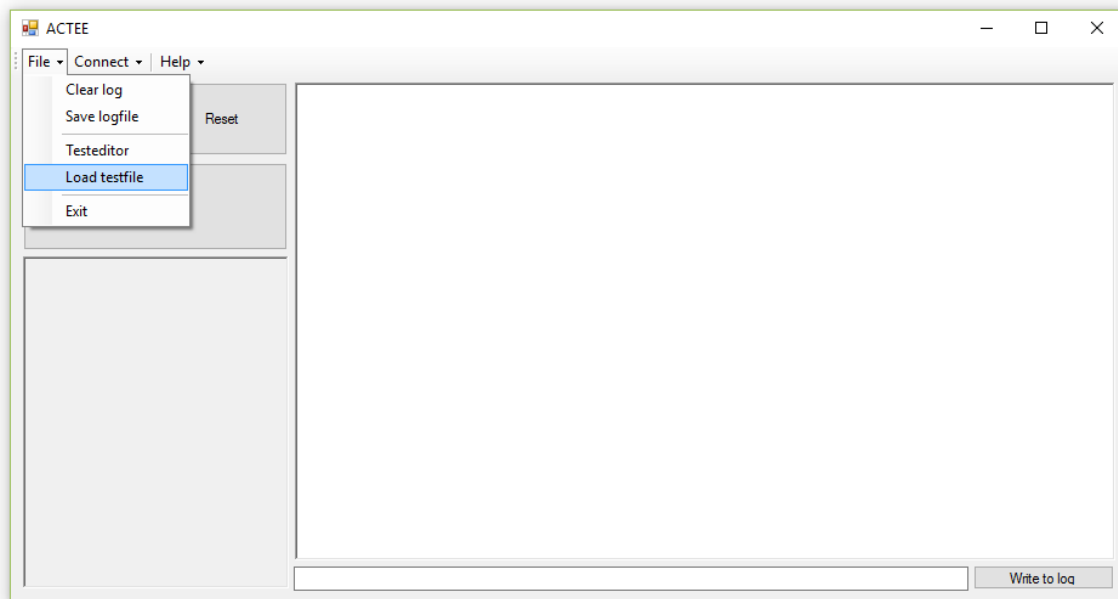


Figure 37: Load test file

After clicking on open, the content of the chosen file will be written to the box on the left as shown in figure 6.



Figure 38: Test information

You will need to verify that this is the sequence you want to run. If you press the Run Test button now, the test will start.

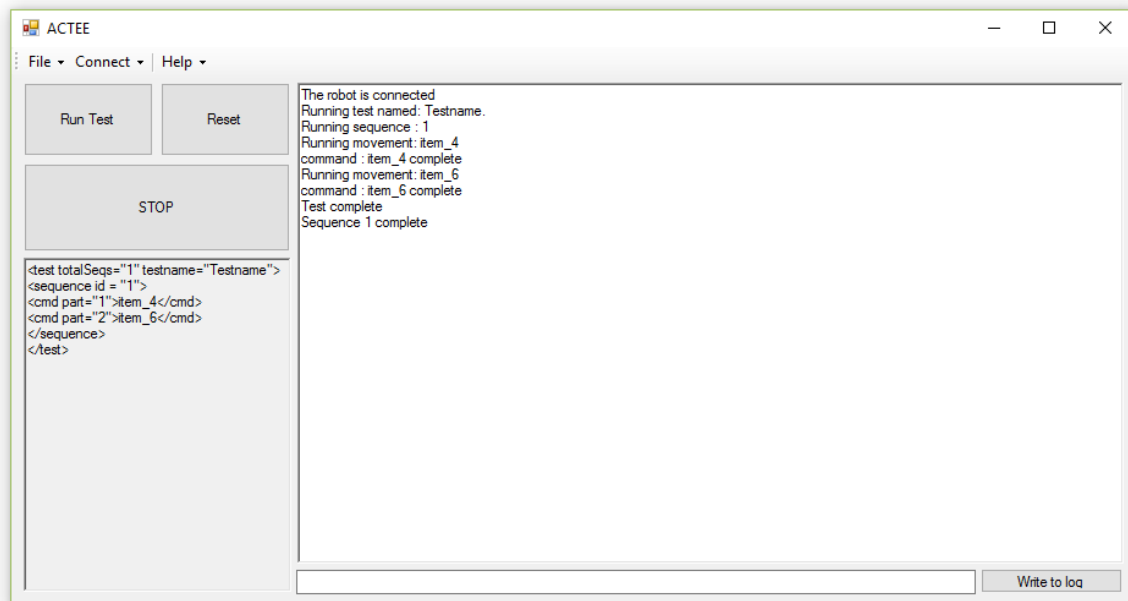


Figure 39: Test output

When running the test, information about the test running will be written to the log for each movement, and you will get information when the current sequence and the current test is completed.

4.7 Creating a test using the Test editor

In this section we will show how the user can build their own test sequences from a selection of the possible movements.

After launching the program:

1. Click on File.
2. Select Test editor.

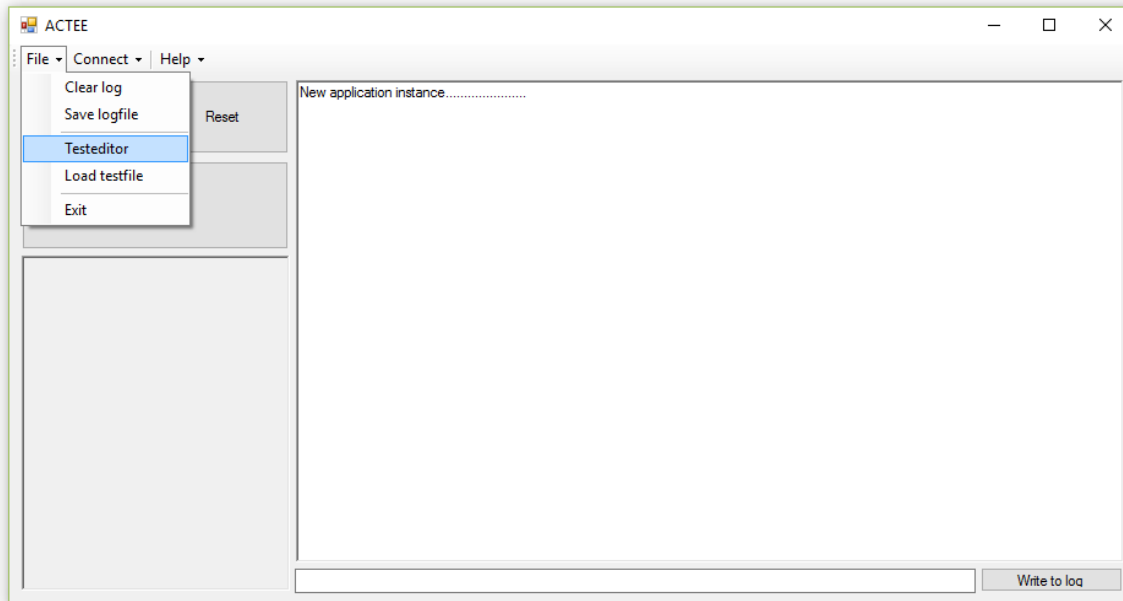


Figure 40: Launching the test editor

This will open the test editor where it's possible to create a test sequence and save it using the XML file format.

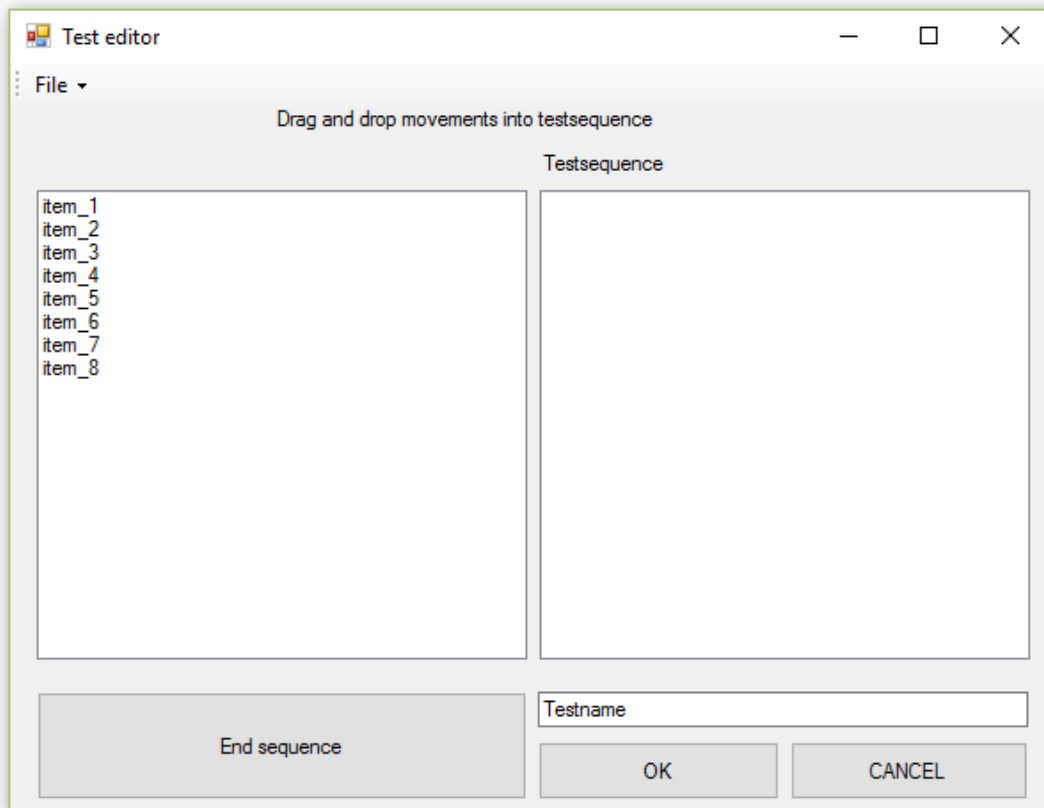


Figure 41: Test editor GUI

Here the user can drag and drop the wanted movements from the left side to the right. After each sequence is completed the user has to click on the «End sequence» button to mark that the current sequence is completed.

The “End sequence” button is used to mark the end of a sequence, this button needs to be pressed after each part of the test and at the end of the test sequence.

An example test sequence would look something like this.

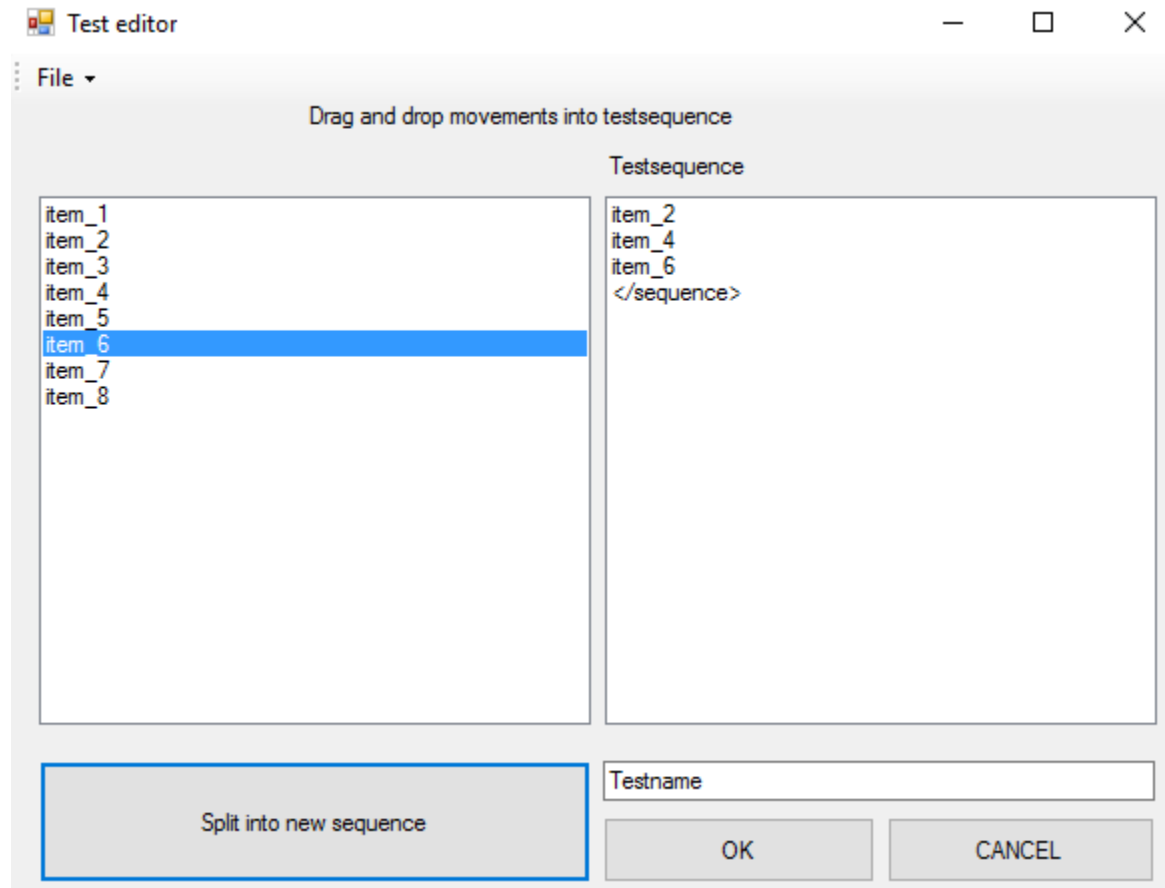


Figure 42: Building a sequence

If there were placed an item in the test sequence that is not supposed to be there, it can be removed by selecting it and pressing the delete button.

By inputting the test name in the "Test name" field, it will be written to the XML and set as the filename when opening the save xml dialog.

To save the sequence to disk as an XML file press file, then save test. You will get a save file dialog box, choose the location on disk where you want to store the file and press save.

You can now go back to the main program by pressing the OK button, and loading the test sequence you created by pressing file, load test file and choosing the location on disk.

4.8 Saving the log file

To save the log file after a test, go to file press export log file.

Sources

[1] A.K. Svendsen; E.L. Roa; H.B. Sørsum; S. Rudin. "Automated CROWS Testing" Bachelor Thesis, Fakultet for teknologi og maritime fag, Høgskolen i Buskerud og Vestfold, Kongsberg, 2014.

Available from:

<http://hdl.handle.net/11250/216910>

Last visited: 16.03.2016

[2] Universal Robots AS. (2015).

UR5 - robot.

Available from:

<http://www.universal-robots.com/no/produkter/ur5-robot/>

Last visited: 16.03.2016

[3] Universal Robots support

Ethernet socket communication via URScript

[Webpage]

Available from:

<http://www.universal-robots.com/how-tos-and-faqs/how-to/ur-how-tos/ethernet-socket-communication-via-urscript-15678/>

Last visited: 16.03.2016

[4] Universal Robots

«The URScript programming language»

Available from:

http://www.wmv-robotics.de/home_htm_files/scriptmanual_en_1.5.pdf

Last visited 02.05.2016

[5] Universal Robots

“Polyscope Manual”

Version 1.7, 2012, side 5.

Available from:

http://support.universal-robots.com/foswiki/pub/Downloads/SoftwareUpdates/software_manual_en_Global_1.7beta.pdf

Last visited 03.05.2016

[6] Microsoft Developer Network

“NetworkStream.Write Method”

[Webpage]

[https://msdn.microsoft.com/en-](https://msdn.microsoft.com/en-us/library/system.net.sockets.networkstream.write%28v=vs.110%29.aspx?f=255&MSPPError=-2147217396)

[us/library/system.net.sockets.networkstream.write%28v=vs.110%29.aspx?f=255&MSPPError=-2147217396](https://msdn.microsoft.com/en-us/library/system.net.sockets.networkstream.write%28v=vs.110%29.aspx?f=255&MSPPError=-2147217396)

Last visited:

05.05.2016

ØKONOMI

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAKSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Økonomi
VERSJON / DATO	1.0 / 19. Mai 2016
FORFATTER	Espen Svenne Kraglund
SIDEANTALL	6



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAURSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



i. Abstrakt

Dette dokumentet gir en kort oversikt over økonomien for ACTEE prosjektet. Det vises til budsjett utarbeidet underveis og til en oversikt over hvor mye midler som ble brukt.

ii. Innholdsfortegnelse

i. Abstrakt.....	1
ii. Innholdsfortegnelse	2
iii. Tabell over tabeller	2
iv. Dokumenthistorie.....	2
1 Innledning	3
2 Budsjett.....	4
3 Regnskap.....	5
4 Konklusjon	5

iii. Tabell over tabeller

Tabell 1: Dokumenthistorie.....	2
Tabell 2: Budsjett	4
Tabell 3: Regnskap	5

iv. Dokumenthistorie

Tabell 1: Dokumenthistorie

v 1.0 19.05.16	Espen: Opprettet dokumentet og skrevet dokumentet.
----------------	--

1 Innledning

Dette er en økonomisk oversikt for bachelor gruppen ACTEE. I dette dokumentet er det et budsjett som utarbeidet under planleggingsfasen av prosjektet og en oversikt over hvor mye penger vi endte opp med å bruke.

For å utarbeide budsjettet ble hele gruppen spurt om forventet forbruk og det ble justert underveis i prosjektet ettersom gruppen fikk større innsikt i hva som måtte til for å oppfylle oppdragsgivers krav, og hvilke kostander de enkelte delene kom til å ha.

Ved prosjektets start ble det gitt en absolutt ramme fra oppdragsgiver på 20000,- kr. Dette ble satt som øverste grense for budsjettet og ble tatt til etterretning når vi vurderte mulige løsninger på oppgaven.

2 Budsjett

Tabell 2: Budsjett

Utgift	Beskrivelse	Budsjettet
Kontorutgifter	Penner, Papir og CD/DVD for Dokumentasjon, Poster	kr 1000
Mek. utvikling	Komponenter, materialer til endelig prototype	kr 3000
	3D print av prototype	kr 1200
Elektro utvikling	Mikrokontroller, komponenter	kr 2000
	Servoer	kr 4000
Webside	Domene	kr 275
Presentasjonsutgifter	Presentasjon 1	kr 200
	Presentasjon 2	kr 200
	Presentasjon 3	kr 400
Dokumentutgifter	Plakat, printing av dokumentasjon.	kr 500
Totale utgifter		kr 12775

3 Regnskap

Tabell 3: Regnskap

Utgift	Beskrivelse	Kostnad
Kontorutgifter	Skilleark	kr 24
Mek. utvikling	Fire 3D printer av prototype	kr 1595
Elektro utvikling	Mikrokontroller, komponenter	kr 1358,21
	Servoer	kr 4515
Presentasjonsutgifter	Presentasjon 1	kr 195
	Presentasjon 2	kr 95
	Presentasjon 3, stipulert	kr 400
Totale utgifter		kr 8185,21

4 Konklusjon

Per 20.05.2016 har vi brukt 4589,76,- kr. Mindre enn vi hadde budsjettert med og havner godt innenfor budsjettgrensen gitt fra oppdragsgiver på 20000,- kr. Fra tabell 2 kan vi se at det er flere av postene som faller helt bort, der har vi enten gått for en annen løsning eller så har postene blitt så mye lavere enn budsjettert at det ikke har blitt søkt om refusjon.

Vi er fornøyd med å ha funnet en løsning på oppgaven som gjør at vi ikke bruker unødvendig mye penger, og ser verdien av og budsjettere under ett prosjekt. Utarbeidelsen av budsjettet førte til at vi måtte ta kostnadene med i overveielsene vi gjorde noe som hjalp oss til å gjøre bestemmelser.

Timelister

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAGSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Timelister
VERSJON / DATO	1.0 / 19. Mai 2016
FORFATTER	Thomas Wegener
SIDEANTALL	37



KONGSBERG

**AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER**



i. Abstrakt

Dette dokumentet inneholder en oversikt over timer brukt for hvert enkelt medlem av bachelorgruppen. Listene referer til «User Storys» i «Project Backlog». For en beskrivelse av disse arbeidsoppgavene se vedlegget Project Backlog under Prosjektplanen.

ii. Innholds oversikt

i. Abstrakt.....	2
ii. Innholds oversikt	2
iii. Dokumenthistorie.....	3
1. Innledning	4
2. Timeliste: Thomas Wegener	4
3. Timeliste: Aina Engen Nilsen	9
4. Timeliste: Bård Simen Hamborg Enget	17
5. Timeliste Carl Martin Mausest.....	23
6. Timeliste Monica Nadia Haugen.....	27
7. Timeliste Espen Svenne Kraglund	32
8. Konklusjon	37

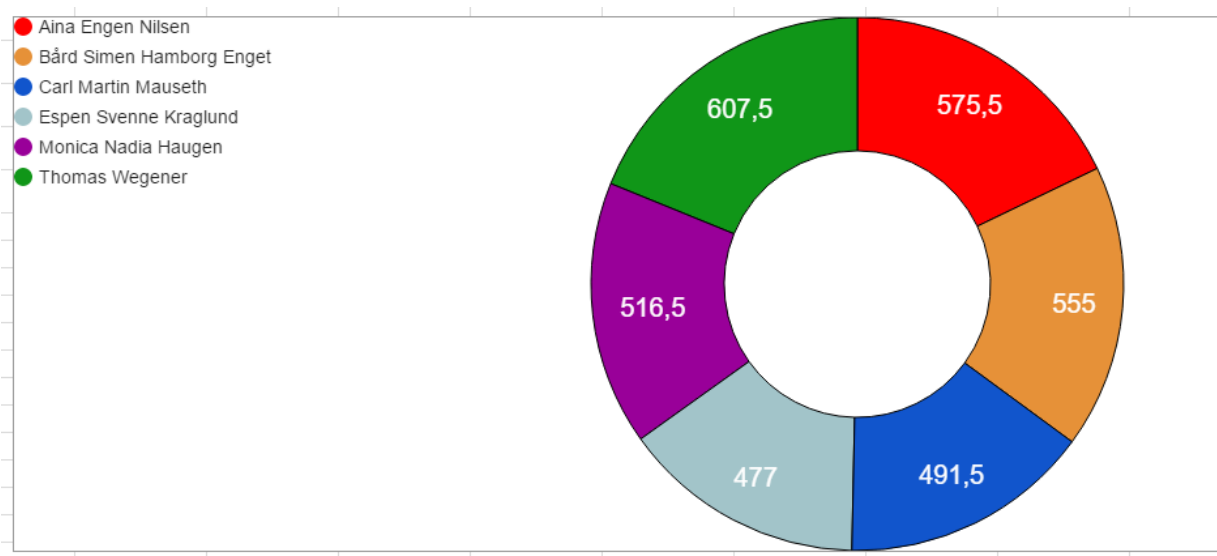
iii. Dokumenthistorie

Tabell 1 Dokumenthistorie

Versjon 0.1	20.02.16	Mal av Thomas.
Versjon 1.0	19.05.16	Thomas: Kopiert inn timelister og skrevet tekster til listene.

1. Innledning

Her er en oversikt over arbeidsfordelingen internt i gruppen.



Det har blitt lagt ned en stor innsats med oppgaven, det totale timeantallet er blitt 3223, i tillegg kommer arbeidet med den siste presentasjonen og demonstrasjon av systemet.

2. Timeliste: Thomas Wegener

Totalt timeantall: 607,5

DATO	ARBEIDSBESKRIVELSE	ARBEIDSOPPGAVE (Fra tids budsjettet)	ARBEIDSSTED	TIMER BRUKT
06.10.2015	Første møte		HBV	2
13.10.2015	Møte internt, hvilken oppgave skal vi søke på.		HBV	3
14.10.2015	Skrevet søknad og cv	Dokumentasjon / Søknad / ADM	HJEMME	8
20.10.2015	Møte internt	SØKNAD/ADM	HANGOUTS	3
03.11.2015	MØTE INTERNT	SØKNAD /ADM	HBV	3
09.11.2015	MØTE INTERNT	OPPGAVE/ADM	HBV	2
23.11.2015	Møte nr 1 hos KPS informasjon om oppgaven	MØTE/ADM	KPS	3
01.12.2015	Forberedende møte Internt	MØTE/ADM	HBV	3

02.12.2015	Møte nr 2 hos KPS intensjon	MØTE/ADM	KPS	4
23.12.2015	Planlegging og struktur		HJEMME	7
27.12.2015	Planlegging og research		HJEMME	7
02.01.2016	Planlegging og research		HJEMME	4
03.01.2016	Planlegging og research		HJEMME	2
04.01.2016	Test hos kps	RESEARCH	KPS	5
05.01.2016	Møte	DOKUMENTASJON	HSN	6
05.01.2016	Timeregistreing	DOKUMENTASJON	HJEMME	3
07.01.2016	Work Breakdown		HSN	2
08.01.2016	Logo		HSN	0,5
08.01.2016	Webreseach		HSN	1
08.01.2016	Møte		HSN	0,5
09.01.2016	Dokumentmal og Kooordinering og	FORPROSJEKT	HJEMME	1
09.01.2016	Rapport forprosjekt		HJEMME	3
09.01.2016	Kommunikasjon		HJEMME	1,5
10.01.2016	Sammskrivning og korrektur, kommunikasjon	FORPROSJEKT	HJEMME	3
11.01.2016	Renskrivning Rapport	FORPROSJEKT	HJEMME	2
11.01.2016	Kommunikasjon		HJEMME	0,5
13.01.2016	Kommunikasjon	Møte	HJEMME	1,5
13.01.2016	Research		HSN	3
14.01.2016	Møte	Møte	HSN	6
15.01.2016	Research	Kommunikasjon	HSN	2
16.01.2016	Telefon	Kommunikasjon	HJEMME	0,5
17.01.2016	Korrektur og mail	Kommunikasjon	HJEMME	0,5
18.01.2016	Mail	Kommunikasjon	HJEMME	0,5
19.01.2016	Brief og veiledermøte	Møte	KPS	3
20.01.2016	Mail	Kommunikasjon	HJEMME	0,5
21.01.2016	Møte kravspec		HSN	6
21.01.2016	Planlegging med scrum		HJEMME	4
22.01.2016	Møte Userstoies		HSN	5
22.01.2016	Mail skrivning	Kommunikasjon	HSN	4,5
22.01.2016	Scrumpanel	[US-070301]	HSN	2
23.01.2016	Userstories	[US-070301]	HJEMME	2
24.01.2016	Ukeplanlegging		HJEMME	3,5
26.01.2016	Møte	SPRINT PLANNING	HSN	4
27.01.2016	User Stories	User Stories	HJEMME	1
28.01.2016	Planning Sprint1	SPRINT PLANNING	HSN	6

29.01.2016	MØTE KPS	[US-070201]	KPS	4,5
29.01.2016	Planning Sprint1	[US-070301]	HSN	3
29.01.2016	Oppdatering veileder	Kommunikasjon	HJEMME	0,5
29.01.2016	Referat input	Kommunikasjon	HJEMME	1
29.01.2016	Risk Input	[US-020232]	HJEMME	1,5
30.01.2016	Referat	Kommunikasjon	HJEMME	1
30.01.2016	Scrumpanel	[US-070301]	HJEMME	4
31.01.2016	Legge inn siste formler i Sprint1	[US-070301]	HJEMME	3
01.02.2016		[US-020250]	HJEMME	4
02.02.2016	Møte	Møte	HSN	1
02.02.2016	Research software	US-021702	KPS	7
03.02.2016		[US-020250]	HJEMME	4
03.02.2016	Kommunikasjon	Adm, Telefon	HJEMME	1
04.02.2016	Kommentarer til Prosjektplan		HSN	2
04.02.2016	Møte	Møte	HSN	1
04.02.2016	Re estimering av userstories i sprint 1	[US-070501]	HSN	2
04.02.2016	Kommunikasjon	Adm, Telefon	HSN	2
04.02.2016	Kommunikasjon	Adm, Telefon	HJEMME	1
05.02.2016	Tidsestimat	[US-020251]	HSN	2
05.02.2016	Møte	Møte	HSN	1
05.02.2016	Prosjektplan	[US-020250]	HSN	4
05.02.2016	Kommunikasjon	Adm, Telefon	HJEMME	1
06.02.2016	Scrumpanel	[US-070301]	HJEMME	2
06.02.2016	Korrekturlesning	[US-020250]	HJEMME	2
07.02.2016	Korrekturlesning	[US-020250]	HJEMME	2
07.02.2016	Ukeplanlegging	Adm, Telefon	HJEMME	2
08.02.2016	Kommunikasjon	Telefon	HJEMME	2
08.02.2016	Dokumentkontroll	Møte, Lesing, Korrektur. Oppdatering av dokumentmal.	HSN	4
08.02.2016	Kommunikasjon	[US-020102]	HSN	1
09.02.2016	Arb fordeling	Møte, Lesing.	HSN	2
09.02.2016	Issu håndtering	[US-070501]	HSN	2
09.02.2016	Dokumentstruktur	Samle innhold. Lesing, Korrektur Ferdigstilling.	HSN	2
09.02.2016	Dokumentformatering	Konvertering til word. Innholdsfortegnelse etc.	HJEMME	4
10.02.2016	Møte presentasjons forberedelser	Adm, Telefon	HSN	4
10.02.2016	Skrive manus	Presentasjon	HJEMME	1

11.02.2016	Øving Møte	Presentasjon	HSN	6
11.02.2016	Bildebehandling	Dokumentasjon	HJEMME	2
12.02.2016	Møte øving og presentasjon	Adm	HSN	6
12.02.2016	Status rapport	[US-020103]	HJEMME	2
15.02.2016	Planlegning av sprint review	Adm	HJEMME	4
16.02.2016	Sprint Review	Adm	HSN	4
18.02.2016	Kommunikasjon	[US-070101]	HJEMME	1
20.02.2016	Kommunikasjon	Adm	HJEMME	1
23.02.2016	Gjennomgang av act software design	[US-040101]	HJEMME	8
25.02.2016	Møter	Adm	HSN	4
25.02.2016	Kommunikasjon	Adm	HSN	2
25.02.2016	Status til KPS	[US-020103]	HSN	5
25.02.2016	Kommunikasjon	Adm	HSN	3
29.02.2016	Kommunikasjon KPS	[US-020103]	HJEMME	1
29.02.2016	Endringer tidsplan. Ferdigstilling dokumenter.	Adm	HJEMME	4
01.03.2016	Administrasjon reestimering og gjennomgang	Adm	HSN	5
01.03.2016	Lese kode	[US-040101]	HJEMME	3
02.03.2016	Kommunikasjon	Adm	HJEMME	1
03.03.2016	Programmere på act systemet	[US-040101]	KPS	4
03.03.2016	Lese kode & UML	[US-021701]	HJEMME	4
03.03.2016	Mail og administrasjon	Adm	HSN	1,5
04.03.2016	Programmere på act systemet	[US-040101]	KPS	6
05.03.2016	Gjennomgang og planlegging av sprint review	Adm	HJEMME	3
07.03.2016	Gjennomgang og planlegging av Sprint Review	Adm	HJEMME	2
08.03.2016	Sprint Review	Adm	KPS	4
08.03.2016	Sprint Planing	Adm	KPS	3
09.03.2016	Sprint Planing	Adm	KPS	2
10.03.2016	Status rapport og veiledermøte	[US-020103]	KPS	4
10.03.2016	Sprint retrospektiv	Adm	HSN	1
10.03.2016	Sprint retrospektiv	Adm	HJEMME	2
11.03.2016	Kommunikasjon	Adm, Telefon	HJEMME	1
11.03.2016	Sprint retrospektiv	[US-020103][US-070101]		5,5
11.03.2016	Status rapport	[US-020103][US-070101]		0,5
12.03.2016	Burndown	[US-070702] Møter	HSN	5
12.03.2016	KlasseDiagram	[US-021702]		1

12.03.2016	KlasseDiagram	[US-021701]	HJEMME	2
13.03.2016	Burndown	[US-070702]	HJEMME	3
14.03.2016	Dokumentasjon	[US-020102][US-021705]	HSN	7
15.03.2016	Møte	[US-070101]	KPS	1
15.03.2016	Dokumentasjon	[US-021705]	HSN	10
16.03.2016	Dokumentasjon	[US-021705]	HSN	8
17.03.2016	Presentasjon 2 og kommunikasjon	[US-020801]	HSN	8
18.03.2016	Øving Kommunikasjon		HSN	5
18.03.2016	Presentasjon		KPS	4
07.04.2016	Burndown	[US-070702]	HSN	8
08.04.2016	Planlegging	[US-070702]	KPS	8
12.04.2016	Konfigurert en gitserver backupløsning		HJEMME	4
14.04.2016	Møte med alle,			2
14.04.2016	Forberedende arbeid med oppfølgingsdokumentasjon			4
14.04.2016	Kommunikasjon med int. ekst.			2
15.04.2016	Dokumentasjon	[US-021707]		9
16.04.2016	Lese og skrive Kode	[US-042001]		4
18.04.2016	Skolen	[US-040601][US-042001]		7
20.04.2016	Skolen	[US-040601][US-042002]		9
21.04.2016	Skolen	[US-042004] Møter og kommunikasjon.		7
22.04.2016	Kommunikasjon	Adm, Telefon	SYK HJEMME	1
23.04.2016	Kommunikasjon	[US-060501]	SYK HJEMME	1
24.04.2016	Kode	[US-060501]	SYK HJEMME	5
25.04.2016	Programmering	[US-042007][US-042001][US-060501]	SYK HJEMME	10
26.04.2016	Programmering	[US-060501]	SYK HJEMME	3
27.04.2016	Teambuilding			8
28.04.2016	Programmering	[US-060501]	KPS	10
29.04.2016	Programmering	[US-040601]	HJEMME	6
02.05.2016	Programmering	[US-040601][US-042007]	KPS	5
02.05.2016	Møte / Kommunikasjon	Adm, Telefon		3
03.05.2016	Programmering	[US-040601][US-042007]	KPS	6
03.05.2016	Møte / Kommunikasjon	Møte		2
04.05.2016	Programmering	[US-040601][US-042007]		6
04.05.2016	Møte / Kommunikasjon	Adm, Telefon		1
05.05.2016	Programmering	[US-040601][US-042007]	KPS	4

05.05.2016	Møte / Kommunikasjon	Adm, Telefon		2
06.05.2016	Ringte alle fra bilen, gjennomgang sprint 5	Samle inn forventninger	HJEMME	2
08.05.2016	Dokumentasjon og forberedelse sprint 5	[US-070501]	HJEMME	3
09.05.2016	Sprint review and planning	Planlegging av sprint5	HSN	7
10.05.2016	Skrive sprint retrospektiv	Innsamling og kommunikasjon	HJEMME	9
11.05.2016	Statusrapport Test verifikasjon	[US-020103]	HJEMME	8
12.05.2016	Møte og testing	Møte og testing	KPS	3
12.05.2016	Møte og Dokumentasjon	Veiledermøte [US-020102][US-021801]	HSN	4
12.05.2016	Kommunikasjon	Telefon	HJEMME	2
13.05.2016	Brukermanual	[US-021803][US-021801]	HJEMME	6
14.05.2016	Dokumentasjon	[US-020103][US-021801][US-021803]	HJEMME	4
15.05.2016	Rapport	[US-020103][US-070101]	HJEMME	2
16.05.2016	Dokumentasjon møte og testing.	[US-021803]	KPS	8
17.05.2016	Teknologidokument	[US-021707]	HNS	9
18.05.2016	Teknologidokument	[US-021707]	KPS	3
18.05.2016	Ferdigstilling av Teknologidokumnt	[US-021707]	HSN	5
19.05.2016	Samling av dokumentasjon	Samling av dokumentasjon[US-020701]	HSN	9
20.05.2016	STIPULERT	STIPULERT Samling av dokumentasjon	HSN	10

3. Timeliste: Aina Engen Nilsen

Totalt timeantall: 575,5

DATO	ARBEIDSBESKRIVELSE	ARBEIDSOPPGAVE (Fra tids budsjettet)	ARBEIDSSTED	TIMER BRUK T
06.10.2015	Første møte		HBV	2
13.10.2015	Møte internt, hvilken oppgave skal vi søke på.		HBV	2
14.10.2015	Skrevet søknad og cv	Dokumentasjon / Søknad	HJEMME	2
20.10.2015	Møte internt	SØKNAD	HANGOUTS	0

03.11.2015	MØTE INTERNT	SØKNAD	HBV	2
09.11.2015	MØTE INTERNT	OPPGAVE	HBV	2
23.11.2015	Møte nr 1 hos KPS informasjon om oppgaven	MØTE	KPS	3
01.12.2015	Forberedende møte Internt	MØTE	HBV	2
02.12.2015	Møte nr 2 hos KPS intensjon	MØTE	KPS	3
04.01.2016	Test hos kps	RESEARCH, referat	KPS/hjemme	5
05.01.2016	GRUPPEMØTE	DOKUMENTASJON	HSN	7
06.01.2016	Hjemmekontor	Nødvendig programvare,div administrativt,skisser mulig design	HJEMME	9
08.01.2016	Møte internt etter infomøte skolen	Planlegging, referat	HSN/Hjemme	1,5
09.01.2016	Forprosjektrapport utforming, SW	Dokumentasjon samskriving, 2 skisse (part) SW	HSN/Hjemme	6
11.01.2016	Mal oppfølg.dok, dokumentasjon, SW	Dokumentasjon, nødvendig programvare feilrett	HSN/Hjemme	5
13.01.2016	Forprosjektrapport gjennomgang, dokumentasjon	Dokumentasjon, retting og omskriving	HSN/hjemme	4
14.01.2016	GRUPPEMØTE	Planlegging, omstrukturering, dokumentasjon	HSN	7,5
15.01.2016	GRUPPEMØTE	Kontrollrapport, Intensjonsavtale, statusrapport	HSN	12
16.01.2016	Hjemmekontor	Retting dok: se 15.01., Research RISK	Hjemme	3,5
18.01.2016	Hjemmekontor	RISK	Hjemme	4,5
19.01.2016	MØTE KPS, MØTE intern veileder	Sikkerhet og klarering, 1. møte intern veileder	KPS/HSN/hjemme	5,5
20.01.2016	Hjemmekontor (syk)	Risk (2), forbereder jobbing med pros. plan i morgen	Hjemme	4
21.01.2016	GRUPPEMØTE	Prosjektplan utforming	HSN rom 1208	6,5
22.01.2016	GRUPPEMØTE	User stories, oppfølgingsdokument, referater	HSN rom 1208/hjemme	9

23.01.2016	Hjemmekontor	Leser Scrum/ video	Hjemme	2,5
26.01.2016	GRUPPEMØTE	SCRUM, planlegging og arbeid med oppg	HSN rom 1208	4
26.01.2016	Hjemmekontor	RISK, referat, oppfølgingsdok Sigmund	Hjemme	3
27.01.2016	Hjemmekontor	RISK	Hjemme	3
28.01.2016	GRUPPEMØTE	User stories, acceptanse criterias, veiledermøte, ref. veiledermøte	HSN 1208	7
28.01.2016	Hjemmekontor	RISK, referat.	Hjemme	3
29.01.2016	Hjemmekontor (dokumentert fravær)	RISK, SW	Hjemme	6
31.01.2016	Hjemmekontor	Administrativt, RISK	Hjemme	7
01.02.2016	HSN	Risk	HSN 1208/1214	2
02.02.2016	GRUPPEMØTE	RISK (3,5), Prosjektplan	HSN 1208	6,5
03.02.2016	Hjemmekontor	Risk (2) , Prosjektplan	Hjemme	3
04.02.2016	GRUPPEMØTE	Prosjektplan	HSN 1208	4,5
		Felles møte, oppdatering sprint 1		1
		RISK		1
05.02.2016	Kjernetid gruppe, sprint1	Prosjektplan	HSN 1208	7
		Internt veiledermøte		0,5
	Hjemmekontor	Prosjektplan og administrativt (ref og innkalling)	Hjemme	5
06.02.2016	Hjemmekontor	Prosjektplan	Hjemme	5
		Oppfølgingsdokument	Hjemme	0,5
07.02.2016	Hjemmekontor - innspurt dokumentasjon	Prosjektplan og vedlegg risiko(1)	Hjemme	4
08.02.2016	Finpusser på dokumentasjon HSN	Prosjektplan og vedlegg risiko(1)	HSN 1208	5
	Hjemmekontor - innspurt dokumentasjon	FERDIGSTILLING prosjektplan og risiko (0,5) m.m.	Hjemme	2
09.02.2016	Gjennomgang og godkjenning Prosjektplan	Gjennomgang og godkjenning Prosjektplan	HSN 1208	4
10.02.2016	Manus til presentasjon	Manus til presentasjon	Hjemme	2,5

11.02.2016	1. presentasjon	Manus, øving, sammenstilling	HSN 1208	5,5
	Referat intern veileder	Referat	Hjemme	0,5
	Finpuss manus 1. pres	Manus til presentasjon	Hjemme	2
12.02.2016	Forberedelse/øving 1. presentasjon	Øving 1. presentasjon	HSN 1208/ Kinosal	4
	1. presentasjon og forberedelser	1. presentasjon	HSN	3
	Status risiko, oppf. dok, referat 1. pres	Dokumentasjon	Hjemme	2
16.02.2016	Sprint Review 1	Gjennomgang Sprint 1.	HSN	2,5
	Referat Sprint Review	Referat	Hjemme	1,5
17.02.2016	Konvertere PDF til SW fil	SolidWorks - filer til 2D CG	Hjemme	4
18.02.2016	Gruppemøte, sprint2. Gjennomgang Us/Ac	Klargjør Us/Ac for sprint 2 i plenum, litt oppklaringer	HSN rom 1208	6
19.02.2016	Gjennomgang Us/Ac	Jobber videre med Us/Ac, noe research styringskomponenter EE	HSN rom 1208	6
		Referat, research, SW	Hjemme	3
20.02.2016	Research SW	Konvertere PDF til SW, noe research	Hjemme	3
21.02.2016	SW	Forsøk konvertere filer	Hjemme	2
		SW tegning	Hjemme	1
22.02.2016	Konseptutvelgelse Styring EE (mekatronikk)	El og maskin møte for gjennomgang av konsepter	HSN rom 1208	5
	Modellert opp CG dummy i SW	SW tegning	Hjemme	3
23.02.2016	Risiko	Risiko oppdatering	Hjemme	2
25.02.2016	Risiko til statusrapport, håndtering risk	Risikohåndtering og risk til statusrapport	HSN rom1208	3
	Referat internveileder	Referat	HSN rom 1208	0,5
	CAD, konseptdiskusjon	EE konsept styring/mekanikk	HSN rom 1208	2
	Føringer for risiko, risiko til statusrapp	Risiko	Hjemme	2
26.02.2016	Oppfølgingsdokument		HSN rom 1208	1
	Oppdatert risiko vedlegg B		HSN rom 1208	3
	Dokumentasjon		Hjemme	2

29.02.2016				
01.03.2016	Gruppemøte før undervisning. Oppdatert sprint 2	EE research, dokumentasjon	HSN 1208	2
		Gjennomgang timer, oppdatering sprint 2	HSN 1208	1
02.03.2016	Jobber med konsept EE – CAD	US-051401 Konsept EE	HSN 1208	2
		US-04 400201 Design EE	HSN 1208	3
03.03.2016	Design EE - CAD	US-04 400201 Design EE	HSN 1208	6
		US-04 400201 Design EE	Hjemme	2
04.03.2016	Design EE - CAD	US-04 400201 Design EE	HSN 1208	6
05.03.2016	Design EE - CAD (og konsept)	US-051401 Konsept EE offline	Hjemme	2
06.03.2016	Design EE, kommunikasjon	US-051401 Konsept EE	Hjemme	3
		US-080102 Simulering forsøk		1
		US-044004 Anordning håndflate endring		1
		Research TEST (for dokumentasjon)		3
07.03.2016	CAD Simulering	US-080102 Simulering forsøk	Hjemme	0,5
	Dokumentasjon	Research og noe tekst ment for testverifikasjon	Hjemme	4
08.03.2016	Sprint review/ diverse KPS	Sprint review	KPS	3,5
	CAD design/ utbedring/simulering	US-080102 Simulering forsøk	Hjemme	3
09.03.2016	Sprint planlegging		KPS	3
	Risiko. Møte med Merethe, fått verktøy	US-070901 oppdatering risiko	KPS	5
10.03.2016	Risiko Møte, gjennomgang med gruppa	US-070901 oppdatering risiko	KPS	1
	Internt veiledermøte		KPS	1
	Risiko oppdatering og verktøy	US-070901 Oppdatering risiko	KPS	5
	Risiko til statusrapport	US-020103 Innhold statusrapport	Hjemme	1
	Møtereferat intern og oppfølgingsdokument	US-020102 Ukentlig statusrapport	Hjemme	1

11.03.2016	Testverifikasjonsdokument	US-021001 Testverifikasjonsdokument	KPS	4
	Design EE - konsept (endring grunnet servo), oversikt CG	US-051401 Konsept EE	KPS	1,5
12.03.2016	Testverifikasjonsdokument	US-021001 Testverifikasjonsdokument	Hjemme	2
13.03.2016	Testverifikasjonsdokument	US-021001 Testverifikasjonsdokument	Hjemme	3
14.03.2016	Konsept EE beregninger og målinger	US-051401 Konsept EE	KPS	3
	Testverifikasjonsdokument	US-021001 Testverifikasjonsdokument	KPS	1
15.03.2016	Testverifikasjonsdokument	US-021001 Testverifikasjonsdokument	HSN 1208	5,5
	Testverifikasjonsdokument	US-021001 Testverifikasjonsdokument	Hjemme	4
	CAD 3Dfiler, forbered bilder dok/møte	US-051401 Konsept EE	Hjemme	1
	Research og oppdatering på kildehenvisning.	Skal oppdatere prosjektplanen for levering	Hjemme	1
16.03.2016	Oppdatere dok og overlevere testverifik.dok	US-021001 Testverifikasjonsdokument (ferdigstilling)	Hjemme	2
	Manus 2. presentasjon	US-020403 Manus fremføring	Hjemme	0,5
17.03.2016	Møter, manus 2. pres	US-020403 Manus fremføring	HSN 1208	0,5
		Møter (intern veileder, gruppe, Scrum)	HSN 1208	6
18.03.2016	Øving	[US-020404] Øving fremføring.	HSN 1208 og andre	4
	Referat intern veileder	Ble skrevet 19.03 grunnet arbeidsmengde 18.03	Hjemme	0,5
	2. presentasjon	[US-020401] Fra formøte start til individuelt ettermøte ferdig	HSN datarom	2
07.04.2016	US gjennomgang og rangering		Hjemme (sykt barn)	2,5
08.04.2016	US Sprint 4		HSN 1208	6,5
11.04.2016	Sprint planlegging (AC, tidsestimering), Dok	[US-020102] Oppfølgingsdokument 10	HSN 1208	1
		Sprint 4 planlegging	HSN 1208	5,5
12.04.2016	CAD. Tilpasse finger ferdig til ny motor for print	[US-010701] Prototype (klargjøring første print)	HSN 1208	6,5

		[US-010701] Prototype (klargjøring første print)	Hjemme	1,5
13.04.2016	Cad tilpasning og forberede print	[US-010701] Prototype (klargjøring første print)	HSN 1208 / Devotec	3
	Risiko	[US-070901] Risikodokument oppdatert	HSN 1208	2
	Risiko, Forberede gjennomgang og statusrapp	[US-070901] Risikodokument oppdatert	Hjemme	2
14.04.2016	Risiko gjennomgang plenum, risk statusrapp	[US-070901] Risikodokument oppdatert	HSN 1208	4
	Prototype: første finger til print. Litt tid på dette	[US-010701] Prototype (under print)	HSN 1208	1
	Risiko til statusrapp, oppdatering hoveddokument	[US-070901] Risikodokument oppdatert	Hjemme	3
15.04.2016	Korrektur		Hjemme	0,5
18.04.2016	Oppfølgingsdokument 11	[US-020102] Oppfølgingsdokument	Hjemme (sykdom)	1
	Samling av teknisk informasjon	[US-021507] Teknologidokument. Innholds generering		3
19.04.2016	CAD/ FEM "opprydning" og analyse	[US-010701] Prototype	Hjemme (sykdom)	2
		[US-021507] Teknologidokument. Innholds generering		4
20.04.2016	Samling av teknisk informasjon	[US-021507] Teknologidokument. Innholds generering	Hjemme (sykdom)	
21.04.2016	Små endringer CAD, klargjøring ny print.	[US-010701] Prototype	HSN 1208	4
	Referat intern veil. møte	[US-020102] Oppfølgingsdokument (referat)	HSN 1208	0,5
22.04.2016	Oppfølgingsdokument 12	[US-020102] Oppfølgingsdokument	Hjemme	1
25.04.2016	Endringer og print	[US-010701] Prototype	HSN 1208	6
26.04.2016	Generering dok	[US-021507] Teknologidokument. Innholds generering	HSN 1208/KPS	2
	Endringer og print	[US-010701] Prototype	HSN 1208/KPS	4
28.04.2016	Teknisk dokumentasjon TEST	[US-021002] Testverifikasjonsdokument generering	KPS	1
	Endringer og print	[US-070901] Oppdatering risiko	KPS	3
	Gjennomgang risk plenum	[US-070901] Oppdatering risiko	KPS	0,5

	Risk til Statusrapport	[US-020103] Statusrapport	KPS	1
	Gjennomgang tidsestimering sprint 4		KPS	1
	Oppfølgingsdokument 13	[US-020102] Oppfølgingsdokument	KPS/hjemme	1
29.04.2016	Oppdatering riskdokumenter	[US-070901] Oppdatering risiko	Hjemme	2
	Generering tekdok	[US-021507] Teknologidokument. Innholdsgenerering	Hjemme	2
30.04.2016	Samler tråder for teknisk dok	[US-021507] Teknologidokument. Innholdsgenerering	Hjemme	2
02.05.2016	Lager tabeller og setter inn tekst i dok	[US-021002] Testverifikasjonsdok generering	KPS	4
			Hjemme	2
03.05.2016	Forberede print kabinett, noe testverifikasjon	[US-021002] Testverifikasjonsdokumentgenererin g	HSN 1208	1
		[US-010701] Prototype	HSN 1208	4
		[US-021507] Teknologidok. innholdsgenerering	Hjemme	1
04.05.2016	Risiko tilknytning, generering innhold	[US-021002] Testverifikasjonsdok generering	KPS	3
		[US-070901] Oppdatering risiko	KPS/hjemme	2
05.05.2016	Forbered print smal finger, pute til håndflate	[US-010701] Prototype	Hjemme	3
	Samling av tråder mot dokumentasjon	[US-021507] Teknologidok. Innholdsgenerering	Hjemme	2
06.05.2016	Testverifikasjon (dok innsamling) og fysisk test	[US-021002] Testverifikasjonsdok generering	KPS	3
	Print smal finger	[US-010701] Prototype	HSN	2
08.05.2016	Oppfølgingsdok, risiko	[US-020102] Oppfølgingsdokument	Hjemme	1
	Risiko mot test - testverifikasjon	[US-070901] Oppdatering risiko	Hjemme	2
09.05.2016	Sprint review og planlegging dok sprint		HSN 1208	6,5
10.05.2016	Teknisk innhold for finger (noe design). Rydding i filer for oversikt i SW		Hjemme	8
11.05.2016	Testverifikasjon (for maskin, samt rettet endel feil)	[US-021003] Testverifikasjonsdokument	Hjemme	6
	Risiko (til statusrapp, forberedt gjennomgang.)	[US-070901] Risiko		3

12.05.2016	Teknisk innhold: beregninger og analyser.	[US-]	KPS/HSN/hjemme	5
	Risiko oppdatering verktøy og analyse	[US-070901] Risiko	Hjemme	3
	Oppfølgingsdokument og møtereferat	[US-020102] Oppfølgingsdok og ref	HSN/hjemme	1
13.05.2016	Arbeid med CAD - filer	[US-021508] Teknologidok EE innhold	Hjemme	9
14.05.2016	Risiko dokument for innlevering	[US-070901] Risiko	Hjemme	2
15.05.2016	Arbeid med Masseegenskaper og beregninger	[US-021508] Teknologidok EE innhold	Hjemme	8
16.05.2016	Tekdok innhold for finger disponering og tillegg	[US-021508] Teknologidok EE innhold	KPS	5,5
	Fremtidsdokument		Hjemme	0,5
	Tekdok innh finger beregninger og justeringer	[US-021508] Teknologidok EE innhold	Hjemme	5
17.05.2016	tekdok finger FEM inn	[US-021508] Teknologidok EE innhold	Hjemme	5
	Start refleksjonsdok		Hjemme	0
18.05.2016	tekdok finger FEM inn	[US-021508] Teknologidok EE innhold	KPS	5
	Tekdok finger ferdig til disponering	[US-021508] Teknologidok EE innhold	Hjemme	1
	Risikodokument stort ferdigstilles	[US-070901] Risiko	Hjemme	7
19.05.2016	Refleksjonsdok		Hjemme	2
	Testverifikasjonsdok ferdigstilles	[US-021003] Testverifikasjonsdokument	HSN/hjemme	8
20.05.2016	Korrektur og sammenstilling dokumentasjon		HSN	6
	LEVERING DOKUMENTASJON		HSN	1

4. Timeliste: Bård Simen Hamborg Enget

Totalt timeantall: 555

DATO	ARBEIDSBESKRIVELSE	ARBEIDSOPPGAVE (Fra tids budsjettet)	ARBEIDSSTED	TIMER BRUKT
06.10.2015	Første møte		HBV	2
13.10.2015	Møte internt, hvilken oppgave skal vi søke på.		HBV	2
14.10.2015	Skrevet søknad og cv	Dokumentasjon / Søknad	HJEMME	3
20.10.2015	Møte internt	SØKNAD	HANGOUTS	2
03.11.2015	MØTE INTERNT	SØKNAD	HBV	2
09.11.2015	MØTE INTERNT	OPPGAVE	HBV	2
23.11.2015	Møte nr 1 hos KPS informasjon om oppgaven	MØTE	KPS	3
01.12.2015	Forberedende møte Internt	MØTE	HBV	2
02.12.2015	Møte nr 2 hos KPS intensjon	MØTE	KPS	3
04.01.2016	Demo hos KPS	RESEARCH	KPS	5
05.01.2016	Møte	DOKUMENTASJON	HSN	3
06.01.2016	Tids Estimert Dokument	DOKUMENTASJON	HJEMME	2
07.01.2016	UNDERVISNING		HBV	
08.01.2016	UNDERVISNING		HBV	
10.01.2014	Konseptbeskrivelse	DOKUMENTASJON	HJEMME	4
11.01.2014	Konseptbeskrivelse	DOKUMENTASJON	HJEMME	1
13.01.2014	Research		HJEMME	2
14.01.2016	Møte	MØTE	HSN	6
15.01.2016	Møte	KONTROLLOKUMENT	HSN	8
16.01.2016	DOKUMENTASJON GJENNOMGANG	KONTROLLOKUMENT, INTENSJONSAVTALE	HJEMME	2
17.01.2016	DOKUMENTASJON GJENNOMGANG	KONTROLLOKUMENT, INTENSJONSAVTALE	HJEMME	1
19.01.2016	MØTE	KPS, SIKKERHETSBRIF	KPS	2
	MØTE	VEILEDERMØTE	HSN	2
20.01.2016	RESEARCH	KRAVSPEC	Hjemme	3
21.01.2016	KRAVSPEC	KRAVSPEC	HSN	6
	SCRUM	PROSJEKT STYRING	Hjemme	2
22.01.2016	User Stories	USER STORIES	HSN	6
	Korrektur	KOMMUNIKASJON	Hjemme	1
23.01.2016	RESEARCH	PROSJEKT STYRING	Hjemme	3
24.01.2016	RESEARCH	PROSJEKT STYRING	Hjemme	3

25.01.2016	RESEARCH	PROSJEKT STYRING	Hjemme	2
26.01.2016	Møte	SPRINT PLANNING	HSN	4
27.01.2016	User Stories	User Stories	Hjemme	5
28.01.2016	Planning Sprint1	SPRINT PLANNING	HSN	6
29.01.2016	MØTE KPS	Møte	KPS	4
	Planning Sprint1	SPRINT PLANNING	HSN	1
30.01.2016	RISK Input	DOKUMENTASJON	Hjemme	1
31.01.2016	Status rapport til KPS	[US-020103]	Hjemme	1
01.02.2016	User Stories	DOKUMENTASJON	Hjemme	2
02.02.2016	Møte	Scrum Møte + litt felles	HSN	1
		[US-020240] [US-020241] [US-020242] [US-020243] [US-020244] (User Stories)		
02.02.2016	Sprint 1	(User Stories)	HSN	3
		[US-020240] [US-020241] [US-020242] [US-020243] [US-020244] (User Stories)		
03.02.2016	Sprint 1	(User Stories)	HSN	6
03.02.2016	Rutiner	DOKUMENTHÅNDTERING	Hjemme	2
		[US-020240] [US-020241] [US-020242] [US-020243] [US-020244] (User Stories)		
04.02.2016	Sprint 1	(User Stories)	HSN	4
	Møte	[US-070501]	HSN	2
		[US-020240] [US-020241] [US-020242] [US-020243] [US-020244] (User Stories)		
05.02.2016	Sprint 1	(User Stories)	HSN	5
	Veiledermøte		HSN	1
		[US-020240] [US-020241] [US-020242] [US-020243] [US-020244] (User Stories)		
06.02.2016	Ferdigstilling	(User Stories)	Hjemme	5
07.02.2016	Prosjektplan	[US-020210]	Hjemme	2
08.02.2016	Prosjektplan	[US-020210]	HSN	2
09.02.2016	Arb fordeling presentasjon	[US-020301]	HSN	2
	Presentasjonsforbredelser	[US-020303], [US-020302]	Hjemme	2
	Møte presentasjons forberedelser	[US-020304]		
10.02.2016		[US-020304]	HSN	4
	Skrive/pugge manus	[US-020303], [US-020302]	HJEMME	4
11.02.2016	Øving Møte	[US-020304]	HSN	6
12.02.2016	Møte øving og presentasjon	[US-020304], [US-020301]	HSN	6
16.02.2016	Sprint Review	Sprint Review	HSN	4
17.02.2016	Sprint2 Planlegging	Velge ut US	Hjemme	4

18.02.2016	Sprint2 Planlegging	Velge ut US, tidsestimering	HSN	6
19.02.2016	Sprint2 Planlegging	Velge ut US, tidsestimering, AC	HSN	6
20.02.2016	Sprint2 Planlegging	Velge ut US, tidsestimering, AC	Hjemme	3
21.02.2016	Sprint2 Planlegging	[US-070801], Velge ut US, AC, Redigere og legge til i Sprint2 sheet	Hjemme	5
23.02.2016	Sprint 2 Arduino	[US-046104]		2
24.02.2016	Sprint 2 Arduino	[US-046104]		2
25.02.2016	Møte	Veiledermøte,		4
	Sprint 2 Arduino	[US-046202]		2
26.02.2016	Risk oppdatering/søknad			2
	Sprint 2 Arduino	[US-046302]		3
28.02.2016	Arduino	[US-046202]		2
29.02.2016	Arduino	[US-046302]		2
01.03.2016	Møte			4
02.03.2016	WS-Sensor	[US-046302]		5
03.03.2016	WS-Sensor	[US-046102]		3
	WS-Sensor	[US-046103]		4
04.03.2016	WS-Sensor	[US-046302]		6
05.03.2016	WS-Sensor	[US-046103]		4
06.03.2016	WS-Sensor	[US-02160101]		5
07.03.2016				
08.03.2016	Møte	Sprint2 Review		4
	Sprint 3 Planning	User Stories utvelging		4
09.03.2016	Sprint 3 Planning	User Stories utvelging		2
	Møte	Sprint3 Planning		5
10.03.2016	Sprint3, User Stories	[US-070801]		5
	Møte, veileder			1
	Møte risk	[US-070901]		1
11.03.2016				
12.03.2016	Sprint 3, Burn Down chart	[US-070702]		5
13.03.2016	Sprint3, User Stories	[US-070801]		3
14.03.2016	Sprint3, User Stories	[US-070801]		7

15.03.2016	Møte Burndown	[US-070702]		1
	Møte Servo			1
	Sprint3, User Stories	[US-070801]		4
	Testverifikasjon dokument	[US-021001]		1
	Prosjektplan oppdatering			1
	Korrektur Elektro tek.doc			2
16.03.2016	Korrektur Elektro tek.doc			2
	Ferdigstilling og levering			4
17.03.2016	Planlegge presentasjon	[US-020402], [US-020403]		3
	Veileder møte			1
	Gruppemøte			3
	Manus, og slides presentasjon	[US-020402], [US-020403]		3
18.03.2016	Øving og forberedelse til presentasjon	[US-020404]		6
	Presentasjon og ettermøte	[US-020401]		2
06.04.2016		Sprint 3 review	Hjemme	1
07.04.2016		Sprint 3 review, Sprint 4 Planning, utvalgelse	HSN	7
08.04.2016		Sprint 4 Planning, AC, US dok	HSN	5
09.04.2016				
10.04.2016		Sprint 4 Planning, US dok	Hjemme	1
11.04.2016		Sprint 4 Planning, dokument forberedelse	HSN	6
12.04.2016	EE config	[US-044002], [US-044005],[US-044102],[US-044103],	HSN	6
13.04.2016	EE config	[US-044002], [US-044005],[US-044102],[US-044103],	HSN	7
14.04.2016	EE config	[US-044002], [US-044005],[US-044102],[US-044103],	HSN	6
15.04.2016				
18.04.2016		[US-046103]	HSN	6
19.04.2016		[US-046103]	HSN	6
20.04.2016		[US-046103]	HSN	7
21.04.2016		[US-02160101]	HSN	10
22.04.2016		[US-02160101]	Hjemme	2

25.04.2016		[US-02160101], [US-02160201]	KPS	6
26.04.2016		[US-043101]	KPS	7
27.04.2016		Teambuilding	Meheia	
28.04.2016		[US-043101]	KPS	8
29.04.2016		[US-043101]	KPS	6
		[US-043101], [US-046103]	Hjemme	2
02.05.2016	UR5-MEGA kommunikasjon		KPS	6
			Hjemme	2
03.05.2016	Sammenstilling WSS, lodding + test	[US-046103]	HSN + Hjemme	8
04.05.2016	UR5-MEGA kommunikasjon	[US-044002], [US-044005],[US-044102],[US-044103], [US-043101]	KPS	7
	Ferdigstilte WSS, Problemløsning UR5-Mega com	[US-044002], [US-044005],[US-044102],[US-044103], [US-043101]	Hjemme	5
05.05.2016	UR5-MEGA kommunikasjon	[US-044002], [US-044005],[US-044102],[US-044103], [US-043101]	KPS	5
	Reinskrevet ARDUINO kode	[US-044002], [US-044005],[US-044102],[US-044103], [US-043101]	Hjemme	7
06.05.2016	WSS test, Dokumentasjon av WSS.	[US-02160101], [US-046103]	KPS	7
	Dokumentasjon WSS	[US-02160101]	Hjemme	2
07.05.2016				
08.05.2016				
09.05.2016	Sprint Reveiw Sprint 4		HSN	6
	Info sanking, uten resultater	[US-046401]	KPS	2
	Lodding og kretstegning UR5-Mega	[US-060301]	Hjemme	3
10.05.2016	WSS dokumentasjon	[US-047001]	KPS	5
	WSS dokumentasjon	[US-047001]	Hjemme	3
11.05.2016	WSS sammenstilling	[US-046401]	KPS	5
	WSS sammenstilling	[US-046401]	Hjemme	4
12.05.2016	Tesing WSS	[US-046401]	KPS	3
	Møte Veileder		HSN	1
	Testing WSS	[US-046401]	KPS	4
	Oppdatering kode WSS	[US-046401]	Hjemme	3
13.05.2016	WSS dokumentasjon, testing, ferdigstilling kode	[US-047001]	KPS	7

14.05.2016				
15.05.2016	WSS dokumentasjon	[US-047001]	Hjemme	8
16.05.2016	US dokument	[US-070801]	KPS	6
17.05.2016	Input til testverifikasjons dok		Hjemme	4
18.05.2016	Input til EE dok, and arduino kode	[US-043101]	KPS	5
	US dokument	[US-070801]	KPS	4
19.05.2016	Dokumentasjon ferdigstilling		HSN	6
	Siste test UR5-arduino		KPS	2
	Prosjekt evaluering		Estimert	3
20.05.2016	Dokumentasjon ferdigstilling		Estimert	8

5. Timeliste Carl Martin Mausest

Totalt timeantall: 491,5

DATO	ARBEIDSBESKRIVELSE	ARBEIDSOPPGAVE (Fra tids budsjettet)	ARBEIDSSTED	TIMER BRUKT
06.10.2015	Første møte		HBV	2
13.10.2015	Møte internt, hvilken oppgave skal vi søke på.		HBV	2
14.10.2015	Skrevet søknad og cv	Dokumentasjon / Søknad	HJEMME	4
20.10.2015	Møte internt	SØKNAD	HANGOUTS	2
03.11.2015	MØTE INTERNT	SØKNAD	HBV	2
09.11.2015	MØTE INTERNT	OPPGAVE	HBV	2
23.11.2015	Møte nr 1 hos KPS informasjon om oppgaven	MØTE	KPS	3
01.12.2015	Forberedende møte Internt	MØTE	HBV	2
02.12.2015	Møte nr 2 hos KPS intensjon	MØTE	KPS	3
04.01.2016	Test hos kps	RESEARCH	KPS	5
04.01.2016	Research/skissering av verktøy	RESEARCH	HJEMME	2
05.01.2016	Møte	DOKUMENTASJON	HSN	6
05.01.2016	Research av kontrollere	RESEARCH	HJEMME	2
12.01.2014	Research		HJEMME	2
14.01.2016	Møte	Møte	HSN	6

15.01.2016	Møte	DOKUMENTASJON	HSN	6
19.01.2016	Brief og veiledermøte	Møte	KPS/HSN	4
21.01.2016	Møte	DOKUMENTASJON	HSN	6
21.01.2016	Research	Prosjekt model	HJEMME	3
22.01.2016	DOKUMENTASJON	DOKUMENTASJON	HSN	7
22.01.2016	Mail skriving		HJEMME	1
24.01.2016	DOKUMENTASJON	User story	HJEMME	2
26.01.2016	DOKUMENTASJON/Scrum møte	User story	HSN	4
27.01.2016	Research/dokumentasjon	User story/prosjektplan	HJEMME	5
28.01.2016	DOKUMENTASJON/Scrum møte	User story	HSN	6,5
28.01.2016	DOKUMENTASJON	Prosjektplan/quality	HJEMME	2,5
29.01.2016	MØTE KPS	Møte	KPS	4,5
	Planning Sprint1	SPRINT PLANNING	HSN	2
30.01.2016	DOKUMENTASJON	Risk	HJEMME	1
02.02.2016	Research	Kvalitetshåndtering i Scrum	HSN	3
03.02.2016	Research	Kvalitetshåndtering i Scrum	HSN	7
04.02.2016	Møte		HSN	1
	Kommunikasjon		HSN	1
	Re estimering av userstories i sprint 1	[US-070501]	HSN	2
	Presentasjon	[US-020301] [US-020303]	HSN	3
05.02.2016	Presentasjon	[US-020302]	HSN	6
	Prosjektplan	[US-020210]	HJEMME	2
08.02.2016	Prosjektplan	Lese over prosjektplan US og Risk	HJEMME	2
09.02.2016	Prosjektplan	Lese over prosjektplan US og Risk	HSN	2
10.02.2016	Presentasjon			7
11.02.2016	Presentasjon			7
12.02.2016	Presentasjon			7
16.02.2016	Sprint Review 1	Gjennomgang Sprint 1.	HSN	2,5
17.02.2016	Research mekanisme	US-050103	HJEMME	2
18.02.2016	Research mekanisme	US-050103	HJEMME	5
19.02.2016	Teknologidokument komponenter	US-050103	HJEMME	2
20.02.2016	Teknologidokument komponenter/research stepper motor	US-050103	HJEMME	2
21.02.2016	Teknologidokument	US-050103	HJEMME	3
22.02.2016	Research		HSN	1

	Konseptutvelgelse Styring EE (mekatronikk)	El og maskin møte for gjennomgang av konsepter	HSN rom 1208	5
	Konseptutvelgelse Styring EE (mekatronikk)	Konsept-tegning av motor og EE sammen.	HJEMME	3
25.02.2016	Møte, veiledermøte, konsept, statusrapp		HSN	6
26.02.2016	Konsept av styring til EE	SW testing av integrering av motor i EE	HSN	6
27.02.2016	Tekniskdokument		HJEMME	2
28.02.2016	Tekniskdokument		HJEMME	2
01.03.2016	Sprint 2 estimering av US		HSN	1
	Tekdok. Komponenter		HSN	2
02.03.2016	Tekdok. Komponenter		HSN	7
03.03.2016	Tekdok motorer	[US-051001]	HSN	7
04.03.2016	Tekdok motorer	[US-051001]	HSN	6
07.03.2016				
08.03.2016	Møte	Sprint2 review	HSN	4
09.03.2016	Sprint planlegging		KPS	7
10.03.2016	Sprint 3		KPS	5
	Møte veileder.			1
	Møte risk	[US-070901]		1
11.03.2016	Research		KPS	6
	Planlegging møte Servo gutta hos KPS	[US-051501]		1
12.03.2016		[US-051202]	HJEMME	4
13.03.2016		[US-021506]	HJEMME	3
14.03.2016		[US-021506]	HSN	6
15.03.2016	Møte Burndown			1
	Møte Servo	[US-051501]		1
	Teknisk dokument elektronikk	[US-051202]		7
16.03.2016	Teknisk dokument elektronikk	[US-051202]		5,5
	Revidering og gjennomlesing av dokumenter for innlevering til 2.Presentasjon.			3,5
17.03.2016	Planlegge presentasjon			3
	Veileder møte			1
	Gruppemøte			3
18.03.2016	Øving og forberedelse til presentasjon			7
	Presentasjon og ettermøte			2

07.03.2016		Sprint 3 review, Sprint 4 Planning, utvalgelse		7
08.04.2016		Sprint 4 Planning, AC, Tidsestimering	HSN	5
11.04.2016	Prototype tilpassing av servomotor	[US-010701]	HSN	6
12.04.2016	Prototype tilpassing av servomotor	[US-010701]	HSN	6
13.04.2016	Prototype tilpassing av servomotor		HSN	6
14.04.2016	Prototype tilpassing av servomotor		HSN	6
15.04.2016	Gjennomlesing av dokumenter		HJEMME	1
16.04.2016				
17.04.2016				
18.04.2016	Kommunikasjon prototype	[US-010701]	HSN	4,5
	Blokkdiagram for dokumentasjon.	[US-021507]	HSN	2
19.04.2016	Oppdatering på Elektronikk dokument.	[US-021507]	HSN	5
20.04.2016				
21.04.2016	Skjematisk tegning Cadence	[US-021507]	HSN	5,5
	Veiledermøte		HSN	0,5
22.04.2016	Skjematisk tegning Cadence	[US-021507]	HSN	5
23.04.2016				
24.04.2016	Tilføyning av innhold på teknologidokument. Mistet dokumentasjon. på v1.2 prøvd å gjenopprettete dette..	[US-021507]	HJEMME	4,5
25.04.2016	Mottakelse av Servomotorer, samt implementering av disse til printet design.		KPS	6
		[US-010701]	HJEMME	2,5
26.04.2016	Videre tilpassing av Prototype	[US-010701]	HSN	6
27.04.2016	Tilpassing av prototype/endring	[US-010701]	Hjemme	3,5
28.04.2016	Testing av EE testet EE mot CG	[US-044005]	KPS	6
29.04.2016	Testing av EE laget kabler	[US-044005]	KPS	6
30.04.2016				
01.05.2016		[US-044103]	HJEMME	1,5
02.05.2016	Montert klart EE til UR samt testet at alle servoer får forsyning fra lab strømforsyningen.	[US-044002](4t) [US-044005](2t) [US-044102]	KPS	6
03.05.2016	Testet og notert ned posisjon til servomotorer og UR5, klart for programmering og test av sekvenser.	[US-044002] [US-044005] (2t) [US-044102](3t)	KPS	5

04.05.2016	Tilpassing av EE montert på UR5	[US-044002] [US-044005] [US-044102]	KPS	6
05.05.2016	Dokumentasjon	[US-21507]	HJEMME	2
06.05.2016	Tilpassing av Kabinett til EE og UR5.	[US-044002] [US-044005] [US-044102]	KPS	6
07.05.2016				
08.05.2016				
09.05.2016	Sprint review. Sprint 5 planing.		HSN	6
	Tilføyning og endring i Elektronikk dokumentet	[US-021508]	HJEMME	3
10.05.2016	Monterings manual	[US-021806]	Hjemme	6
11.05.2016	Tek dok elektronikk	[US-021508]	Hjemme	6
	Tekdok elektronikk	[US-021508]	Hjemme	2
12.05.2016	Montering av kabinett	[US-010101]	KPS	6
	Tekdok elektronikk	[US-021508]	Hjemme	2
13.05.2016	Sugru/kabel fix	[US-010101]	KPS	3
	Dokumentasjon testverifikasjon	[US-021003]	Hjemme	3
	Assembly manual	[US-021806]	Hjemme	1
	solidworks bilde for manual		Hjemme	2
14.05.2016	Assembly manual	[US-021806]	Hjemme	4
	solidworks bilde for manual		Hjemme	4
15.05.2016	Assembly manual	[US-021806]	Hjemme	4,5
	solidworks bilde for manual		Hjemme	3
16.05.2016	Assembly manual	[US-021806]	KPS	5
	Test EE,	[US-010101]	KPS	1
17.05.2016	17. Mai hipp hipp hurra!			
18.05.2016	Elektronikk dokument	[US-021508]	KPS	5,5
	Testverifikasjon	[US-021003]	Hjemme	3
19.05.2016	Dokumentasjon	Ferdigstilling	HSN	6
	Dokumentasjon	Ferdigstilling	Hjemme	3
20.05.2016	Dokumentasjon	Ferdigstilling, Stipulert	HSN	6

6. Timeliste Monica Nadia Haugen

Totalt timeantall: 516,5

DATO	ARBEIDSBESKRIVELSE	ARBEIDSOPPGAVE (Fra tids budsjettet)	ARBEIDSSTED	TIMER BRUKT
06.10.2015	Første møte		HSN	2
13.10.2015	Møte internt, hvilken oppgave skal vi søke på.	Dokumentasjon / Søknad	HSN	2
14.10.2015	Skrevet søknad og cv		HJEMME	3
20.10.2015	Møte internt	SØKNAD	HANGOUTS	0
03.11.2015	MØTE INTERNT	SØKNAD	HSN	2
09.11.2015	MØTE INTERNT	OPPGAVE	HSN	2
23.11.2015	Møte nr 1 hos KPS informasjon om oppgaven	MØTE	KPS	3
01.12.2015	Forberedende møte Internt	MØTE	HSN	2
02.12.2015	Møte nr 2 hos KPS intensjon	MØTE	KPS	3
03.12.2015	Research og lesing av ACT + UR5	OPPGAVE	HJEMME	5
04.01.2016	Test hos kps	RESEARCH	KPS	5
05.01.2016	Møte	DOKUMENTASJON	HSN	6
09.01.2016	Alternativt konsept, ulike aktuatorer	Oppgave	HJEMME	5
10.01.2016	Vurdering av rett prosjektmodell		HJEMME	3
14.01.2016	Kommunikasjon, oppklaring, vurd av mod.	MØTE	HSN	3
15.01.2016	Kontrloldokument, Int.avt, revidert utgave	MØTE,	HSN/HJEMME	9
19.01.2016	Sikkerhet/taushetsplikt KPS	Møte intern veileder	KPS/HSN	2
20.01.2016	Stakeholder, krav, oversikt prosj.mod.	Stakeholder, Kravspesifikasjon	HJEMME	5
21.01.2016	Kravspesifikasjon, Spikret prosj.mod.	Kravspesifikasjon	HSN	5,5
21.01.2016	Lest om Scrum, og litt designkonsept.		HJEMME	2
22.01.2016	Gruppemøte	User Stories, Henvendelse/sp.mål KPS	HSN, HJEMME	7,5
24.01.2016	User Stories, lest definisjoner	User Stories	HJEMME	4
26.01.2016	Scrum-møte, tidsestimat		HSN	4
27.01.2016	Samlet modeller, klare i SW, specs på implement. m UR5	Til bruk i senere sprint	HJEMME	3
28.01.2016	SYK			
29.01.2016	Møte hos KPS, referat, status, rettelser		KPS, HSN, HJEMME	8
30.01.2016	Statusrapport	Rapport til KPS	HJEMME	1
31.01.2016	Statusrapport red. mek risk, teknisk utvkonsept	Rapport til KPS, User Stories	HJEMME	6

01.02.2016	Gjennomgang og vurdering; resultater av sysmod. Vurdert opphavsrett på design. Prototype og konvertert for jevn print.	Sprint 1	HJEMME	8
02.02.2016	Scrum-møte, prosjektplan, kommunikasjon	Sprint 1	HSN	5
02.02.2016	Grundig, foreløpig revisjon av Prosjektplan.	Sprint 1	HSN	3,5
03.02.2016	SolidW. laget "simplify" utg av part/mod. Mail til Richard ang. fil ++	[US-051301]	HJEMME	3,5
04.02.2016	Prosjektplan, korrektur, tillegg og interessenter	US - 020210	HSN	1,5
04.02.2016	Møte, oppdatering av tidsestimat, kommunikasjon	Møte	HSN	3,5
05.02.2016	Møte, avklart ang. print, 3D, prosjektplan	[US-051301]	HSN	5
07.02.2016	Ferdigstillt 3D, sendt til print, Bilder og dok til vedlegg	[US-051301]	HJEMME	2
08.02.2016	Omgjøring av modell, prosj.plan,	[US-051301]	HJEMME	6
09.02.2016	Print, møte, dokumentasjon		HSN	4
10.02.2016	Hentet ferdig prototype		HSN	1
11.02.2016	Møte og øving til 1.pres		HSN	5
12.02.2016	Møte og presentasjon	Statusrapport 3	HSN/HJEMME	6,5
16.02.2016	Evalueringsmøte, videre plan	Møte	HSN	3
18.02.2016	Oppdatering av US, tidsestimering for sprint 2	Evaluering og plan	HSN	5,5
18.02.2016	Plan for grensesnitt UR5/EE, US	US-043001	HJEMME	2
19.02.2016	AC, spalting av US, tidsestimat, konseptplanlegging, Disp. for tekn.dok.	Planlegging	HSN	6
19.02.2016	AC, litt tillegg på tek.dok. - ramme	Planlegging av sprint 2, US-021505	HJEMME	2
20.02.2016	SW, teknisk		HJEMME	2
21.02.2016	Forarbeid planlegging videre EE	US-021505	HJEMME	3
22.02.2016	Konseptmøte - mek/el	US-051401	HSN	4
23.02.2016	Lest og gjort notater/kladd for tekdok. Design		HJEMME	6
24.02.2016	SW		HJEMME	4
25.02.2016	Møte, veiledermøte, konsept, statusrapp		HSN	6
25.02.2016	Dok		HJEMME	0,5
26.02.2016	Statusrapport (syk)	US-	HJEMME	2
26.02.2016	SW. "Mate, design med "stag"	US-	HJEMME	4
27.02.2016	Forts SW, revurdert mot enklere konsept, fortsatt 3-grip. Kun som avtalt.	US-	HJEMME	4

27.02.2016	Undersøkt rundt torsjon og bøyespenninger. Konstr.dok	US-	HJEMME	3
28.02.2016	Skrevet ferdig grunnlag for vurderinger i dok	US-	HJEMME	4
01.03.2016	Møte, re-estimering		HSN	4
02.03.2016	Jobbet med samarbeid på design		HSN	4
03.03.2016	Komponentboks, innfestning, noe kladd til dok, opprettet issues		HJEMME	4
04.03.2016	Samkjøring av assemblies, kommunikasjon, sw-konflikter, mail ang print..		HSN	6
05.03.2016	Rekonstr dok strategi...Vurdert modell, Testverifikasjon		HJEMME	4
06.03.2016				
07.03.2016	Produsert mye teknisk dok, skissert endringer/ muligheter på bakgrunn av faglitteratur. Vi går for mod, men noen små endringer må til		HJEMME	6
08.03.2016	Fellesmøte, sprint 2		KPS	3
08.03.2016	Innføring i simulering, planlegging til sprint 3		HJEMME	7
09.03.2016	Dok, issues til spr#2		HJEMME	3
09.03.2016	Sprint review, estim, kommunikasjon	Plan sprint#3	KPS	5
09.03.2016	Konflikt m plan i mod med CG. Var feil m fingre.		HJEMME	3
10.03.2016	Veiledermøte(int)		KPS	3
10.03.2016	Konflikt med simulering	Motion manager/sim i SW	HJEMME	2
11.03.2016	Tekdok EE samarbeid og skrijving, samlet div utklipp fra div kilder	US-021506	KPS	5
11.03.2016	Korrektur og innlegg i status	US-020103	HJEMME	0,5
12.03.2016	Redigering, forsøk på motion analyse i SW	US-080102	HJEMME	6
13.03.2016	Vurdering av løsning på present, pga manglende parametre, kuttet ut for "grundig" tekst til dok	US-021506/080102	HJEMME	5
14.03.2016	CAD;lastet inn UR5, probl. med mate. Oppdaget globale problemer med versjon 2016..(ref: SW.com)		HJEMME	8
15.03.2016	Jobber med å gjøre ferdig dokumentasjonen har problemer med pc		HJEMME	8
16.03.2016	Dokumentasjon		HJEMME	8
17.03.2016	Møte, internt		HSN	6

17.03.2016	Manus, edrawings-forsøk	US-020402, US-020403	HJEMME	5
18.03.2016	2.presentasjon.	US-020404	HJEMME	1,5
18.03.2016		US-020404,020401	HSN	3
07.04.2016	Prioriteringer, user story (SYKDOM)			1,5
08.04.2016				
11.04.2016	Modifisering mot ny motor,samarbeid elektro	Forberedelse til ny prototype US	HSN	6
12.04.2016			HSN	3,5
13.04.2016	Samarbeid 3D, dok		HSN	6,5
14.04.2016	Print, ny sammenst. for vurdering og bilder	US- 010701 + henta print på skolen kveld	HSN	5
15.04.2016	Videre med begr. av mates for rett presentasjon i dokum.	Utvide kabinett, , skrive dokumentasjon	HJEMME	5
16.04.2016	Endring i dokumentasjon		HJEMME	6
18.04.2016	Endringer CAD - for rett rotasjon. Visuell dok.	US-010701, US:02150	HSN	5
18.04.2016	Foreløpig kabinett i plexi	US-010701, US:02150	HJEMME	3
19.04.2016	Videre SolidWorks	US-010701, US:02150	HJEMME	6
20.04.2016	Brakett til Arduino	US-010701, US:02150	HSN	5
21.04.2016	Veiledermøte, gruppemøte internt	Sammarbeid div deler til print	HSN	4
24.04.2016	Borra og fiksa komponentboks, sammenstilling med UR5 i SW for div visualisering		HJEMME	4
25.04.2016	Prototype	US:010701	HSN/HJEMME	5
26.04.2016	Møte KPS og jobbing HSN	US:010701	HSN	5
27.04.2016	Hentet print	US:010701	HSN	0,5
28.04.2016	Møte HSN	US:010701	HSN	0,5
29.04.2016	Testing prototype, statusrapp	US:010701	KPS	5,5
30.04.2016	Modifisering av plassering av fingre	US:010701	HJEMME	3
01.05.2016	Stikkborring, rom til nedre servoer, montering	US:010701	HJEMME	1
02.05.2016	Montering med servoer i boks, testingi	US:010701	KPS	6
03.05.2016	Kabinett til print - med nye mål, kommunikasjon		HSN	4

03.05.2016	Word-problemer, ingen bilder. opprydding		HJEMME	3
04.05.2016	Progresjon, test, dokumentasjon		KPS	6
05.05.2016	Tek.dok, designdok, nye versjoner med innhold, sammendrag, innledning, tabeller og figurer, bilder på nytt. Nye mål.	US:	HJEMME	7
06.05.2016	Monitering ny print, innkjøp "sugru", standardskruer.	US:	KPS	5
06.05.2016	Opprettet UR5 - dok	US:	HJEMME	1
07.05.2016	Dokumentasjon, oversikt og gjennomgang av kilder		HJEMME	5
08.05.2016	Dokumentasjon		HJEMME	5
09.05.2016	Planlegging, re-estimering	Sprint evaluering	HJEMME	4
10.05.2016	Dokumentasjon, standarder inn i SW	US: ,Dokumentasjon	HJEMME	5
11.05.2016	Dok, prøvd meg på program for bilder til User Manual	US:	HJEMME	6
12.05.2016	Dokumentasjon, lite jobbing, ikke i form.	US:021508	HJEMME	4
13.05.2016	Ny sammenstilling med nyeste endringer, hele systemet fra tilkobling til UR5 til fingre. Notert type skruer og størrelser for innsetting i CAD for å kunne generere BOM	US:021508	HJEMME	6
14.05.2016	Probl. med toolbox, foreløpig nedprioritert fremfor dokumenter. Litt teknisk.	US:021508 og bistand til brukerman	HJEMME	6
15.05.2016	Teknisk dokumentasjon, noe redigering, samt diskusjoner/begr for konseptvalg	US:021508	HJEMME	4,5
16.05.2016	Skrevet litt i fremtidsdokument, setter opp disposisjon for EE-dok så det er klart, jobber videre med teknisk spesifisering.			7,5
17.05.2016	Lagt inn partnr fra manualen i SW egen konfig for bilde av hele syst. Sletta, redigert og fylt på litt tekst.	US:	HJEMME	4
18.05.2016	Dokumentasjon, Klargjøring for levering		HSN	6
19.05.2016	Sammenfletting, dokumenter, korrektur..		HSN	8
20.05.2016	Innlevering av dokumentasjon		HSN	4

7. Timeliste Espen Svenne Kraglund

Totalt timeantall: 477 timer

DATO	ARBEIDSBESKRIVELSE	ARBEIDSOPPGAVE (Fra tids budsjettet)	ARBEIDSSTED	TIMER BRUKT
06.10.2015	Første møte		HBV	2
13.10.2015	Møte internt, hvilken oppgave skal vi søke på.		HBV	2
14.10.2015	Skrevet søknad og cv	Dokumentasjon / Søknad	HJEMME	3
20.10.2015	Møte internt	SØKNAD	HANGOUTS	2
03.11.2015	MØTE INTERNT	SØKNAD	HBV	2
09.11.2015	MØTE INTERNT	OPPGAVE	HBV	2
23.11.2015	Møte nr 1 hos KPS informasjon om oppgaven	MØTE	KPS	3
01.12.2015	Forberedende møte Internt	MØTE	HBV	2
02.12.2015	Møte nr 2 hos KPS intensjon	MØTE	KPS	3
04.01.2016	Test hos kps	RESEARCH	KPS	5
05.01.2016	Møte	DOKUMENTASJON	HSN	6
06.01.2016	Research/Usecase/Budsjett	RESEARCH	Hjemme	2,5
09.01.2016	Forprosjekt dokument	DOKUMENTASJON	Hjemme	2,5
11.01.2016	Webseite		Hjemme	2,5
12.01.2016	Webseite		Hjemme	2,5
13.01.2016	Møte	Møte	Drammen	3
18.01.2016	Webseite		Hjemme	1,5
19.01.2016	MØTE	KPS, SIKKERHETSBRIEF	KPS	2
19.01.2016	MØTE	VEILEDERMØTE	HSN	2
20.01.2016	Research/webseite	Webseite	Hjemme	2
21.01.2016	MØTE		HSN	6
22.01.2016	MØTE	User stories	HSN	6
25.01.2016	Research		Hjemme	1
26.01.2016	Webseite		Hjemme	2
27.01.2016	Prosjektplan	Prosjektmodell	Hjemme	3
28.01.2016	MØTE	User stories	HSN	3,5
28.01.2016	Hjemmerarbeid	Scrum/prosjekt plan	Hjemme	1,5
29.01.2016	MØTE KPS	Møte	KPS	4
	Planning Sprint1	SPRINT PLANNING	HSN	2
31.01.2016	Research		Hjemme	1
	Risk			1
	Oppfølgingsdokument			0,5
01.02.2016	Software Risk		Hjemme	1
	Research			2

02.02.2016	Møte	Møte	HSN	1
	Research software	US-021702	KPS	7
03.02.2016		Prosjektplan	Hjemme	0,5
		User stories		0,5
04.02.2016		US-020252	HSN	1
		Prosjektplan		2,5
		Webside		0,5
		Fremdriftsmøte		1
		Scrummøte etc		1
05.02.2016		US-020251	HSN	2
		US-020210		3
		Møte		1,5
08.02.2016		US-020210	HSN	2
		US-070601		1
		Møte		1
09.02.2016		Korrektur prosjektplan	HSN	2
		Møte		2
		Webside		2
10.02.2016		Korrektur dokumentasjon	HSN	2
		Planlegging presentasjon		3
		Forberedelse presentasjon	Hjemme	2
11.02.2016		Forberedelse presentasjon	HSN	6
		Webside		1
12.02.2016		Forberedelse presentasjon	HSN	3
		Webside		1
		Presentasjon		1
15.02.2016		Webside	Hjemme	1
16.02.2016	Sprint evaluering		HSN	3
18.02.2016	Sprint 2 plan		HSN	6
19.02.2016	Sprint 2 plan		HSN	3
	Blog	US-070603		3
22.02.2016	Blog	US-070603		1
	Software designdokument	US-021706		1
	Use case	US-021706		1
23.02.2016	Blog	US-070603		4
24.02.2016	Use case	US-021706		1
	Software designdokument			1
25.02.2016				2,5
		US-040401	HSN	4

26.02.2016	use case	US-021702	HSN	4
		møter og kommunikasjon	HSN	2,5
27.02.2016	Use case	US-021702	HSN	2
		Reestimering tid	HSN	1
		Sprint møte	HSN	
01.03.2016		Sprint møte	HSN	
	Use case diagram	US-021702	HSN	3
03.01.2016		US-040401	Næringsparken	6
04.01.2016	Budsjett	US-070401	Hjemme	3
		US-021706		2
05.01.2016		komm.		1,5
08.03.2016		Sprint review	Næringsparken	4
	Økonomi	US-070401		1
		Research/komm		2
		User stories		2
09.03.2016		Sprint planning/review	Næringsparken	3
	Budsjett	US-070401		1
	Blog	US-070607		1
	Sftwr. design	US-021705		1
10.03.2016		Statusrapport	Næringsparken	4
		Komm.		1
		Veiledermøte		1
		Risk		1
		Møte		1
11.03.2016		Komm.	Hjemme	0,5
		Sprint retrospektiv		1
		Status rapport		2,5
12.03.2016		Burndown	HSN	5
	UML	US-021701		0,5
13.03.2016		korrektur	hjemme	1
14.03.2016	Software designdokument	US-021705	HSN	6
15.03.2016		Møte	KPS	1
	Software designdokument	US-021705		7
	UML	US-021701	HSN	3
16.03.2016		Innlevering av dokumentasjon	HSN	7,5
17.03.2016	Møter		HSN	6
	manus	US-020403		1
18.03.2016	Øving presentasjon,	US-020404	HSN	6

	holde presentasjon	US-020401		1
	møter og diverse			1,5
06.04.2016		Sprint 3 review	Hjemme	1
07.04.2016		Sprint 3 review	HSN	8
08.04.2016		sprint 4 planning	HSN	5
11.04.2016		[US-040701]progg xml del	HSN	6,5
12.04.2016		[US-040701]progg xml del	HSN	6
13.04.2016		[US-045101]progg connection	HSN	6
14.04.2016		[US-045101]progg ++	HSN	6,5
17.04.2016		Samarbeid om GIT	HSN	1
18.04.2016		[US-040701]Progg XML del	HSN	6,5
19.04.2016		[US-045101]Programmering kom. TCP/IP	HSN	5
20.04.2016		[US-045101]Programmering kom. TCP/IP	HSN	7
21.04.2016		lese om programmering. møter	HSN	5
22.04.2016		[US-045201]Programmere ur5	KPS	6,5
25.04.2016		[US-060201]trådkommunikasjon og gui	KPS	6,5
26.04.2016		[US-060201]	KPS	6
27.04.2016		Input statusrapport og diverse	Hjemme	1
27.04.2016		Teambuilding		8
28.04.2016		[US-045201] og GUI improvements	KPS	6
29.04.2016		[US-045201] og Bugfixing i software	KPS	4,5
29.04.2016		Oppfølgingsdokument, timer, etc.	Hjemme	1
02.05.2016		[US-042003] kode	KPS	4
02.05.2016		[US-021002]Testverifikasjon og risiko	KPS	2
02.05.2016		Blogpost	KPS	0,5
02.05.2016		[US-021707]Dokumentasjon	KPS	2,5
02.05.2016		kommunikasjon	Hjemme	0,5
03.05.2016		[US-021707]Dokumentasjon	KPS	5
04.05.2016		[US-040301]	KPS	2
		[US-021707]Dokumentasjon		4,5
04.05.2016		Kommunikasjon	Hjemme	0,5
05.05.2016		[US-040301]teste komm. arduino /Ur 5	KPS	4

05.05.2016		[US-021707]Dokumentasjon		2,5
06.05.2016		Kommunikasjon	Hjemme	0,5
09.05.2016		Kommunikasjon og diverse	Hjemme	1
10.05.2016		Diverse admin	Hjemme	3
10.05.2016		[US-021003]Testverifikasjon	Hjemme	3
10.05.2016		Korreksjon + problemer med g disk	Hjemme	2,5
10.05.2016		Kommunikasjon	Hjemme	0,5
10.05.2016		[US-021707]Tek dok	Hjemme	1
11.05.2016	skrivning	[US-021707]	Hjemme	6
12.05.2016	skrivning	[US-021707]	KPS	3
12.05.2016		Veiledermøte++	HSN	2
12.05.2016		Kommunikasjon	Hjemme	1
13.05.2016		[US-021707]Tek dok	Hjemme	6,5
14.05.2016		[US-021803]Brukermanual	Hjemme	2
15.05.2016		komm	Hjemme	1
15.05.2016		[US-021803]brukermanual	Hjemme	1,5
15.05.2016		[US-021707]tek dok	Hjemme	1
16.05.2016		[US-021707]Software tek. dok.	KPS	3
		Korrektur		2
		[US-021803]Brukermanual		2
17.05.2016		[US-021707]Software tek. dok.	HSN	9
18.05.2016				9
19.05.2016	dokumentasjon og korrektur		HSN	9
20.05.2016	ferdigstilling og innlevering dokumentasjon		HSN	9

8. Konklusjon

Vi har hatt fokus på at arbeidet skal fordels jevnt på alle medlemmene av gruppen. Det syntes jeg vi har fått til bra. Alle har samarbeidet, så det har blitt en balansert fordeling mellom skrivning og praktisk arbeid.

Fremtidsdokument

ACTEE

AUTOMATED CROWS TESTING END EFFECTOR

PROSJEKT	Automated CROWS testing end effector
OPPDRAAGSGIVER	Kongsberg Protech Systems
UTFØRT VED	Høgskolen i Sørøst-Norge
DOKUMENTNAVN	Fremtidsdokument
VERSJON / DATO	1.0 / 21. mai 2016
FORFATTER	Bård Simen Hamborg Enget
SIDEANTALL	11



KONGSBERG

AINA ENGEN NILSEN
BÅRD SIMEN HAMBORG ENGET
CARL MARTIN MAUSETH
ESPEN SVENNE KRAGLUND
MONICA NADIA HAUGEN
THOMAS WEGENER



i. Abstrakt

Dette dokumentet beskriver de ideene som prosjektgruppen ACTEE hadde for forbedringer på den valgte løsningen. Grunnet restriksjoner på tid og kompleksitet var det flere ideer vi hadde som ikke ble gjennomført. Når produktet ble ferdig er det også enklere å se hvor vi har forbedringspotensial. Dokumentet vil inneholde ting vi mener kan forbedre systemet, men også ting vi mener burde vurderes å gjøre annerledes enn dagens løsning.

ii. Innholdsfortegnelse

i. Abstrakt.....	1
ii. Innholdsfortegnelse	2
iii. Oversikt tabeller.....	2
iv. Oversikt figurer	3
v. Dokumenthistorie	3
vi. Forkortelser	3
1 Innledning	4
2 Software.....	5
2.1 Weapon_ID valg ved oppstart av Software	5
2.2 Legge alle bevegelser til som funksjoner	5
2.3 Forbedre kildekoden.....	5
3 Elektro.....	6
3.1 WSS	6
3.2 UR5-Arduino kommunikasjon	6
3.3 Kode bevegelse av flere servoer samtidig	7
3.4 Effektforsyning til End Effector.....	7
4 Maskin	8
4.1 Foredling og ytterligere tilpasning av fingrene og grepet.....	8
4.2 Design innkapsling av servomotorer	9
4.3 Utvidet funksjonalitet	9
4.4 Materialvalg	9
5 Referanser	10

iii. Oversikt tabeller

Tabell 1: Dokumenthistorie.....	3
Tabell 2: Forkortelser	3

iv. Oversikt figurer

Figur 1: Eksempel på forbedring av fingerdesign 1	8
Figur 2: Eksempel på forbedring av fingerdesign 2	8

v. Dokumenthistorie

Tabell 1: Dokumenthistorie

Versjon 0.1	19.05.16	Disposisjon av Bård Simen Lagt til kapittel 3.1, 3.2 og 3.3 av Bård Simen Lagt til kapittel 3.4 og 4.2 av Carl Martin Lagt til kapittel 4.1 av Aina Lagt til kapittel 2.3 av Thomas
Versjon 0.2	20.05.16	Lag til kapittel 4.3 og 4.3 av Monica Omskrevet Abstrakt og Innledning av Bård Simen
Versjon 1.0	21.05.16	Espen: Forklart grundigere hvor nødvendig, Korrektur.

vi. Forkortelser

Tabell 2: Forkortelser

Forkortelser	Forklaring
ACTEE	Automated CROWS Testing End Effector
EE	End Effector, gripeverktøy
CG	Control Grip, styringsjoystick for våpenstasjon
CROWS	Common Remotely Operated Weapon Station
UR5	Universal Robots 5 robotarm

1 Innledning

I et prosjekt av dette omfanget er det flere restriksjoner som må tas hensyn til. På bakgrunn av disse vil det være kompromisser som må inngås for å komme frem til et produkt som er tilfredsstillende innenfor de rammene som er gitt.

Det er i den forbindelse vi har laget dette dokumentet for å sette fokus på de områdene gruppen selv mener burde utbedres, eller eventuelt endres. Vi har delt dokumentet inn i de respektive fagområdenes fokusfelt for at oppdragsgiver lettere skal kunne identifisere hvilken kompetanse vi ser for oss at er nødvendig for å gjøre disse utbedringene.

2 Software

Her vil vi presentere forslagene til forbedringer/endringer som kan gjøres med Software biten av vårt prosjekt.

2.1 Weapon_ID valg ved oppstart av Software

Om det er nødvendig å skifte våpenet på våpenstasjonen vill det vært hensiktsmessig å kunne skifte Weapon_ID direkte fra testprogrammet istedenfor å måtte modifisere koden på Arduino. Weapon_ID er nødvendig for å opprette kommunikasjon mellom WSS og våpentårn. Dette kan implementeres i programmet ved at det opprettes en meny hvor brukeren kan velge mellom mulige Weapon_ID og at denne verdien så sendes som en streng til sensoren.

2.2 Legge alle bevegelser til som funksjoner

For å kunne verifisere at de handlingene vi ber systemet vårt om å gjøre, faktisk blir gjort, er det nødvendig med verifikasjon. En slik verifikasjon måtte omhandlet kommunikasjon med våpenstasjonen som vi på grunn av tidsmangel valgte å nedprioritere. Dette ville bli gjort ved at hver bevegelse ville fått en egen funksjon i koden i testprogrammvaren. Der kunne vi utført de handlingene som ville vært nødvendig for å verifisere om de handlingene systemet skulle gjøre hadde blitt utført. Ett eksempel på dette ville vært, om vi skrudde på våpenstasjonen kunne vi fått en beskjed tilbake fra våpenstasjonen at den var på.

2.3 Forbedre kildekoden.

Hvis vi hadde hatt mer tid ville vi skrevet programmet i en gjennomgående stil. Og byttet ut flere av de stedene der vi har brukt tråder for å hente informasjon om variabler, med «iNotifyOnPropertyChanged classer» og «eventlisteners», det er flere steder i koden vi kunne dratt nytte av get og set funksjonaliteten i c#, vi har skrevet noe som fungerer og ikke hatt til å prioritere å forbedre det.

3 Elektro

Her vil vi presentere forslagene til forbedringer/endringer som kan gjøres med elektrodelen av vårt prosjekt.

3.1 WSS

Gruppen ser flere muligheter for forbedringer på funksjonaliteten til denne komponenten, spesielt med i forhold til fyringsdetektering, men også med tanke på utvidet funksjonalitet.

Fyrings deteksjon kan forbedres om man kan få en bedre oversikt over fyringssignalene. Dette kan gjøres ved å koble et scoop med minne funksjon til D3 pinnen på Arduinoen som er koblet til WS. Arduino koden kan med noe modifisering kunne tilpasses til å kun reagere om signalet er høyt i en viss periode. Om funnene viser at fyringssignalet holdes høy i f.eks. en periode på 20ms, kan det fokuseres inn på denne i koden og transienter vil ikke lenger bli et problem. Da vil man i teorien også kunne telle skuddene i burst fire og med dette kunne få en mer nøyaktig validering av også denne funksjonen på WS.

Siden noe av det som gir WSS mest verdi er muligheten til å verifisere handlinger uten input direkte fra WS, men heller se på signaler som blir gitt ut til eksterne enheter, hadde vi også i tankene å legge til en bevegelsessensor på WS. Det ble i den forbindelse kjøpt inn et magnetometer som med forholdsvis liten innsats kan settes opp sammen med mikrokontrolleren til å detektere endringer i posisjon på WS, og på denne måten overvåke om dette skjer når det er planlagt som en del av en test, men også om det skjer når det ikke er planlagt på grunn av en feil i Software. Da tiden dessverre ikke strakk til, ble denne muligheten skrinlagt, men vi ser at den kan ha stor nytte verdi og er verdt å investere litt tid i senere.

3.2 UR5-Arduino kommunikasjon

UR5 kommunikasjonen er langt fra perfekt slik den er i dag. Vi er kvitt alle spor av støy, men løsningen er ikke fullgod. Under utbedring av dagens løsning så vi at UR5 har støtte for bruk av Modbus[1]. Kilden var litt uklar i om støtten også inkluderte UR5 tool I/O portene, eller om det kun gjaldt de øvrige digitale I/O portene på systemet. Siden vi ikke hadde tid til videre utforskning på daværende tidspunktet nevnes muligheten her. Dette er absolutt en mulighet som burde utforskes videre. Arduino mikrokontrolleren støtter også bruk av Modbus via tillegget ModbusMaster Library[2] som gjør at å utforske muligheten har enda større verdi.

En liten forbedring på eksisterende løsning ville være å legge til en tilbakemelding via UR5 sine Digitale input porter på tool I/O connectoren der UR5 koden kan lytte etter tilbakemelding på at EE har gjort sine bevegelser. Dette vil gjøre systemet smidigere og raskere, da man ikke må legge inn en estimert Wait funksjon med litt ekstra buffer etter hver bevegelse gjennomført av EE.

3.3 Kode bevegelse av flere servoer samtidig

For at koden skal kunne kjøre raskere enn den gjør i dag, uten å måtte sette opp hastigheten på bevegelsen på servoene kunne det vært nyttig å lage en funksjon som kan håndtere bevegelse på to eller tre servoer samtidig. Det er lite sannsynlig at man skulle trenge at flere enn tre servoer kjører samtidig, så å tilrettelegge for flere enn dette anser vi som unødvendig.

3.4 Effektforsyning til End Effector

Det kunne vært ønskelig med et egen strømforsyningsadapter til EE slik at ved senere bruk av systemet kan man kunne slippe å måtte benytte seg av dagens løsning med lab strømforsyning og kabelstrekk slik som det er blitt brukt på dagens prototype.

4 Maskin

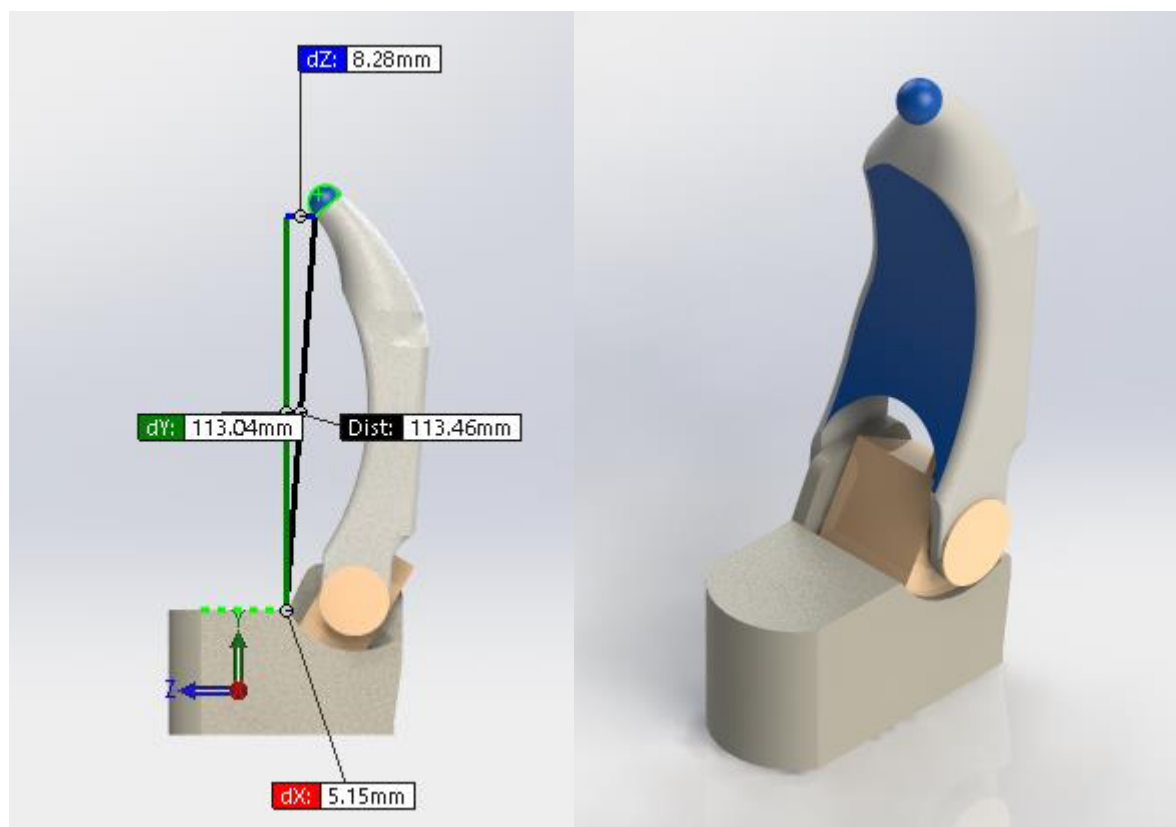
Her vil vi presentere forslagene til forbedringer og endringer som kan gjøres med den mekaniske delen av vårt prosjekt.

4.1 Foredling og ytterligere tilpasning av fingrene og grepet.

Ut ifra de tester vi har gjennomført, ser vi at det vil være fordelaktig med ytterligere tilpasninger av fingrene for bedre passform i grepet rundt CG. Fingertupp som skal betjene trigger – knapp kan med fordel utformes bedre til knappen den skal betjene.

Buede mellomledd og en tanke kortere grep vil være optimalt, anordning i håndflaten bør tas frem på nytt med litt større fokus.

Et tegnet forslag for videreutvikling;



Figur 2: Eksempel på forbedring av fingerdesign 2 Figur 1: Eksempel på forbedring av fingerdesign 1

I utgangspunktet skulle EE betjene samtlige knapper på CG. Målet ble revurdert underveis, og redusert til betjening av palm, trigger og bevegelse av CG grunnet begrenset tid til rådighet.

Betjening av øvrige knapper på CG vil derfor være et klart mål for fremtiden. Den buede tommelen (figur 1 og 2) er utformet med tanke på dette, med kun ett ledd i base og smal fingertupp.

Nok et fremtidsmål for fingrene er enten implementert verktøy i fingertupp, eller enkel utskiftbar fingertupp med verktøy. Ønsket er at EE skal kunne skifte ut sikringer fra en «break out – box», da vil en løsning med tilpasset verktøy for dette være fordelaktig.

Sensorer er planlagt inn i blå felt. Ønsket er både en kraftsensor, økt friksjon samt noe formbar anordning i ett. Sensorene skal gi tilbakemelding om trykk og posisjon.

4.2 Design innkapsling av servomotorer

Ved designet til EE har vi et ønske om et slakere finesse som gjør at fingre skal være mest mulig i størrelse av servomotorene, ved ett sterkere materiale vil dette kunne oppfylles i en senere utvikling av prototype til EE.

4.3 Utvidet funksjonalitet

Det har blitt skrenket inn en del på ønskelige oppgaver EE skal kunne utføre. Vårt gripeverktøy vil sannsynligvis ikke klare å trekke ut sikring fra break-out-box, så det kan løses ved å ha utbyttbare flate fingertupper. For et slikt verktøy er vekt viktig, massesenter bør ligge lavest mulig, dvs. i håndflata. Generelt bør det etterstrebtes lavest mulig tyngde. Og de geometriske forhold, størrelse på hvert ledd i forhold til hverandre samt riktig størrelse på fingre i forhold til håndflaten. En endring kan være å ha en mer buet fingertupp, og bruke motorer kun i bunnen. Dette kan gi like mye kraft til fingertuppene.

4.4 Materialvalg

Materialet: I utgangspunktet vil en høyfast aluminiums type, f.eks. Al6160 være et egnet materiale for produksjon, men en titanlegering vil inneha mer elastisitet og evne til å ta opp vibrasjoner. Generelt vil det også være energibesparende å fordele kraft på minst mulige flater og lengder. Vi tror fortsatt en tre-finger hånd er bra, og med muligheter for å endre mellom flere konfigurasjoner både med tanke på fingerlengder, antall fingre og innfesting i håndflate, samt utbyttbare fingertupper ville dette resultere i et fleksibelt og bra konsept. Det ville da være like ledd som man kunne ta fra hverandre, samt mulighet for justering av avstand mellom innfesting i håndflate.

5 Referanser

[1] Spehro Pefhany, (Jan 2000), *Modbus Protocol*, Tilgjengelig:
http://www.interlog.com/~speff/usefulinfo/modbus_protocol.pdf
Sist besøkt: 19.05.2016

[2] Doc Walker, (ukjent), *ModbusMaster library for Arduino*, [Nettside]. Tilgjengelig:
<http://playground.arduino.cc/Code/ModbusMaster>
Sist besøkt: 19.05.2016

Konklusjon Hovedrapport

I dette prosjektet har vi utviklet et system som skal hjelpe bedriften med å komme nærmere målet med automatisert utførelse av tester på CROWS M153 våpenstasjoner til den hensikt å effektivisere arbeidet med testing og dermed spare tid og penger.

For å utvikle det automatiske systemet har vi benyttet oss av prosjekt modellen Scrum som bryter prosjektets mål ned i user stories som må gjennomføres for å oppnå ønsket funksjonalitet. Da bredden i prosjektet i utgangspunktet var ganske stor har vi ikke klart å gjennomføre alle de planlagte user storiene, men vi har prioritert å gjøre de som ansees som viktigst for å oppnå god funksjonalitet.

Prosjektets resultat er en prototype av et gripeverktøy som oppfyller en del av ønskene oppdragsgiver hadde i starten av prosjektet, og som med litt mer tilpassing kan oppfylle flere av ønskene. Verktøyet kan i dag betjene DCP, og de to viktigste funksjonene på CG, palm og trigger knappene.

Systemet er styrt av en egen testprogramvare som kontrollerer UR5 roboten og kan via UR5 sine digitale tool I/O utganger kontrollere EE sine bevegelser. Systemet har lagt til rette for at brukerne enkelt kan generere sine egne testsekvenser satt sammen av ferdig programmerte sekvenser på UR5.

Vi har også utviklet et loggsystem som rapporterer UR5 sine bevegelser og mottar informasjon fra en egenutviklet sensor som er koblet mot våpenstasjonen og gir tilbakemeldinger til systemet om fyringsinformasjon uavhengig av våpenstasjonen sin egen programvare. Denne kan også utvikles videre med mer og bedre funksjonalitet. Bedre støtte for avfyringstelling og bevegelsesdeteksjon er eksempler på dette.

Vi har samlet en del tanker om veien videre for prosjektet i et eget fremtidsdokument. Der står det litt mer utfyllende om svakheter vi ser med systemet og forslag til hvordan disse kan utbedres.