



Høgskulen på Vestlandet

Bacheloroppgave

ELE350

Predefinert informasjon

Startdato:	08-05-2023 09:00 CEST	Termin:	2023 VÅR
Sluttdato:	22-05-2023 14:00 CEST	Vurderingsform:	Norsk 6-trinns skala (A-F)
Eksamensform:	Bacheloroppgave		
Flowkode:	203 ELE350 1 O 2023 VÅR		
Intern sensor:	Tom Kjøde		

Deltaker

Naun:	Sindre Eikeland
Kandidatnr.:	223
HVL-id:	591472@hvl.no

Informasjon fra deltaker

Egenerklæring *: Ja
**Inneholder besvarelsen
konfidensielt
materiale?:** Ja
**Jeg bekrefter at jeg har
registrert
oppgavetittelen på
norsk og engelsk i
StudentWeb og vet at
denne vil stå på
vitnemålet mitt *:**

Gruppe

Gruppenavn: Enmannsgruppe
Gruppenummer: 4
**Andre medlemmer i
gruppen:** Benjamin Våge Nilsen, Ådne Måkestad

Jeg godkjenner avtalen om publisering av bacheloroppgaven min *

Ja

Er bacheloroppgaven skrevet som del av et større forskningsprosjekt ved HVL? *

Nei

BACHELOROPPGAVE:

BO23EB-34

Oppsett og testing av Mitsubishi Cobot

Sindre Eikeland, K.nr: 223
Benjamin Våge Nilsen, K.nr: 273
Ådne Måkestad, K.nr: 327

21. mai. 2023

Dokumentkontroll

<i>Rapportens tittel:</i> BO23EB-34	<i>Dato/Versjon</i> 21. mai. 2023/1.10
	<i>Rapportnummer:</i> BO23EB-34
<i>Forfatter(e):</i> Sindre Eikeland Benjamin Våge Nilsen Ådne Måkestad	<i>Studieretning:</i> AUT20
	<i>Antall sider m/vedlegg</i> 81
<i>Høgskolens veileder:</i> Tom Kjøde	<i>Gradering:</i> Åpen
<i>Eventuelle Merknader:</i> Vi tillater at oppgaven kan publiseres.	

<i>Oppdragsgiver:</i> Fagskolen Vestland	<i>Oppdragsgivers referanse:</i> Guttorm Lyngvær / Bjørnar Seim
<i>Oppdragsgivers kontaktperson(er) (inkludert kontaktinformasjon):</i> Bjørnar Seim (bjornar.seim@vlfk.no) og Guttorm Lyngvær (guttorm.lyngver@vlfk.no)	

Revisjon	Dato	Status	Utført av
0.11	08.03.23	Forprosjekt lagt inn	BVN
0.12	11.04.23	Første utkast	Gruppen
0.20	04.05.23	Andre utkast	Gruppen
1.00	12.05.23	Ferdig	Gruppen
1.10	19.05.23	Revidert (Korreksjon)	BVN

Forord

Denne rapporten dokumenterer vår avsluttende bacheloroppgave på studiet «Automatisering med robotikk» ved Høgskulen på Vestlandet, avdeling Bergen. Prosjektet inneholder hva vi har lært om roboten og utforming av lab-oppgaver med løsningsforslag, tilkoblingsguide og valg knyttet til bygging av moduler og simuleringsmuligheter.

Det har vært et lærerikt prosjekt og vi takker for oppgaven vi fikk av Fagskolen Vestland. Vi var tidlig i kontakt med oppdragsgiver og fikk satt i gang et godt samarbeid. I løpet av prosjektet har vi hatt flere utfordringer som har dukket opp, dette har gitt oss muligheten til å bruke kunnskapen vi har fått i fra tidligere fag gjennom studiet. Vi har i tillegg tilegnet oss ny kunnskap underveis for å løse problemer som vi har hatt knyttet mot oppgaven. Vi har lært mye om planlegging av et langvarig prosjekt og arbeid mot krav og forventninger fra en oppdragsgiver.

Vi vil gjerne takke vår veileder Tom Kjøde, for reflekterende møter og gode tips og råd underveis i prosjektarbeidet. I tillegg ønsker vi å takke Bjørn Seim og Guttorm Lyngver ved fagskolen, som har tilrettelagt for gjennomføring av arbeid, samt innkjøp av utstyr. Til slutt vil vi takke studentene ved fagskolen som testen lab-oppgavene våre og ga gode tilbakemeldinger.

Sammendrag

Oppdragsgiver for denne bachelor oppgaven har vært Fagskolen Vestland. Fagskolen har nylig anskaffet seg en samarbeidende robot fra Mitsubishi Electric, denne skal brukes i forbindelse med undervisning ved fagskolen. Hensikten med oppgaven var at studenter ved bachelor programmet *Automatisering med Robotikk* ved Høgskulen på Vestlandet, campus Bergen skulle gjøre seg kjent med roboten og dens funksjoner. Dette skulle hjelpe lærere ved fagskolen med å komme i gang med roboten, og samtidig skulle det utarbeides et forslag til et undervisningsopplegg som kunne brukes ved fagskolen i fremtiden.

Gjennom dette arbeidet har det blitt utviklet et laboratorieoppgavesett med tre oppgaver, samt oppgaveveiledning for underviser, undervisningsvideoer og guide til oppkobling mot roboten. Sammen med oppgavene er det konstruert en modul som inneholder elementer som skal løses i oppgavene, det er også laget forslag til enda en modul for siste oppgave. Modulene henges på roboten som gjør det mulig å flytte på roboten uten å miste funksjonene til programmet man lager.

Denne rapporten og arbeidet rundt den svarer godt til kravspesifikasjonen fra oppdragsgiver, dette er også tilbakemeldinger vi har fått fra oppdragsgiver og at de ser nytte i vår leveranse. Det er noen begrensninger ved løsningen da den siste oppgaven i oppgavesettet ikke ble realisert, dette skylder problemer rundt anskaffelse av en vital del av oppgaven. Dette har blitt avklart med oppdragsgiver og løst med å levere design og materiale for å lage siste modul til siste oppgave.

Abstract

The client for this bachelor's thesis has been Fagskulen Vestland. The vocational school had recently acquired a collaborative robot from Mitsubishi Electric, which is intended to be used for teaching at the school. The purpose of the thesis was for students in the Automation with Robotics bachelor's program at Western Norway University of Applied Sciences, campus Bergen to become familiar with the robot and its functions. This was intended to help teachers at the vocational school get started with the robot. In addition to the thesis, a proposal for a teaching plan that could be used for the school's students in the future was to be created.

Through this work, a laboratory task set consisting of three tasks was developed, as well as task guidance for instructors, teaching videos, and a guide for connecting to the robot. Along with the tasks, there are modules that contain elements that must be solved in the tasks. These are modules that are attached to the robot, making it possible to move the robot without losing the functions of the program being created.

This report and the work around it meet the client's requirements, and we have received feedback from the client that they see the value in our delivery. There are some limitations to the solution, as the final task in the task set was not realized due to problems related to the acquisition of a vital part of the task. This has been clarified with the client and resolved by providing design and material for creating the final module for the final task.

1 Innhold

Dokumentkontroll	2
Forord	3
Sammendrag	4
1 Forklaring og definisjon av uttrykk	8
2 Innledning	9
2.1 Organisering av rapport	9
2.2 Oppdragsgiver	9
2.3 Problemstilling	9
2.4 Cobot	10
2.5 Mitsubishi - MELFA ASSISTA	11
3 Kravspesifikasjon	12
3.1 Tilkoblingsguide	12
3.2 Moduler	12
3.3 Simulering	13
4 Analyse av problemet	14
4.1 Utforming av mulig løsning	14
4.1.1 Løsningsalternativ 1	14
4.1.2 Løsningsalternativ 2	15
4.2 Programvarer	16
4.3 Maskinvare	18
4.4 Konklusjon fra forprosjekt	18
5 Realisering av valgt løsning	19
5.1 Oppgavesett	19
5.1.1 Laboratorieøving 1	20
5.1.2 Laboratorieøving 2	21
5.1.3 Laboratorieøving 3	22
5.2 Leveranse	23
5.2.1 Modul	23
5.2.2 Klargjort programfil	24
5.2.3 Videoer	24
5.2.4 Oppstarts manual	25
5.2.5 Veiledninger til underviser	26
5.2.6 Design	26
6 Testing	27

6.1	Test av fysisk utstyr.....	27
6.2	Test av virtuell modell.....	28
6.3	Test gjennomført mot studenter.....	28
7	Diskusjon.....	29
7.1	Vurdere valgt løsning	29
7.2	Problem undervegs	30
7.3	Videre arbeid og utvidelser	30
7.4	Pris.....	31
8	Konklusjon	32
	Referanser.....	33
	Figurliste.....	34
	Appendiks A Prosjektledelse og styring.....	35
	A.1 Prosjektorganisasjon.....	35
	A.2 Fremdriftsplan	36
	A.3 Risikoliste.....	37
	Appendiks B Tilkoblingsguide.....	38
	B.1 Tilkoblingsguide RT VisualBox.....	38
	B.2 Tilkoblingsguide RT Toolbox3	44
	Appendiks C Tegninger.....	53
	Appendiks D Oppgavesett og veiledning til underviser.....	60
	D.1 Oppgavesett	60
	D.2 Veiledning til underviser	72

1 Forklaring og definisjon av uttrykk

I dette kapitelet vil ord og uttrykk som er nyttig å forstå når man leser gjennom rapporten bli forklart.

Cobot: Forkortelse for collaborative robot (Samarbeidende robot), som ved hjelp av sikkerhetsteknologi gjør det mulig for mennesker å samarbeide med roboten. Coboter kommer i forskjellig fasonger og med forskjellig antall ledd. Mer utfyllende om cobot se [avsnitt 2.4](#).

MELFA ASISSTA: MELFA ASISSTA er cobotserien til Mitsubishi Electric. Mer utfyllende om dette se [avsnitt 2.5](#).

Digital-tvilling: En digital tvilling er ofte definert som en digital eller virtuell kopi av en fysisk ting som finnes i den virkelige verdenen [1]. I vårt tilfelle har vi en digital kopi av roboten MELFA ASISSTA i programmeringsprogrammene til Mitsubishi.

RT Visualbox og RT Toolbox3: Dette er Mitsubishi sine egne programvarer, som vi har brukt for å programmere coboten. En kan lese mer om programmeringsprogramma i [avsnitt 4.2](#).

Syntax feil: Syntax feil er et begrep som brukes under skriving av koding til programmer. Når dette oppstår betyr det at det er en feil i den måten koden er skrevet, som gjør at den bryter med reglene i det programmeringsspråket som brukes.

IO: IO er forkortelsen for «Input/Output». IO er kommunikasjon mellom et informasjonsbehandlingssystem, for eksempel en datamaskin. I vår sammenheng er det kommunikasjonen mellom roboten og datamaskinen. Innganger er signaler eller data som mottas av systemet, og utganger er signaler eller data som sendes fra systemet.

DIP switch: Er en manuell bryter som kan stå i av eller på posisjon. De blir brukt til å kontrollere oppførselen til elektroniske enheter.

2 Innledning

2.1 Organisering av rapport

En god del av innholdet i denne rapporten består av appendikser, ettersom hovedmålet med denne oppgaven har vært bygget på å lage undervisningsopplegg for fagskolen. Rapporten beskriver vårt arbeid med de forskjellige delene av leveransen vår til fagskolen, dette er blant annet; oppgavesett, undervisningsvideoer, veiledning og konstruksjon for gjennomføring av oppgaver.

2.2 Oppdragsgiver

Oppdragsgiver for denne oppgaven er Fagskolen Vestland (avdeling Nordnes, Bergen). Fagskolen er en av de største fagskolene i landet, og har studietilbud innen tekniske fag, maritime og marine fag, helsefag, petroleumsfag, jordbruksfag, arboristfag og lokal matkultur. Flere av studietilbudene er også tilrettelagt for å kunne kombineres med jobb. Fagskolen Vestland har rundt 1900 studenter og har åtte studiesteder i Vestland fylke [2]. Arbeidet vi har utført i denne oppgaven har vært rettet mot studieretningene *Robotteknologi og digital produksjon i industrien* samt *Automatisering* ved fagskolen.

2.3 Problemstilling

Fagskolen Vestland har nettopp anskaffet en samarbeidende robot av typen MELFA ASSISTA fra Mitsubishi Electric. Roboten skal brukes til lab-øvelser og undervisning innenfor robotikk. En samarbeidende robot, også kalt cobot, er en robotarm som ved hjelp av avansert sikkerhetsteknologi muliggjør samarbeid mellom mennesker og roboter i produksjonsprosesser, og dermed deling av arbeidsområdet [3]. Coboten egner seg derfor godt til undervisning og lab-øvelser. Fagskolen har imidlertid utfordringen med at roboten er helt ny og ikke har blitt grundig testet, og det er heller ikke utviklet noen oppgavesett knyttet til den.



Figur 1 - Fagskolen Vestland sin cobot

Oppdragsgivers sin grunnide for hva løsningen vår skal inneholde er igangkjøring og testkjøring av den samarbeidende roboten. Vi må også gjøre oss kjent med programvare og simuleringsmuligheter, og utvikle et oppgavesett med forskjellig vanskelighetsgrader.

2.4 Cobot

Den største og viktigste forskjellen på en cobot og en industriell robot er at cobot trenger ikke bli adskilt fra mennesker ved hjelp av inngjerding eller andre sikkerhetstiltak. Navnet «cobot» står for collaborative robot (samarbeidene robot), som betyr at den kan jobbe trygt sammen med mennesker. Grunnen for dette er at den er utstyrt med innebygde sikkerhetssensorer og programvare som gjør at den stopper dersom den møter motstand. I motsetning vil en industriell robot bare fortsette til angitt mål, uten å ta noe høyde for det som er rundt den. Det eneste som gjør at en industriell robot vil stoppe er når sikringer slår ut ved høy last, eller når noen manuelt slå den av. Coboter opererer også i en vesentlig lavere hastighet enn industrielle roboter.

I industrien vil en cobot vanligvis ha en arm med flere ledd, og et gripeverktøy for å kunne plukke opp og håndtere gjenstander. Coboter er vanligvis en del av et automatisert produksjonssystem, og blir programmert til å utføre spesifikke oppgaver som er tilpasset produksjonsprosessene til bedriften. Dette kan for eksempel være monotone og repetitive oppgaver som samlebåndsarbeid eller pakking av varer.

International Federation of Robotics (IFR) definerer fire nivåer av samarbeid mellom industrielle roboter og menneskelige arbeiderer [4]:

Sameksistens: Menneske og robot jobber ved siden av hverandre uten gjerde, men uten delt arbeidsområde.

Sekvensielt samarbeid: Menneske og roboten er aktive i delt arbeidsområde, men deres bevegelse er sekvensielle; de arbeider ikke på en del samtidig.

Samarbeid: Robot og menneske arbeider på samme del samtidig, med begge i bevegelse.

Responsivt samarbeid: Roboten reagerer i sanntid på bevegelse av den menneskelige arbeideren.

Vi bruker roboten som en sameksistens robot der vi jobber ved siden av hverandre. Vi deler dermed ikke arbeidsområdet, men har respekt for den når den beveger seg rundt om i arbeidsrommet. Roboten har evnen til å oppnå alle nivåer etter IFR sin definisjon hvis en tar høyde for hva en kan utstyre roboten og arbeidsrommet med, samt utfordringene med å utvikle programmer og kontroller til det.

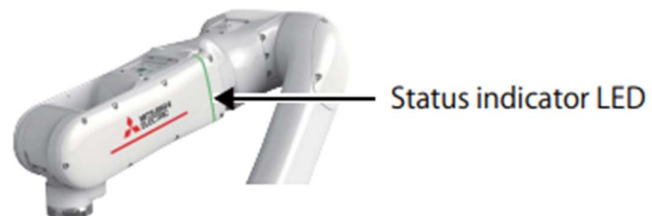
2.5 Mitsubishi - MELFA ASSISTA

MELFA ASSIST er robotserien av samarbeidende roboter til Mitsubishi Electric.

Brukervennlighet gjør at den er enkel i bruk, selv uten spesiell kunnskap om roboter. Denne robotarmen har seks ledd som kan styres på ulike måter. «Direct teaching» er en funksjon som kan aktiveres via operasjonspanelet på en robotarm. Ved å aktivere denne funksjonen kan brukeren enkelt bevege robotarmen manuelt til ønsket posisjon. Når robotarmen er i ønsket posisjon, kan brukeren trykke på «Teach»-knappen, og posisjonen vil lagres i robotens kontroller. Disse lagrede posisjonene kan deretter brukes til å lage et handlingsprogram for roboten. Roboten lar seg også styre via programvaren når man er tilkoblet med PC.

Disse roboten har også mulighet å settes i *high-speed mode* (1000mm/s), når de er i denne modusen er de ikke lengre en cobot. Det stilles da høyere krav til ekstern sikkerhet rundt de, da hastigheten er såpass høy at den kan skade personer og gjenstander den støter på.

Robotarmen har også et LED-lys som indikerer statusen til roboten. På



Figur 2 - LED-lys Indikerer statusen til roboten

fagskolen skal roboten være i

«collaborative mode», de mest vanlige fargene er da:

- Blinkende blått lys: Indikerer at roboten starter opp.
- Konstant blått lys: Indikerer at roboten er i stoppet tilstand i «collaborative mode».
- Grønt blinkende lys: Indikerer at roboten kjører i «collaborative mode».
- Blinkende rødt lys: Indikerer en uventet stopp eller feil.

3 Kravspesifikasjon

Det første og viktigste kravet som danner grunnmuren i hele bacheloroppgaven er å lage lab-oppgaver som gir studentene mestringsfølelse, samtidig som de blir utfordret faglig. Målet er å lære studentene om roboten, bruken av den og få dem til å lære progressivt ved å løse oppgavesettene. Oppgavesettene vil ha ulike vanskelighetsgrader som vil gjøre det lettere for studenter å lære det grunnleggende om roboten før de går i gang med å løse vanskeligere oppgaver. Oppgavesettene skal være rettet mot studenter på fagskolenivå og det skal også følge med løsningsforslag.

3.1 Tilkoblingsguide

Å koble seg til roboten ved hjelp av en LAN-kabel eller router kan være en utfordrende og tidkrevende oppgave. Av denne grunn ønsker oppdragsgiver at vi lager en trinnvis tilkoblingsguide som viser hvordan dette gjøres. Dette vil spare studentene for tid som de heller kan bruke på å løse lab-oppgavene.

Se [appendiks B](#).

3.2 Moduler

I samarbeid med oppdragsgiveren har vi besluttet å utvikle moduler for å løse lab-oppgavene, som enkelt kan festes til bordet hvor robotarmen er fastmontert. Vi mener at dette vil være en hensiktsmessig tilnærming, ettersom modulene vil tillate oss å utarbeide løsningsforslag som sikrer at robotarmens posisjoner alltid stemmer. På denne måten vil det være enklere for læreren på fagskolen å feste modulen til bordet og kjøre løsningsforslaget som en demonstrasjon, før studentene begynner på oppgaven.

3.3 Simulering

Noen av fagene på fagskolen er nettbaserte, med samlinger der tilgangen til den fysiske roboten er begrenset. Oppdragsgiveren ønsket derfor å muliggjøre gjennomføring av lab-oppgaver uten tilgang til den fysiske roboten. Dette kan muliggjøres ved å lage en digital tvilling i Mitsubishi sitt eget programmeringsprogram, dette gjøres ved bruk av 3D-CAD filer som vi må lage av modulene som festes til bordet. På denne måten vil simuleringen se ut som den fysiske modellen og man kan dermed gjennomføre lab-oppgavene digitalt.

4 Analyse av problemet

Etter å ha studert kravspesifikasjonene, har vi fått en ide om hvordan vi ønsker å løse oppgaven vi har blitt tildelt. Vi anser igangkjøring og testkjøring som viktig for å bli kjent med roboten, derfor vil mye tid bli brukt på dette i starten. Oppdragsgiver ønsker at vi i denne fasen lager en tilkoblingsguide for tilkobling med LAN-kabel eller internettruter.

Som tidligere nevnt har vi planer om å utvikle moduler som kan kobles til robotens bord for å løse oppgavene vi lager. Disse modulene vil være forskjellige og laget spesifikt for hver oppgave i oppgavesettet, og det skal være enkelt å feste og fjerne dem fra bordet. For å løse oppgavesettet vil programmeringsprogrammene RT VisualBox og RT ToolBox3 være nødvendige, og disse må installeres på datamaskinen. I tillegg vil det bli utarbeidet en brukerguide for å hjelpe studenter med å komme i gang med roboten og oppgavesettet.

4.1 Utforming av mulig løsning

Fagskolen har ingen spesifikke krav til innholdet i oppgavene vi skal utforme. Dette gir oss frihet til å utvikle forslaget til oppgavene selv, så lenge vi sørger for en naturlig økning i vanskelighetsgraden og legger vekt på det didaktiske innholdet.

4.1.1 Løsningsalternativ 1

Dette alternativet innebærer å utvikle et oppgavesett bestående av tre oppgaver som kan løses både i RT VisualBox og deretter i RT ToolBox3. Hensikten er å gi studentene muligheten til å starte enkelt ved å prøve ut en løsning i RT VisualBox, før de går over til å bruke RT ToolBox3.

En fordel med dette alternativet, er at det blir mer overkommelig for studentene å lære seg å bruke begge programmeringsverktøyene. Ved å gi dem oppgaver som kan løses med begge verktøyene, kan de gradvis bygge opp sin kompetanse og selvtillit med å bli kjent med oppgaven først i en simplere programvare (RT VisualBox), for så å gå over i den tyngre og mer avanserte programvaren (RT ToolBox3). Det kan også være god læring å sammenligne de to løsningene, for å se hvilke fordeler eller ulemper hver av programmene har.

En annen fordel er at studentene vil kunne se mulighetene som RT ToolBox3 tilbyr, og hvordan den kan brukes til å løse en gitt oppgave mer effektivt og komplekst. Ved å gi dem

oppgaver som krever mer avanserte funksjoner, kan de få en dypere forståelse av programvaren og hvordan den kan brukes i virkelige scenarioer.

Det å gi studentene oppgaver som kan løses i både RT VisualBox og RT ToolBox3 er et godt løsningsalternativ. Det vil gi dem muligheten til å gradvis bygge opp sin kompetanse med begge verktøyene, samtidig som de får erfaring med å bruke dem til å løse like typer problemer, med forskjellige fremganger.

4.1.2 Løsningsalternativ 2

Dette løsningsalternativet innebærer å utforme fem oppgaver, hvorav tre av dem skal løses ved hjelp av RT VisualBox og to av dem skal løses ved hjelp av RT ToolBox3. Hensikten med denne tilnærmingen er å skape større variasjon i arbeidet rundt temaet, og bidra til å illustrere forskjellige bruksområder for roboten.

En av fordelene med dette alternativet er at det gir studentene en ny og ukjent utfordring for hver av oppgavene. Dette vil få studentene til å tenke nytt og intuitivt hver gang de går videre i oppgavesettet med å starte på en ny oppgave.

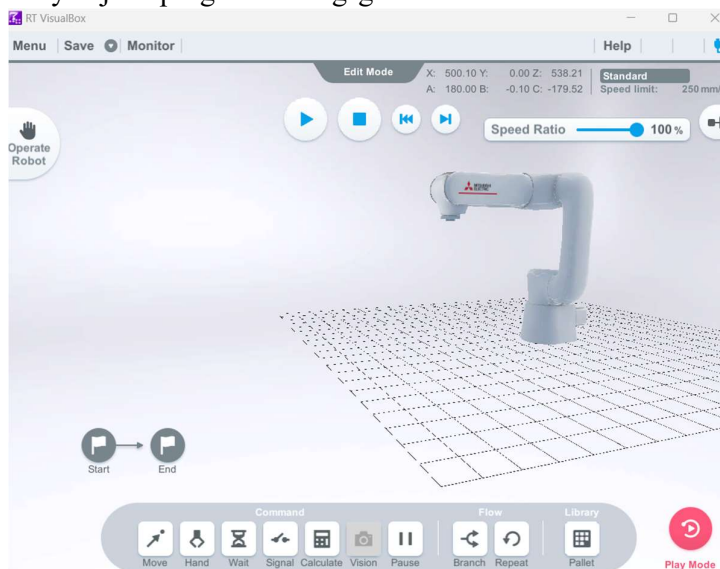
En annen fordel med dette alternativet er at det gir oss muligheten til å utforme oppgavene mer spesifikt. Dette betyr at vi kan belyse færre temaer i hver oppgave, å dermed sette et tydeligere søkelys på det som skal læres.

Det å gi studentene oppgaver som er unikt designet for programmene oppgaven skal løses med, vil være et godt alternativ. Dette vil la oss kunne utforme oppgavene med en tydelig økende vanskelighetsgrad, samt fremme styrkene til de forskjellige programvarene.

4.2 Programvarer

RT VisualBox:

Detter er Mitsubishi sitt egenutvikla programvare for visuell programmering av robotarmen, som lar brukere enkelt lage programmer ved hjelp av et flyt-diagram. Selv om det er noen begrensninger, fungerer programvaren godt for enkle oppgaver og er en god introduksjon til robotarm-programmering. Vi gjorde oss kjent med programvaren i uke 2 under vårt bachelorprosjekt, og det intuitive flytskjemaprogrammeringsgrensesnittet lar brukerne enkelt bygge programmer ved å dra og slippe blokker og angi parametere for funksjonen. Dette resulterer i en kjede av ikoner som viser hvordan roboten vil operere. RT VisualBox kan brukes uten å måtte ha noe tidligere erfaring eller spesifikk kunnskap om robotprogrammering. MELFA ASSISTA kan også

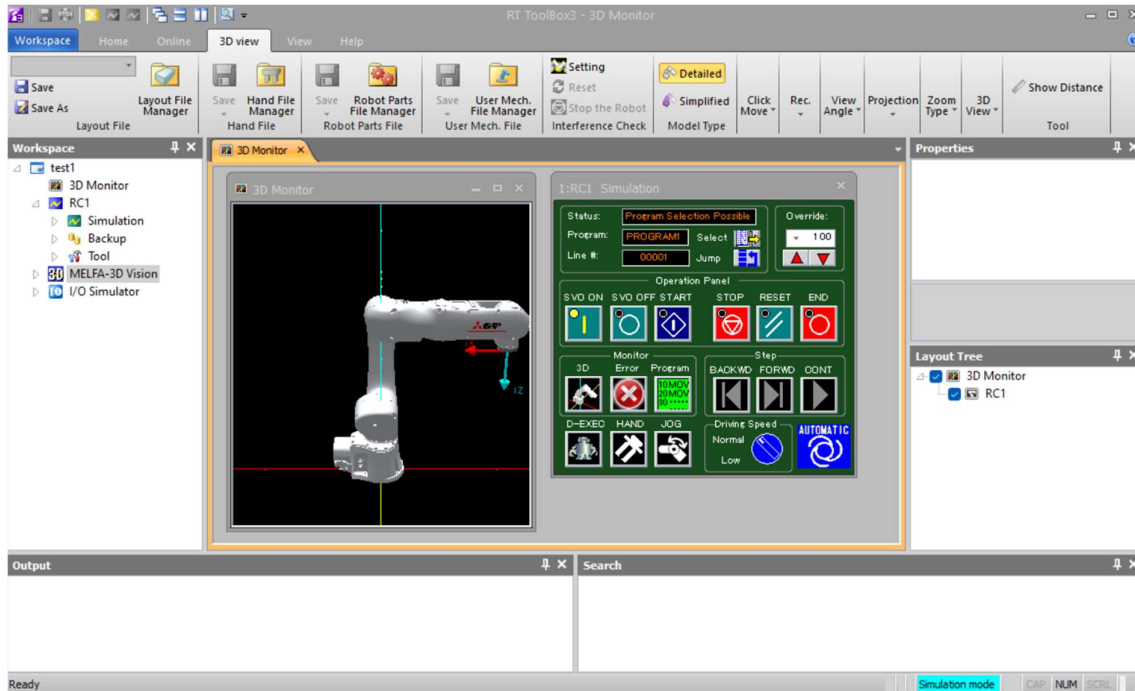


Figur 3 - RT VisualBox

simuleres uten roboten, noe som lar operatører verifisere at programmet ikke har noen syntax feil, uten risiko for roboten.

RT ToolBox3:

Dette er Mitsubishi sin mye mer omfattende programvare for robot-programmering. Den har muligheter for tekstbasert programmering, simulering og samkjøring med andre roboter til Mitsubishi. RT ToolBox3 er programmeringsmiljøet for alle Mitsubishi Electric-roboter. Den gjør det mulig for brukere å lage programmer ved å bruke MELFA BASIC IV, V og VI programmeringsspråk, vi skriver våre program med MELFA BASIC VI. RT ToolBox3 støtter alle prosesser fra systemoppstart til feilsøking og drift, inkludert programmering og redigering, verifisering av driftsomfanget før innføring av en robot, robotfeilsøking før oppstart, og overvåking av robottilstander og funksjonsfeil under drift [5]. RT ToolBox3 kan simulere opprettede programmer for opptil åtte roboter samtidig.



Figur 4 - RT ToolBox3

Autodesk Fusion 360:

Autodesk Fusion 360 er et integrert 3D CAD/CAM-programvare utviklet for design og produksjon. Programvaren tilbyr et omfattende sett med verktøy for modellering, simulering, analyse og produksjon av 3D-modeller. Vi kom frem til at Fusion 360 sin skybaserte tilnærming vil passe oss bra, da det blir enkelt å være flere som jobber mot prosjektet i sanntid. En annen fordel med Fusion 360 er at vi enkelt kan lage målsatte tegninger av produktene vi utvikler.



Figur 5 - Logo Fusion 360

4.3 Maskinvare

Robot arm: Robot armen er en MELFA ASSISTA RV-5AS-D levert av Mitsubishi Electric. Dette er en 6- akset robotarm med mulighet for 5kg arbeids last, den har en rekkevidde på 910mm og en repeterbarhet på $\pm 0.03\text{mm}$ [6]. Armen har innebygde sikkerhetssensorer som sammen med kontrolleren gjør at den kan operere i samme arbeidsrom som mennesker.

TP-Link Archer AX1500: Dette er en router som gjør det mulig å kunne sette opp et lokalt trådløst nettverk, som robotarmen kan koble seg opp mot. Det er vilkårlig hvilken internettruter som kan brukes, dette var en som fagskolen hadde liggende som vi fikk bruke.

Kamera: Vi har etterspurt fagskolen om å anskaffe et kamera som er kompatibelt med RT VisualBox, her var det ikke så mange valg da det står mellom to stykker. Kameraene det står mellom er In-sight 8402 og MELSENSOR VS80, begge er i bunn og grunn det samme kameraet levert av COGNEX. Men VS80 er en utgave gitt ut av Mitsubishi Electric i samarbeid med COGNEX.

Lysgardiner, F3S-TGR-CL4-014-150: Dette er et sett med sikkerhets-lysgardiner som er produsert av Omron. Lysgardinene fungerer ved å sende en stråle fra en transmitter til en mottaker, når noe kommer mellom dem vil forbindelsen mellom dem brytes. Dette vil føre til at sikkerhetssignalet som de sender ut stopper. Ved å utnytte denne funksjonaliteten kan man programmere en controller til å reagere på denne hendelsen, og endre funksjonaliteten til systemet. Dette settet med lysgardiner var også noe fagskolen hadde liggende som de ønsket at vi kunne implementere i forbindelse med en oppgave.



Figur 6 - Lysgardiner

4.4 Konklusjon fra forprosjekt

Etter møte med veilederen og i samråd med oppdragsgiver, falt vi på å gjennomføre løsningsalternativ 1, som går ut på å lage tre oppgaver i stedet for løsningsalternativ 2.

Vi har valgt å fokusere på å lage tre oppgaver av høy kvalitet og bruke tilstrekkelig tid på hver av dem, heller enn å lage fem oppgaver som kanskje ikke ville ha vært like gode. Dette valget er begrunnet med at vi også må sette opp moduler og utarbeide en brukerveiledning for roboten, som også tar tid og ressurser.

5 Realisering av valgt løsning

I dette avsnittet går vi gjennom valgt løsning og diskusjon rundt den. Utforming til lab-oppgaver med løsningsforslag, leveranse til fagskolen og valg knyttet til bygging av modul.

5.1 Oppgavesett

Malen på oppgavene i oppgavesettet vårt er inspirert av utformingen på lab-oppgaver vi selv gjennomførte i robotikk-faget ved Høgskulen på Vestlandet (Robotikk ELE306). Malen vi følger er utformet på følgende måte:

Tema: Her blir de sentrale temaene som skal gjennomgås presentert. Dette er for at studenten skal få et innblikk i hva som er fokuset i den aktuelle lab oppgaven. Temaene som blir tatt opp i oppgavene skal være relevant for arbeid med roboten, og er viktige for læringen om bruken av den.

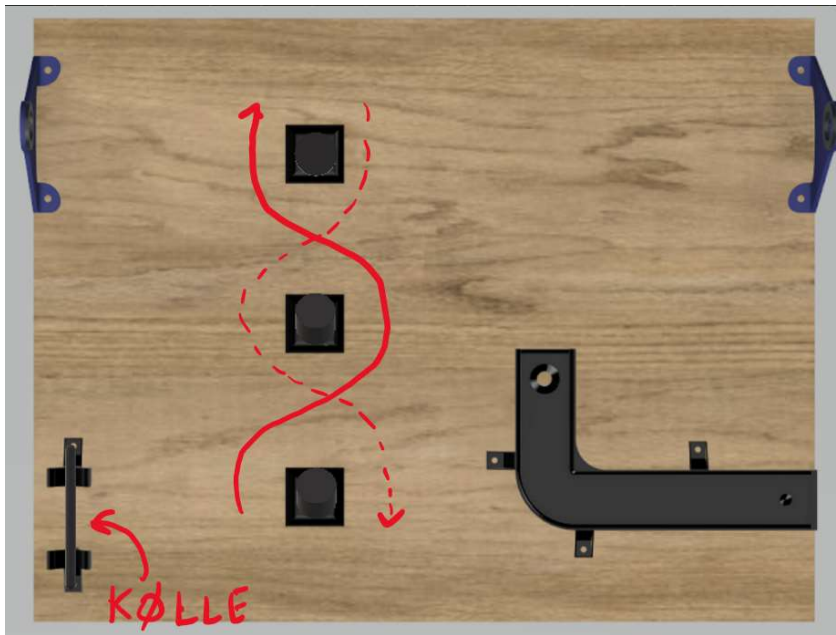
Utstyr: Det er nødvendig å ha ulikt utstyr for å kunne gjennomføre den gitte oppgaven. I tillegg vil man få informasjon om hvilken modul som skal benyttes til lab-oppgaven. Vi har utviklet to moduler for lab-oppgavene; modul 1 skal brukes for lab 1 og 2, mens modul 2 er ment for lab 3. Vi har imidlertid kun utviklet designet for modul 2 og har ikke produsert noen fysiske produkter på grunn av manglende utstyr for det vi hadde planlagt for lab 3.

Oppgavetekst: Her vil oppgaven klart og tydelig beskrive hva studentene skal gjøre, og målet for oppgaven. Oppgaveteksten vil legge vekt på at studentene skal skrive koden selv og bli godt kjent med programmene som er knyttet til roboten.

Spørsmål: Som en avsluttende del av oppgaven vil det bli stilt refleksjonsspørsmål som tar opp temaer og aspekter som studenten har gjennomgått i oppgaven. I tillegg kan spørsmålene omhandle generell kunnskap som studenten skal ha opparbeidet seg gjennom faget. Målet med disse spørsmålene er å evaluere om studenten har fulgt med under gjennomføringen av oppgaven og om de har lært noe i faget. Dette vil også gi studentene muligheten til å reflektere over sin egen læring og identifisere områder som de kan jobbe videre med for å forbedre sin forståelse av emnet.

5.1.1 Laboratorieøving 1

Denne lab-oppgaven går ut på å programmere roboten til at den skal plukke opp en kølle ved hjelp av den påmontert griperen, deretter lede køllen gjennom en slalåmløype som består av tre søyler. Oppgaven krever at roboten beveger seg gjennom løypen tre ganger frem og tilbake, uten å velte noen av søylene. Til slutt skal robotarmen sette køllen tilbake på sin opprinnelige plass.

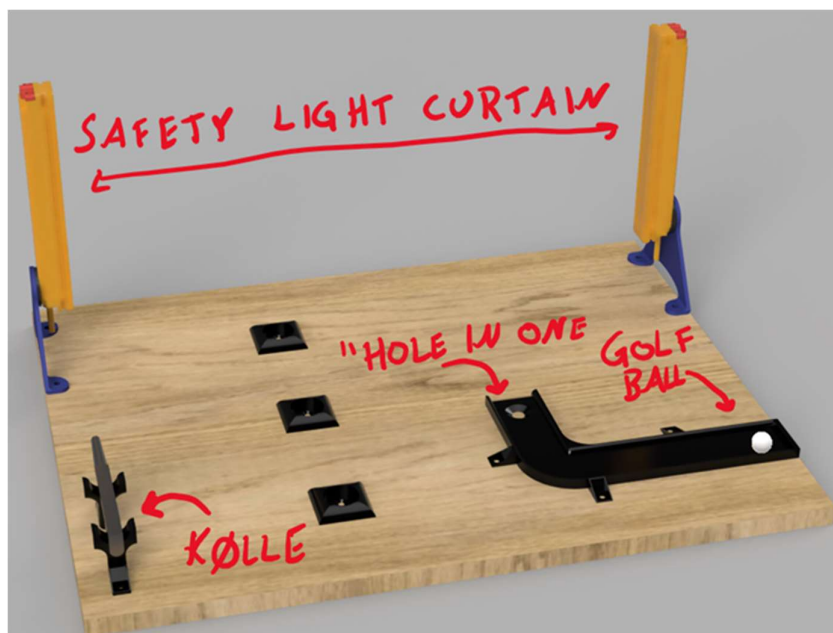


Figur 7 - Illustrasjon av lab oppgave 1

Målet for oppgaven er å sette lys på de fundamentale delene når det kommer til programmeringen av robotarmen. Samtidig som det skal være en morsom og enkel inngang til robotprogrammering, dette er for å skape interesse for videre læring. De fundamentale temaene som studentene skal lære av oppgaven er hvordan man skal lære roboten waypoints og hvordan de brukes, forståelse av hvordan roboten beveger seg mellom punktene (waypoint), bruk av griper, repetering av et gitt handlingsmønster (loop) og lære seg å kjenne til forskjellen mellom lineær og ulineær bevegelse.

5.1.2 Laboratorieøving 2

Denne oppgaven går ut på å støte en golfball ned i hullet på løypen som er montert på modul 1, dette skal gjøres ved å plukke opp køllen å bruke den til å støte golfballen. Her er fokuset rettet mot temaene sikkerhet, safety IO (sikkerhets inn- og utgang), samt nøyaktighet og presisjon. Utfordringen med øvingen er å finne ut hvor mye kraft en trenger for å klare å få ballen ned i hullet, det er også viktig å være presis i valg av bevegelse. Sikkerhet spiller også en viktig rolle i denne øvelsen, her skal det installeres sikkerhetsgardiner (safety light curtain). Disse gardinene vil endre signal dersom noe kommer mellom dem og barrieren blir brutt, roboten skal ved et slikt brudd senke hastigheten eller stoppe fullstendig.

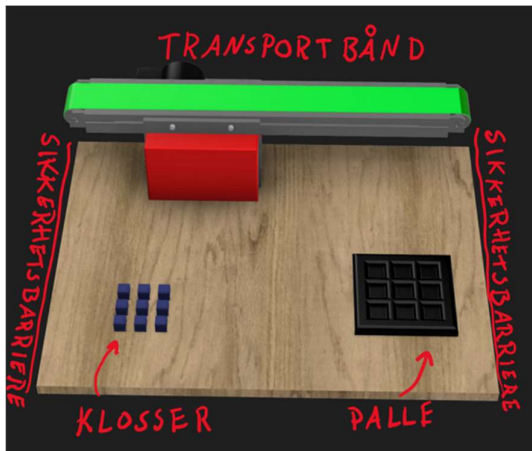


Figur 8 - Illustrasjon av lab-oppgave 2

Målet for oppgaven er å sette lys på nøyaktigheten en robot har og repeterbarevnen dens. Den setter også lys mot viktigheten og funksjoner til sikkerhetstiltak ved bruk av robotarmer generelt. Siden dette er en samarbeidende robot er det egentlig overflødig med lysgardiner og ekstra sikkerhetstiltak, da dette allerede er ivaretatt med innebygde sensorer i roboten (som nevnt i [avsnitt 2.4](#)). Likevel skader det ikke å ha ekstra tiltak, men dette er hovedsakelig for læringens skyld.

5.1.3 Laboratorieøving 3

Denne oppgaven går ut på å sortere klosser som blir fraktet på et transportbånd. Klossene blir matet manuelt på transportbåndet og blir detektert av en sensor som sender et signal til armen. Klossene skal videre bli plukket opp ved hjelp av griperen og sortert på en palle. Videre skal roboten plassere klossene tilbake til sin originale startposisjon. En skal og sette opp virtuelle sikkerhetsbarrierer på begge sidene av modulen som roboten ikke skal kunne krysse.

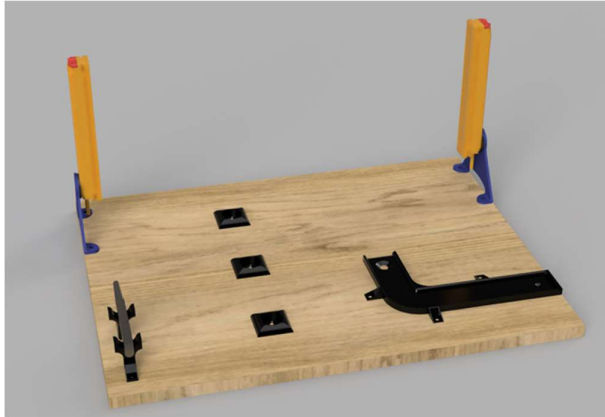


Figur 9 - Illustrasjon av lab oppgave 3.

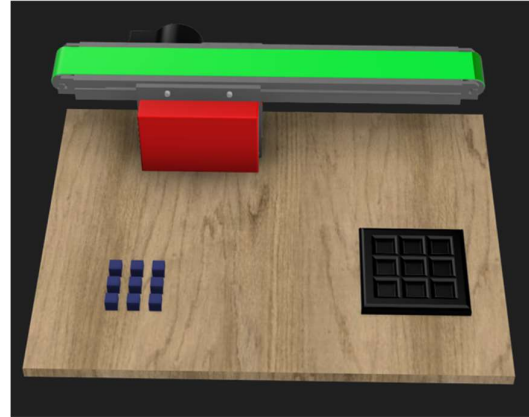
Målet med denne lab-oppgaven er å gi studentene en oppgave som representerer en reell situasjon i industrien der en vil benytte en robot. Sentrale temaer i denne øvelsen er pallesortering, bruk av IO og sikkerhetssoner. I RT VisualBox og RT Toolbox3 finnes det innebygde funksjoner som hjelper til med pallesortering, og disse kan benyttes i denne øvelsen. Transportbåndet og sensoren må kobles til robotens IO-kort, slik at studentene får trening i å håndtere signaler som roboten skal reager mot. De virtuelle sikkerhetssonene fungerer som en virtuell vegg som kan defineres i programmeringsprogrammene. Dette skal være et hinder som gjør at armen ikke skal bevege seg inn i uønsket arbeidsrom. Sikkerhetssonene på begge sidene er god læring, men som i lab-oppgave 2, er ikke sikkerhetstiltak nødvendige ettersom det er en samarbeidende robot.

5.2 Leveranse

I dette avsnitte vil vi gå igjennom det som skal leveres til fagskolen, utenom oppgavesettene. Til laboratorieøvingene skal det leveres to moduler, lab-oppgave 1 og 2 blir utført på samme modul og en egen modul til lab-oppgave 3.



Figur 11 - Illustrasjon av modul 1



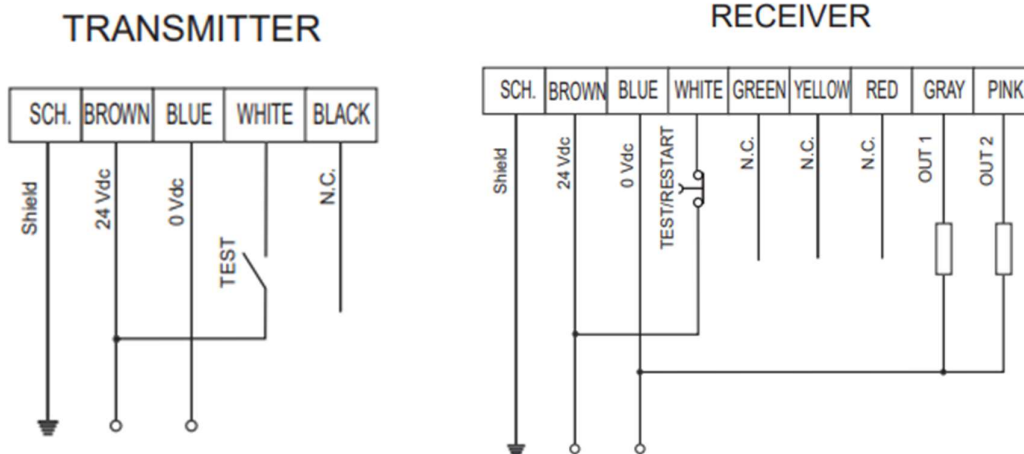
Figur 10 - Illustrasjon av en mulig modul 2

5.2.1 Modul

Vi bygget moduler ut fra en stor finerplate, denne delte vi opp i biter som passet dimensjonen på bordet til roboten. Sammenstillingen av modul 1 var uten noen særlige problemer, da alle elementene ble 3D-printet. Modul 2 derimot ble aldri ferdigstilt da vi manglet transportbåndet oppgave tre var sentrert rundt.

På undersiden av platen monterte vi to hylleknekter på hjørnene mot roboten, dette for kunne skru modulen fast i bordet hvor roboten står montert. For å gi ekstra støtte monterte vi også et sammenleggbart ben under modulen.

Lysgardinene som brukes på modul 1 består av en transmitter og en mottaker, de kan enkelt monteres/demonteres på modulen ved å løsne festeskruen og plugge fra kablene. Kablene er ferdig koblet via et styreskap til roboten, på skapet er det også montert test- og reset-knapper som fungerer mot gardinene. Skapet i seg selv har vi montert til bordet med roboten, med den samme typen T-bolter som brukes til å feste modulene.



Figur 13 - Kablingsskjema for transmitteren

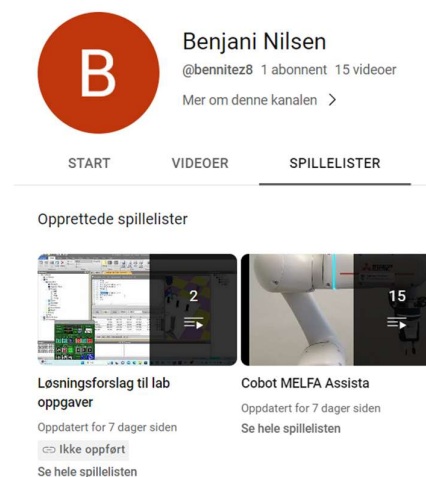
Figur 12 - Kablingsskjema for mottakeren

5.2.2 Klargjort programfil

Sammen med leveransen til fagskolen har vi også laget en programfil til RT ToolBox3, denne inneholder løsning på lab-oppgave 1 og 2, nøyaktig 3D modell av roboten sammen med modul 1 og griperen som er montert på armen (digital tvilling). Dette er gjort for at det skal være enkelt å kunne komme i gang med den faktiske programmeringen, uten å måtte sette seg inn i detaljene rundt modellering i RT ToolBox3. Vårt forslag vil være å ta ut løsningen og levere ut programmet ferdig oppsatt til studenter når de skal løse oppgavene.

5.2.3 Videoer

For at det skal være enkelt for studentene ved fagskolen å finne nyttig info, valgte vi å lage videoer og lastet dem opp på YouTube. Vi har utarbeidet steg-for-steg videoinstruksjoner om hvordan man kobler roboten opp mot datamaskinen, samt grunnleggende veiledning i bruken av RT ToolBox3. I tillegg har vi utarbeidet løsningsforslag til de ulike lab-oppgavene. Lærer ved fagskolen har selv lagt ut gode videoer av bruken av RT VisualBox som vi brukte som inspirasjon og vi lagde derfor ikke videoer rundt dette programmet.



Figur 14 – Utklipp av YouTube kanalen

Ifølge Evan Ackerman, en skribent i IEEE Spectrum med omfattende erfaring innen redigering av robotvideoer og som har sett titusenvis av videoer selv, bør en video være på maksimum tre minutter [7]. Ackerman understreker også viktigheten av å huske på publikumet for videoene, og at videoene bør være på et nivå som alle kan forstå, uansett hvor mye erfaring man har innen roboter og robotprogrammering. For å sikre at studentene opprettholder fokus og interesse, har vi derfor gjort en innsats for å holde videoene korte, enkle og delt dem opp i mindre deler dersom de overstiger tre minutter.

Videoene ligger på YouTube kanalen: Benjani Nilsen (@bennitez8).

Spilleliste med veiledningsvideoer:

https://www.youtube.com/playlist?list=PL1JlioxNIquHqAJBX_8ByCcYvFL_Srzzi

Spilleliste med gjennomgang av løsning til lab-oppgaver (kun tilgjengelig med lenke):

https://www.youtube.com/playlist?list=PL1JlioxNIquEXDShAzNClmn_p2XOaz-kj

5.2.4 Oppstarts manual

Oppstarts manual ble som tidligere nevnt laget etter ønske fra oppdragsgiver. Her vises trinnvis hvordan en kobler seg til roboten med LAN-kabel og trådløs router. Dette kan være utfordrende fordi subnettmasken på datamaskinen må korrelere med robotens. Oppstarts manual ble laget tidlig i bachelorprosjektet vårt da vi selv hadde utfordringer med å få kontakt med roboten, den har dermed vært et nyttig hjelpemiddel for vår egen del. For å ha den lett tilgjengelig valgte vi både å ha en skriftlig trinnvis gjennomgang og en video som viser hvordan dette gjøres, da kan studentene selv velge hvilket hjelpemiddel de ønsker å benytte seg av.

Se [appendiks B](#).

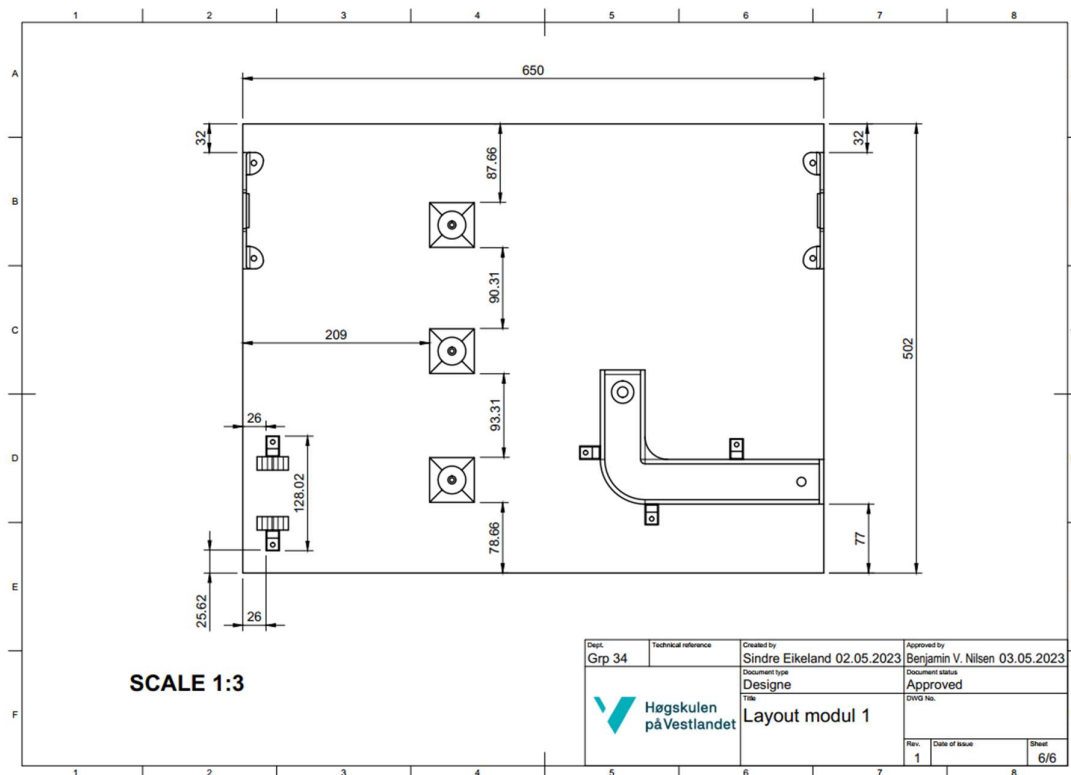
5.2.5 Veiledninger til underviser

Veiledningene ble utviklet samtidig som vi selv løste oppgavene, slik at eventuelle utfordringer som oppsto underveis ble inkludert som tips. Veiledningen er ment som et hjelpemiddel for undervisere, og gir innsikt i hvordan vi ser for oss at oppgavene kan løses på en god måte. Med veiledningen kan underviseren bruke vår erfaring til å veilede studentene i å oppnå en vellykket løsning. Vi har også inkludert vårt eget løsningsforslag for oppgavene i veiledningene.

Se [appendiks D](#).

5.2.6 Design

Fagskolen uttrykket også et ønske om målsatte tegninger over konstruksjonen som ble satt sammen for oppgavene, dette for å kunne rekonstruer de ved behov eller ved gjenoppbygging i andre materialer. Dette ble etterkommet og levert sammen med CAD-filer over konstruksjonene og objektene vi har 3D-printet.



Figur 15 - Tegning over modul 1

Se [appendiks C](#).

6 Testing

I dette avsnittet vil vi gjennomgå ulike tester som er gjennomført i prosjektet. Deriblant testing av modulene, virtuell modell samt test av lab-oppgaver mot studenter ved fagskolen.

6.1 Test av fysisk utstyr

Modulene måtte monteres og testet før vi kunne ferdigstille oppgavene med løsningsforslag. Etter nøye vurdering falt valget av festemetoder på en festeanordning med t-bolter som tres inn i profilen til bordet. En utfordring som ble oppdaget under testing av modulene var at gulvet ved fagskolens avdeling på Nordnes er noe ujevnt. Dette er da noe en må ta høyde for i lab-oppgave 2 der en skal støte en ball.

Lysgardinene som ble brukt i lab-øving 2 ble hentet fra et tidligere hovedprosjekt ved fagskolen. Under testing av lysgardinene oppdaget vi at dersom noen brøt forbindelsen mellom lysgardinene, måtte feilen tilbakestilles manuelt, noe vi ønsket å unngå. Vi gjorde undersøkelser i manualen til lysgardinene og fant ut at «dip-switch» nummer 8 på mottakersiden var satt feil i forhold til det vi ønsket å oppnå. Etter at dette ble rettet opp, fungerte lysgardinene som ønsket, og vi monterte dem på modulen for lab-oppgave 2.



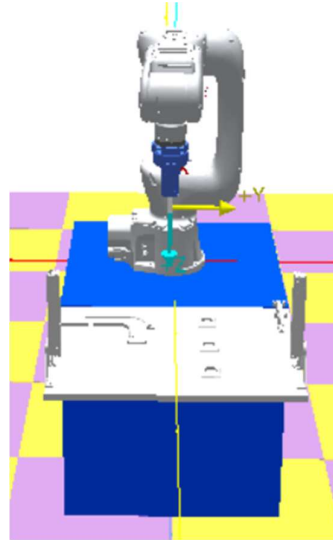
Figur 16 – Festeanordning med T-bolter

6.2 Test av virtuell modell

Etter å ha utviklet en virtuell modell av modulen for lab-oppgave 1 og 2, var det nødvendig å teste modellen. Vi testet den ved å kjøre roboten til ulike steder på modulen og sammenligne resultatene med modellen i simulatoren. Basert på sammenligningen foretok vi nødvendige justeringer på modellen. Til slutt oppnådde vi en tilfredsstillende modell av roboten sammen med modulen, som nå gjør det mulig å løse oppgaven i simulatoren og deretter overføre programmet til den fysiske roboten uten store tilpasninger.



Figur 18 - Fysiske modellen



Figur 17 - Virtuelle modellen

6.3 Test gjennomført mot studenter

Vi fikk anledning å teste oppgavene mot studenter ved fagskolen, ved to forskjellige anledninger. Hovedformålet med testingen var å sjekke at formuleringen av oppgavene var tydelig og velformulert, samt at vanskelighetsgraden var passende. Dette ville bidra til å forhindre misforståelser eller feiltolkninger av hvordan oppgavene skulle løses. Ved testing fikk vi også et forhold til tidsbruken studentene brukte på oppgavene, og om tidsbruken samsvarte med det vi så for oss. Studentene brukte rundt en og en halv time til to timer på oppgavene, dette stemte bra med det vi hadde forespeilet.

7 Diskusjon

Vi har holdt oss til og fulgt den opprinnelige fremdriftsplanen som baserer seg på en Gantt-plan gjennom semesteret. Denne planen viser hvor lang tid som skal brukes og når de forskjellige arbeidsoppgavene skal gjennomføres og være gjennomført. Vi har også laget en timeliste hvor en legger inn hvor mange timer en har arbeidet den dagen med prosjektet.

7.1 Vurdere valgt løsning

Valget vårt var å fokusere på å lage tre gode lab-oppgaver som skulle bli løst med begge programvarene (RT VisualBox og RT ToolBox3). Grunnen til at denne løsningen ble valgt er fordi det tar tid å teste og lage gode oppgaver, samt at studentene skal få god kjennskap til begge programvarene. Dette viste seg og vær en god vurdering, vi fikk tid til å konstruere gode oppgaver som går innom de sentrale temaene i robotprogrammering. Simuleringen av lab-oppgavene fungerte også veldig bra, programmet trengte ingen videre tilpasninger i forhold til posisjonen til modulen, kun kraften på støtet i lab-oppgave 2 måtte tilpasses.

Vi diskuterte i gruppen ulike løsninger for å feste og bytte ut modulene og endte opp med en enkel og effektiv løsning. Vi bruker T-hodebolter som festes i profilene på robotbordet for å feste modulene. Disse boltene tres gjennom en hylleknekt som er skrudd fast på modulen, og festes videre i profilene med en mutter. I tillegg har modulen et ben som kan slås ut for å gi ekstra støtte når det legges litt press på den. På denne måten har modulen tre punkter som holder den på plass og hindrer den i å flytte på seg eller falle sammen mens man utfører lab-oppgavene. Vi diskuterte mange alternative metoder for å feste modulene, men fant ut at denne løsningen var den mest praktiske og pålitelige.

7.2 Problem undervegs

Det tok rundt fire veker før vi fikk tilgang til RT ToolBox3, samt at det tok en del måneder før griperen kom og de fikk koblet den til roboten. Vi ble tidlig ferdig med å gjøre oss kjent med RT VisualBox og ventet bare på at vi skulle få tilgang til RT ToolBox3. Dette førte til at det tok litt tid før en fikk testet modulen som vi hadde laget for lab oppgavene. Søking av midler til transportbånd og kamera/sensor tar lang tid og utstyret som skulle vært brukt til modulen i lab-oppgave 3 kommer ikke før semesteret er over. Det var også noen problemer når studentene skulle gå igjennom og teste lab-oppgave 2. Gulvet på fagskolen er noe skeivt og det førte til litt problemer med presisjonsdelen av laben, men med litt justeringer så gikk gjennomgangen ganske bra.

7.3 Videre arbeid og utvidelser

Som nevnt tidligere kom ikke transportbåndet som skulle brukes til realisering av lab-oppgave 3, dette skulle da bli en del av modul 2. Når transportbåndet eventuelt kommer, vil det være naturlig å ferdigstille modul 2, slik at lab-oppgave 3 som vi har laget kan gjennomføres av studenter ved fagskolen.

Modulene har vi laget i trevirke, et tidlig ønske fra oppdragsgiver var at dette skulle gjøres i metal som sto i stil med roboten og bordet den stod på. Vi kom frem til at dette ville ta lang tid å realisere med tanke på vår tidsramme, og ut ifra vår kompetanse. Dette er en av grunnene for at vi har laget grundige designtegninger og modeller som gjør det mulig for oppdragsgiver ved en senere anledning å få det produsert i metal om ønskelig.

Utvidelse av oppgavesettet er en noe en kan vurdere etter hvert som læreplanen og faget utvikler seg. Kun fantasien setter en stopper på hvilke lab-oppgaver en kan lage, men det er viktig å ta høyde for at det stå i stil med studiets rammer og målet for utdanningen.

7.4 Pris

Vi har i stor grad benyttet oss av eksisterende utstyr og materiale fra fagskolen i prosjektet, og har derfor ikke brukt mye penger. For å lage modulen brukte vi en finerplate som allerede var tilgjengelig på skolen, og fagskolen kjøpte også en lekt som ble brukt for å lage beina til modulene. Lysgardinene vi brukte var også tilgjengelige etter en tidligere oppgave som har vært gjennomført på skolen. De eneste kostnadene vi hadde var knyttet til hyllekneker som vi brukte for å feste modulene til bordet, foldbare braketter for å feste beina til modulene og noen skruer for å feste komponentene vi lagde for lab-oppgavene. Under har vi lagt inn budsjettet for oppgaven, dette inkludere blant annet kostnader fagskolen allerede har lagt ut med anskaffelse av roboten, og utstyr til den.

Antall studenter		3					
Bedrift	Fagskulen Vestland						
HVL Veileder	Tom Kjøde						
				Hvem skal betale ?			
#	Beskrivelse av utgift	Budsjettert	Kost	Studenter	Fagskolen	HVL	
1	Robotarm pakke (Robotarm, bord, gripar, RTVisualBox	218000	218000		x		
2	RTToolBox3	3000	3000		x		
3	Mini USB - USB 2.0 kabel	150	0		x		
4	Bosch Rexroth M6 T-Head Bolt Connecting Component		198,61		x		
5	Diverse feste material		358	x			
6	Trelast		370		x		
	Totalt	221150		358	221198,61		0

Figur 19 - Budsjett for oppgaven

8 Konklusjon

Dette bachelorprosjektet hadde som hovedmål å produsere et forslag til et undervisningsopplegg for fagskolen, og vi har resultert i tre lab-oppgaver med tilhørende løsningsveiledning. Alle oppgavene kan løses både i RT VisualBox og RT ToolBox3, videre har vi utviklet moduler til oppgavene som skal festes til bordet hvor roboten er montert. I tillegg har vi utviklet tilkoblingsguide og videoer som vil være nyttige hjelpemidler for studenter og tilsette ved fagskolen. Vi har også lagt til rette for at alle studenter skal ha mulighet til å simulere lab-oppgavene i RT Toolbox3 uten tilgang til den fysiske roboten.

Gjennom dette prosjektet har vi utviklet en verdifull kompetanse når det gjelder å håndtere større prosjekter, fra start til slutt. I forstudiet ble det utviklet en god fremdriftsplan og timeplan som vi har fulgt gjennom prosjektet. Vi har også lært å samarbeide godt som et team, inkludert å håndtere teknisk krevende oppgaver sammen og fordele arbeidsoppgaver på en effektiv måte. Som gruppe har vi lært å jobbe mot krav på en god måte for å sikre at vi når målene som er satt.

Vi mener at prosjektet oppfyller alle kravene som ble satt av oppdragsgiver, og vi er sikre på at lab-oppgavene som er utviklet vil bidra til et godt og interessant læringsmiljø for fremtidige studenter ved fagskolen, om de tas i bruk til undervisning. Vi har lagt ned mye arbeid i å utvikle oppgavesettet og tilhørende materiale, og vi håper det vil være til stor nytte for lærere og kommende studenter ved fagskolen.

Når fagskolen skal bestille nytt utstyr erfarte vi at dette kan være en tidkrevende prosess. Dessverre tok det så lang tid å bestille transportbåndet at vi ikke fikk det i tide, med tanke på levering av bachelorrapporten. Som et resultat av dette manglende transportbåndet, som skal festes til sin respektive modul for lab-oppgave 3, fikk vi ikke realisert dette. Selv om lab-oppgave 3 kan simuleres i RT ToolBox3, har vi ikke implementert dette, da vi ikke kjenner det faktiske transportbåndet som kommer. Dette vil uansett medføre at 3D modellen vil måtte revideres ved en eventuell ferdigstilling av modulen.

Referanser

[1] Tekna «Helt enkelt: Hva er en digital tvilling – og hva kan den brukes til?» tekna.no.

Hentet fra: <https://www.tekna.no/kurs/innhold/helt-enkelt-hva-er-en-digital-tvilling--og-hva-kan-den-brukes-til/> (lastet ned: 28.04.2023).

[2] Fagskulen Vestland «Om fagskulen Vestland» fagskulen.no.

Hentet fra: <https://www.fagskulen.no/om-oss/om-fagskulen-vestland/om-fagskulen/> (Lastet ned: 17.01.2023).

[3] Mitsubishi Electric «Collaborative Robot - MELFA ASSISTA» mitsubishielectric.com.

Hentet fra: https://no.mitsubishielectric.com/fa/no_en/products/rbt/assista#product-bl7b48f6f637a37e517 (Lastet ned: 17.01.2023).

[4] Ben Pelkington «Collaborative Robots and Cobot Technology» azorobotics.com.

Hentet fra: <https://www.azorobotics.com/Article.aspx?ArticleID=442> (Lastet ned: 20.04.2023).

[5] Mitsubishi Electric «RT ToolBox3 - Total ingeniørstøtte» mitsubishielectric.com.

Hentet fra: https://no.mitsubishielectric.com/fa/products/rbt/robot/robot-engineering-software/rt_toolbox3 (Lastet ned: 12.04.2023).

[6] Mitsubishi Electric «RV-5AS-D - FA-Robot: Robotssystem» mitsubishielectric.com.

Hentet fra: <https://emea.mitsubishielectric.com/fa/products/rbt/assista/assista/rv-5as-d.html> (Lastet ned: 08.05.2023).

[7] Evan Ackerman «How to Make a Good Robot Video» ieee-ras.org. Hentet fra:

<https://www.ieee-ras.org/publications/how-to-make-a-good-robot-video?fbclid=IwAR2i0d4Iq3LvZ9Rpwmqd074sY7wv0wex8XjmnBXWZlvKhwi8zgTkDWwp0AY> (Lastet ned: 16.03.2023).

Figurliste

Figur 1 - Fagskolen Vestland sin cobot.....	9
Figur 2 - LED-lys Indikerer statusen til roboten	11
Figur 3 - RT VisualBox.....	16
Figur 4 - RT ToolBox3.....	17
Figur 5 - Logo Fusion 360.....	17
Figur 6 - Lysgardiner.....	18
Figur 7 - Illustrasjon av lab oppgave 1	20
Figur 8 - Illustrasjon av lab oppgave 2.....	21
Figur 9 - Illustrasjon av lab oppgave 3.....	22
Figur 10 - Illustrasjon av en mulig modul 2	23
Figur 11 - Illustrasjon av modul 1	23
Figur 12 - Koblingsskjema for mottakeren	24
Figur 13 - Koblingsskjema for transmitteren	24
Figur 14 – Utklipp av YouTube kanalen.....	24
Figur 15 - Tegning over modul 1	26
Figur 16 – Festeordning med T-bolter.....	27
Figur 17 - Virtuelle modellen.....	28
Figur 18 - Fysiske modellen.....	28
Figur 19 - Budsjett for oppgaven.....	31

Appendiks A Prosjektledelse og styring

Aktiviteter som har med prosjektgjennomføringen å gjøre passer kanskje ikke så godt å beskrive i de andre kapitlene. Det kan likevel være greit å si noe om hvordan prosjektet er gjennomført. Dette kan gjøres her.

A.1 Prosjektorganisasjon

Prosjektleder: Sindre Eikeland

Vi har hatt individuelle hovedansvars områder, der den gitte personen har hatt overordnet ansvar. Det har allikevel utartet et nokså flytende arbeidsbidrag på de fleste områder, dette har vist seg å fungere godt og har kommet fra et tett samarbeid. Mye av dette skyldes at vi har samlet oss og jobbet fysisk sammen under heile prosjektet.

Inndeling:

- Sindre Eikeland: Oppgaveutforming, video innspilling, sammensetting av modul, CAD design, programmering,
- Benjamin Våge Nilsen: Oppgaveutforming, tilkoblingsguide, video innspilling og redigering, sammensetting av modul, programmering
- Ådne Måkestad: Hovedansvar oppgaveutforming, sammensetting av modul, budsjett, utvikle veiledning til oppgaver

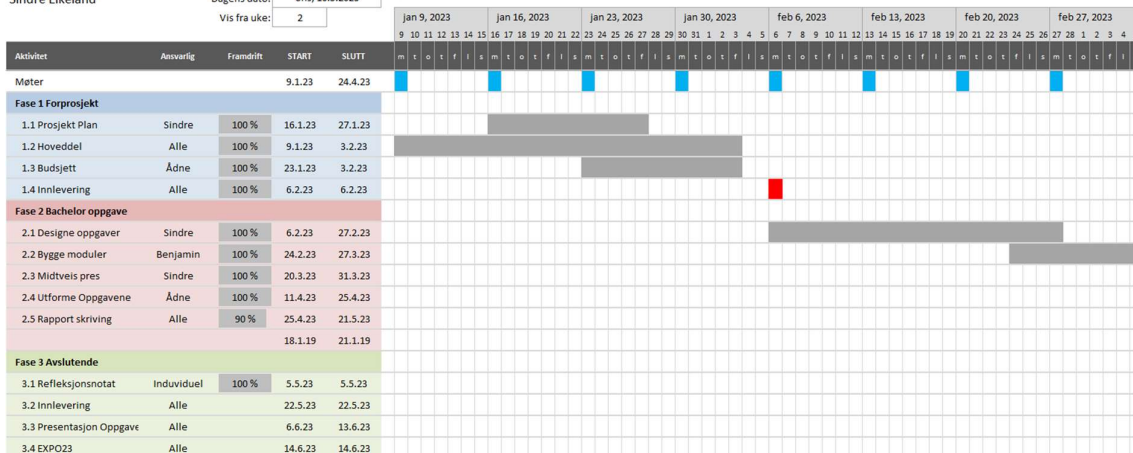
A.2 Fremdriftsplan

I forbindelse med forprosjektet utviklet vi en GANTT plan for prosjektet, denne var nokså generell og vi har fulgt den tilnærmet til punkt og prikke. Planen er bygd på en mal utviklet av Vertex42 (<https://www.vertex42.com/ExcelTemplates/simple-gantt-chart.html>).

Mitsubizhi Cobot

HVL/Fagskolen Vestland
Sindre Eikeland

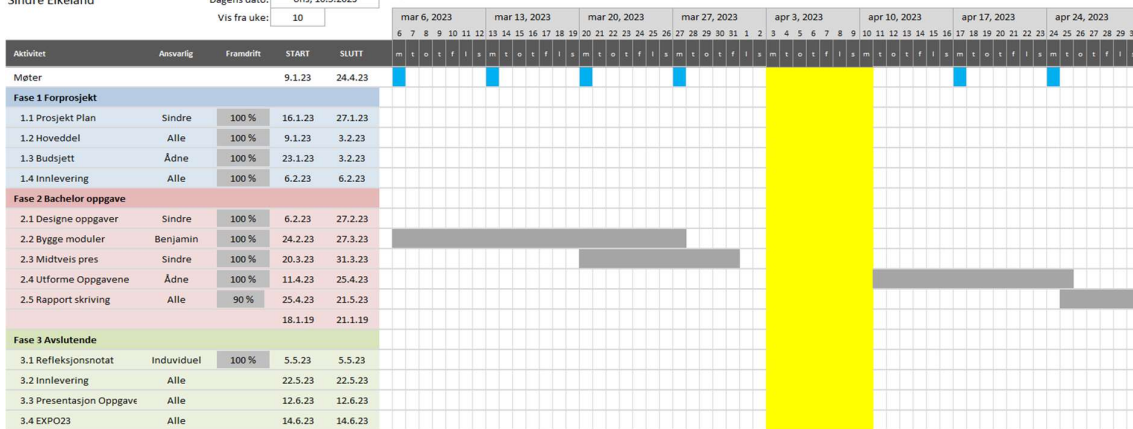
Prosjekt Start: man, 2.1.2023
Dagens dato: ons, 10.5.2023
Vis fra uke: 2



Mitsubizhi Cobot

HVL/Fagskolen Vestland
Sindre Eikeland

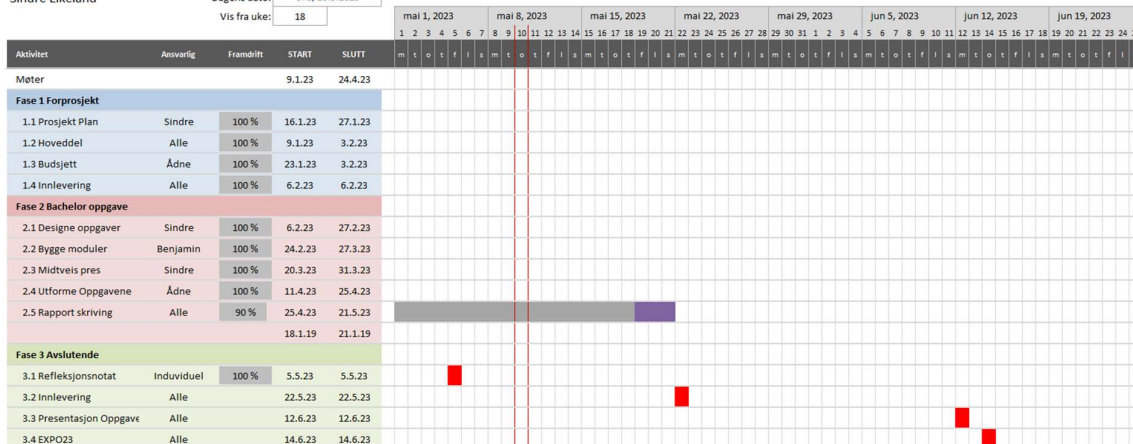
Prosjekt Start: man, 2.1.2023
Dagens dato: ons, 10.5.2023
Vis fra uke: 10



Mitsubizhi Cobot

HVL/Fagskolen Vestland
Sindre Eikeland

Prosjekt Start: man, 2.1.2023
Dagens dato: ons, 10.5.2023
Vis fra uke: 18



A.3 Risikoliste

Risiko	Sannsynlighet	Alvorlighet	Konsekvens	Tiltak
Sykdom og/eller skade	Lav	Middels	Hindrer fremdrift i prosjektet.	Hjemmekontor om gjennomførbart.
Konflikt innad i gruppen	Lav	Middels	Fører til dårligere samarbeid. Hindrer fremdrift i prosjektet.	Snakke med hverandre om eventuelle problemer og konflikter.
Konflikt med arbeidsgiver	Lav	Høy	Resultatet blir dårligere. Lavere karakter/utbytte.	Jevnlige dialoger med veileder og arbeidsgivere.
Katastrofale feil med utstyr	Svært lav	Svært Høy	Alvorlige skader	Sikkerhetstiltak mot arbeid med utstyret

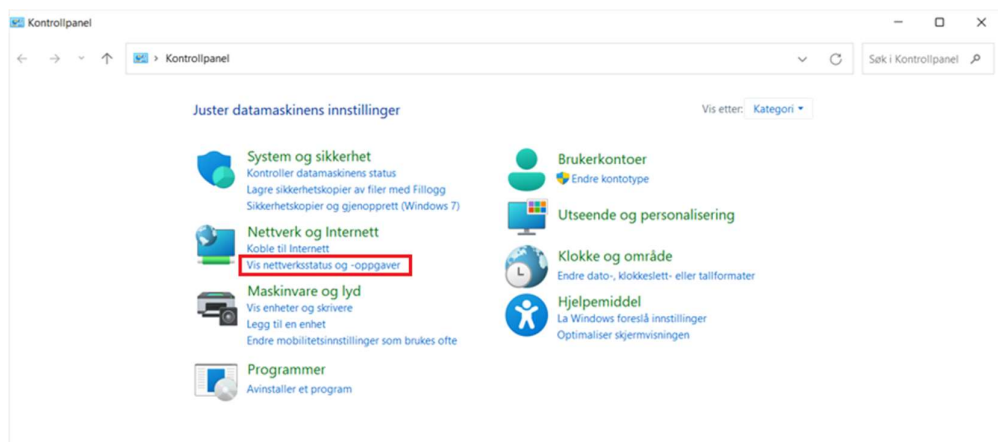
Appendiks B Tilkoblingsguide

Viser hvordan en kobler seg til coboten med LAN-kabel eller router i programmeringsprogramma RT Visualbox og RT Toolbox3

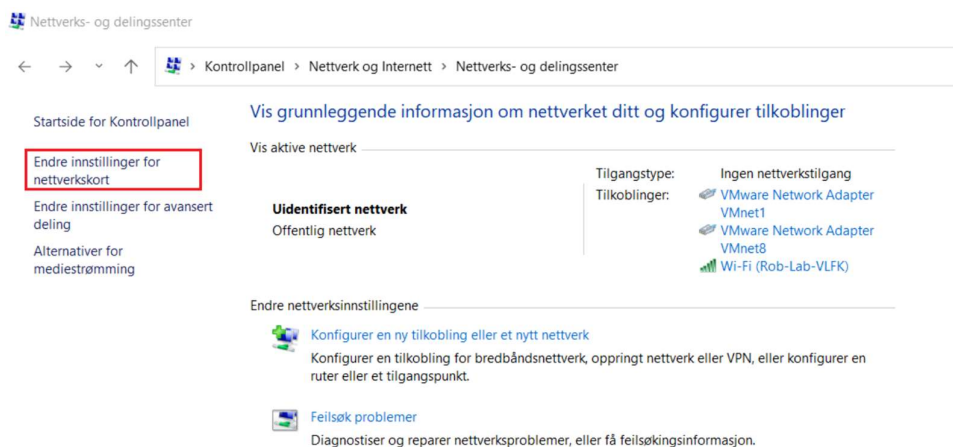
B.1 Tilkoblingsguide RT VisualBox

Tilkoblingsguide med router:

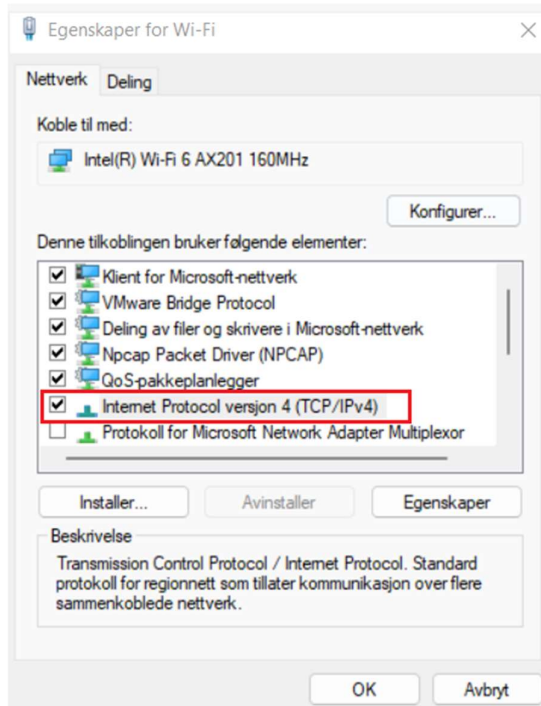
1. Sett i router og koble LAN kabel fra roboten til en av LAN-portene i router.
2. Logg inn på routeren for å konfigurere subnett masken til å være 255.255.255.0
3. Søk på kontrollpanel på din PC og trykk på «Vis nettverksstatus og -oppgaver»



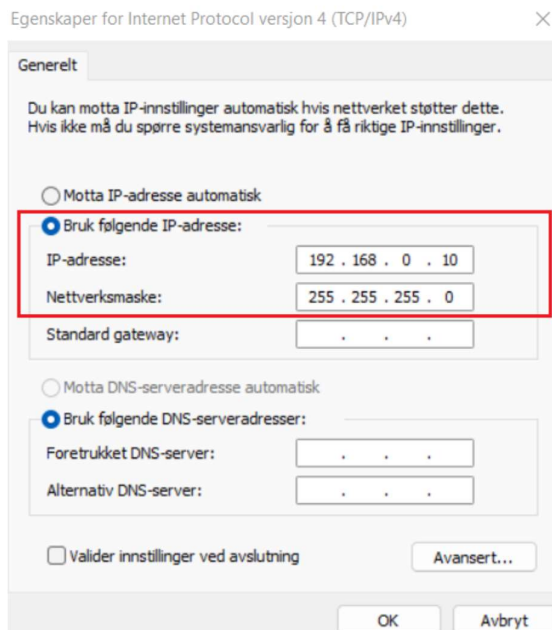
4. trykk på «Endre innstilling for nettverkskort»



- Gå inn på egenskapene for WI-FI og trykk på «Internet Protocol versjon 4 (TCP/IPv4)»



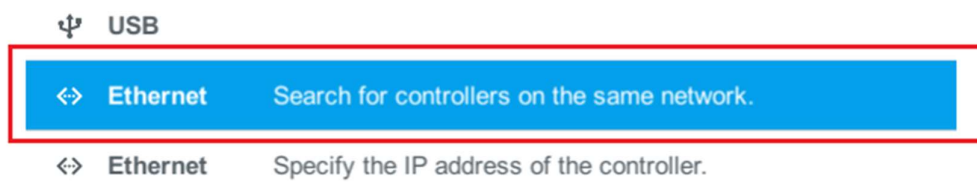
- Bruk følgende IP-adresse (OBS: Sett tilbake til «Motta IP-adresse automatisk» når en er ferdig med roboten og skal på internett igjen)



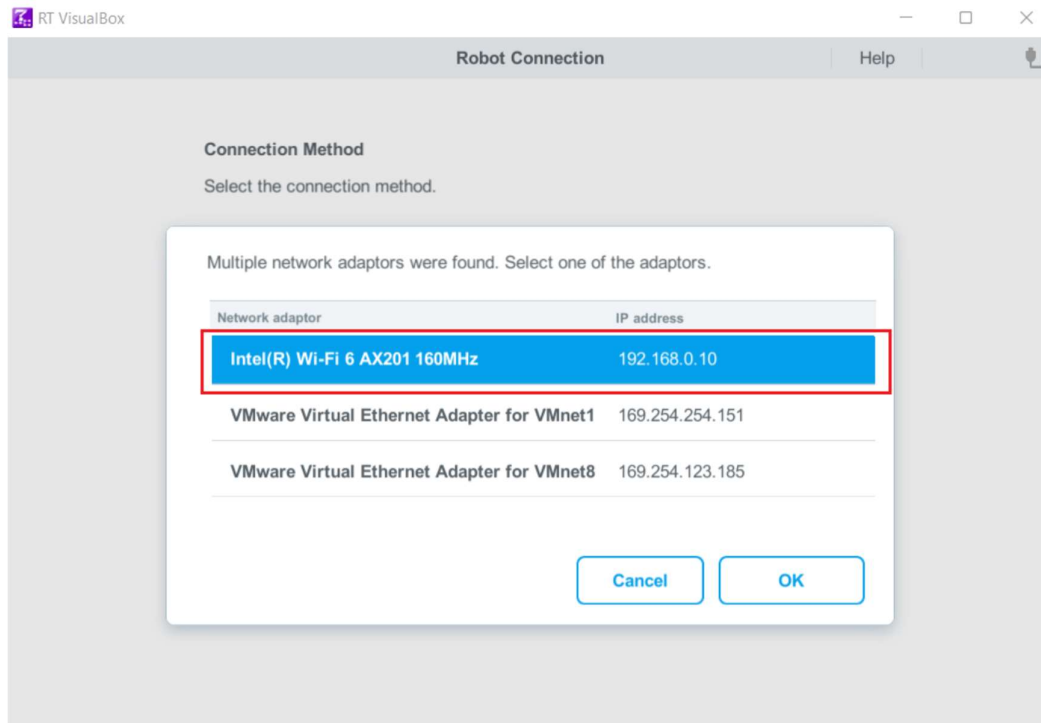
7. Gå inn på RT VisualBox, velg «Search for controllers on the same network» og trykk «next»

Connection Method

Select the connection method.

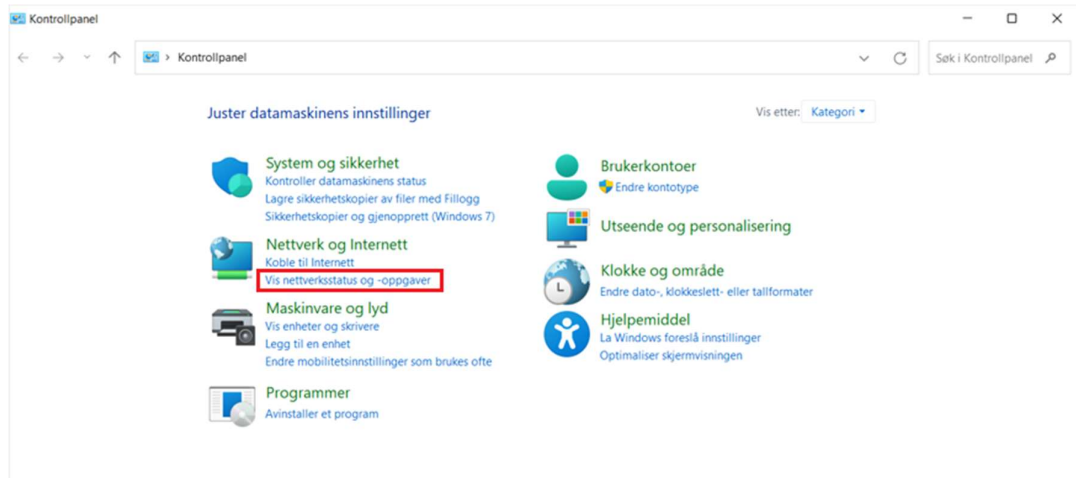


8. Velg riktig nettverk og trykk «OK» så «Next». Roboten skal nå være tilkoblet.

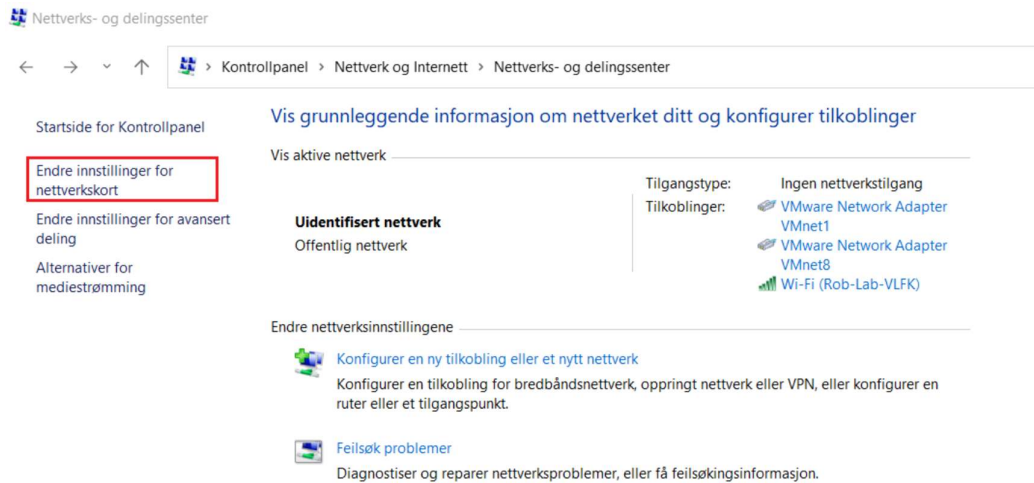


Tilkoblingsguide direkte med LAN kabel:

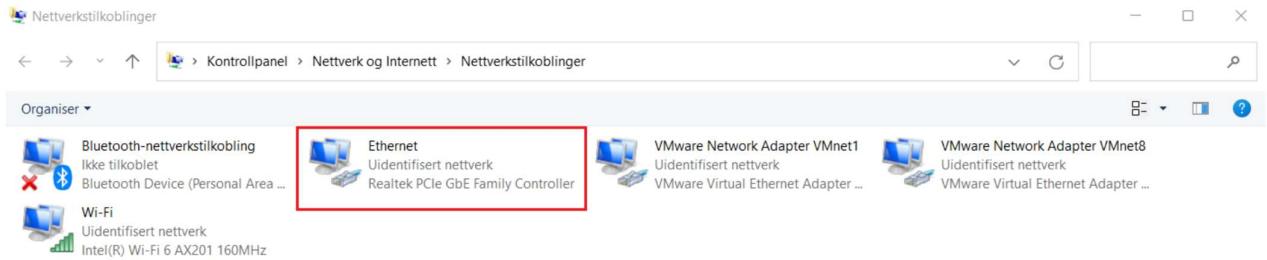
1. Koble LAN kabel fra roboten til en av LAN-portene i din PC.
2. Søk på kontrollpanel på din PC og trykk på «Vis nettverksstatus og -oppgaver»



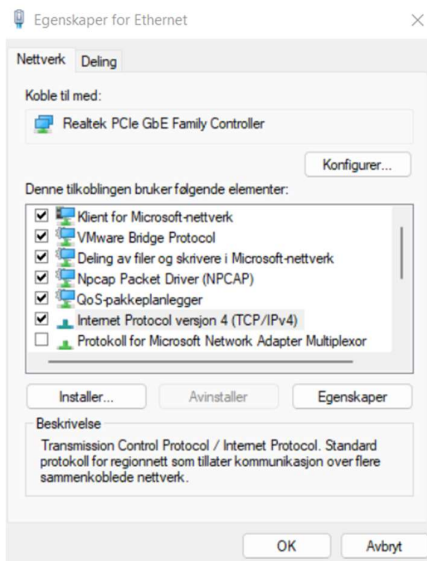
3. trykk på «Endre innstilling for nettverkskort»



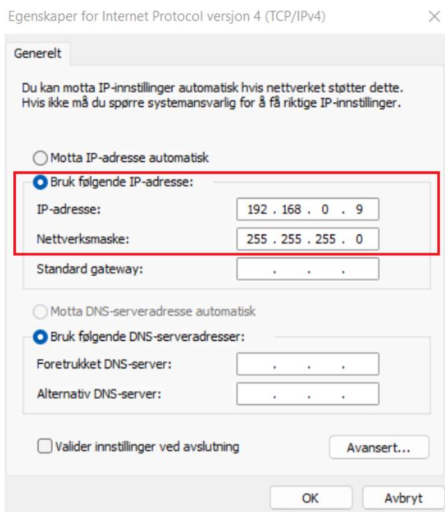
4. Sjekk at det er opprettet kontakt og gå inn på egenskapene til Ethernet.



5. Gå inn på egenskapene for ethernet trykk på «Internet Protocol versjon 4 (TCP/IPv4)»



6. Bruk følgende IP-adresse (OBS: Sett tilbake til «Motta IP-adresse automatisk» når en er ferdig med roboten og skal på internett igjen)



7. Gå inn på RT VisualBox, velg «Search for controllers on the same network» og trykk «next»

Connection Method

Select the connection method.

ψ USB

↔ Ethernet	Search for controllers on the same network.
↔ Ethernet	Specify the IP address of the controller.

8. Velg riktig nettverk og trykk «OK» så «Next». Roboten skal nå være tilkoblet.

Multiple network adaptors were found. Select one of the adaptors.

Network adaptor	IP address
Realtek PCIe GbE Family Controller	192.168.0.9
VMware Virtual Ethernet Adapter for VMnet1	169.254.254.151
VMware Virtual Ethernet Adapter for VMnet8	169.254.123.185

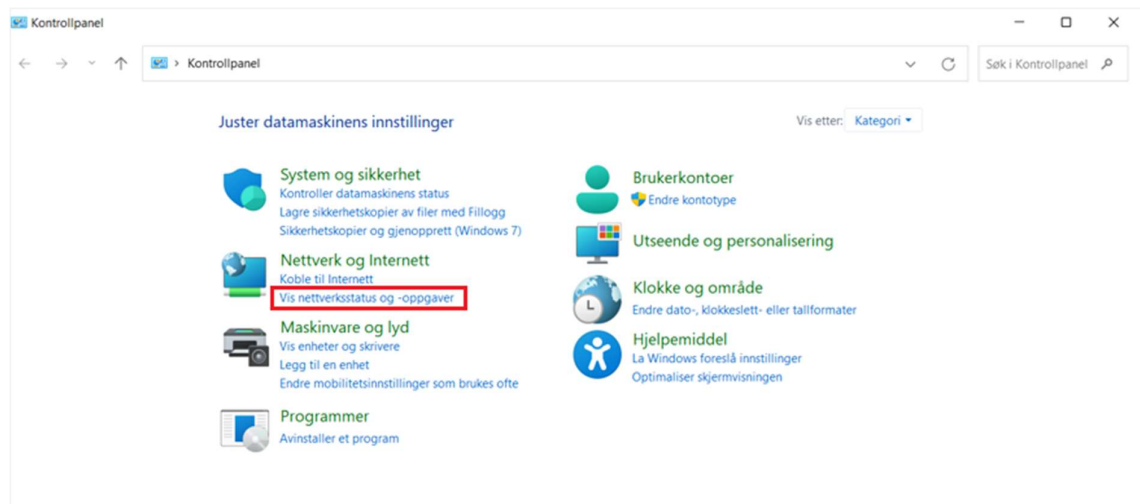
Cancel

OK

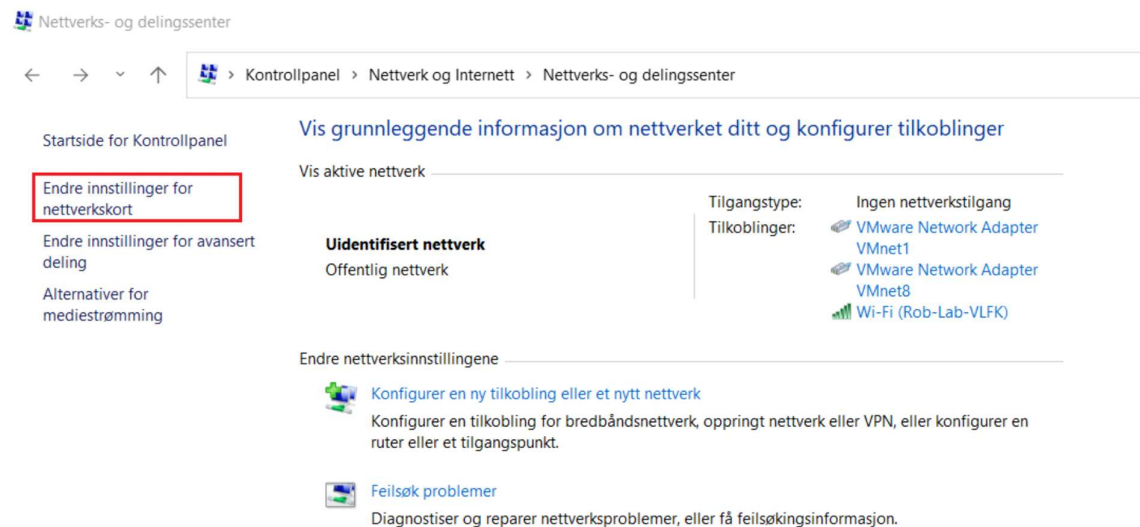
B.2 Tilkoblingsguide RT Toolbox3

Tilkoblingsguide direkte med LAN kabel:

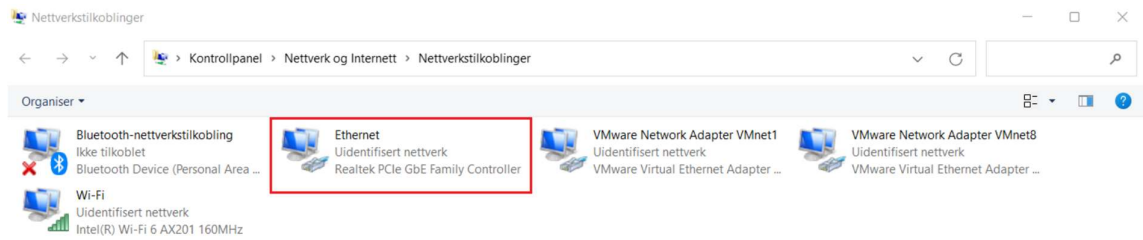
1. Koble LAN kabel fra roboten til en av LAN-portene i din PC.
2. Søk på kontrollpanel på din PC og trykk på «Vis nettverksstatus og -oppgaver»



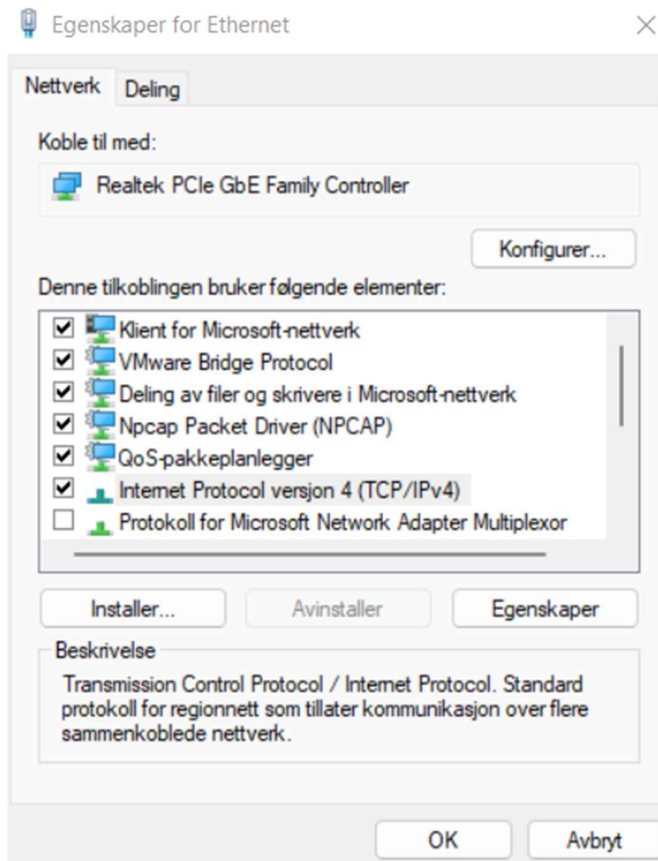
3. trykk på «Endre innstilling for nettverkskort»



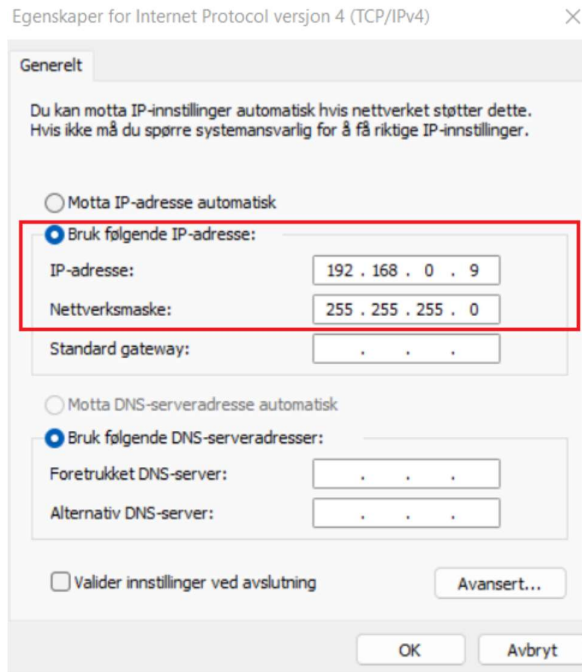
4. Sjekk at det er opprettet kontakt og gå inn på egenskapene til Ethernet.



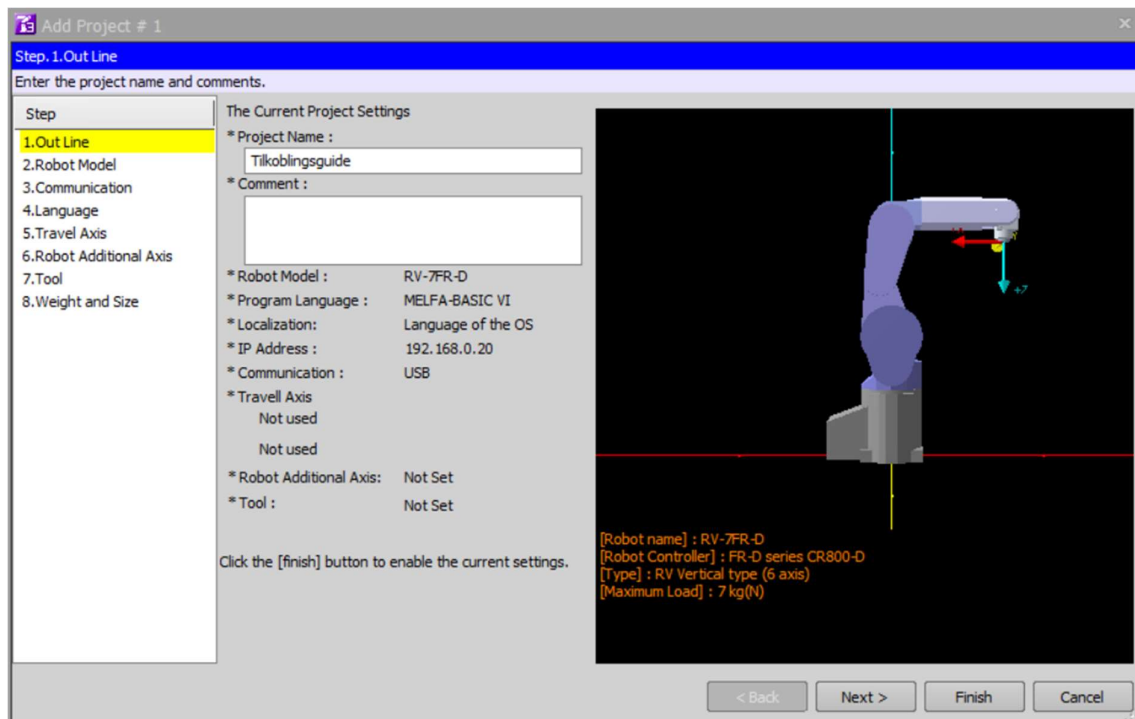
5. Gå inn på egenskapene for ethernet trykk på «Internet Protocol versjon 4 (TCP/IPv4)»



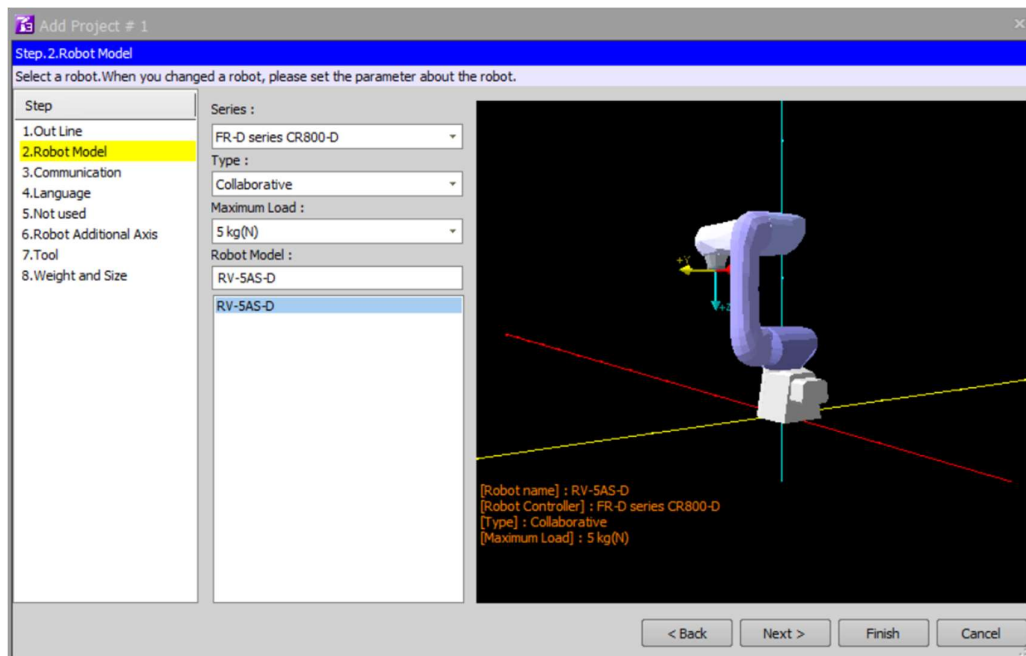
6. Bruk følgende IP-adresse (OBS: Sett tilbake til «Motta IP-adresse automatisk» når en er ferdig med roboten og skal på internett igjen)



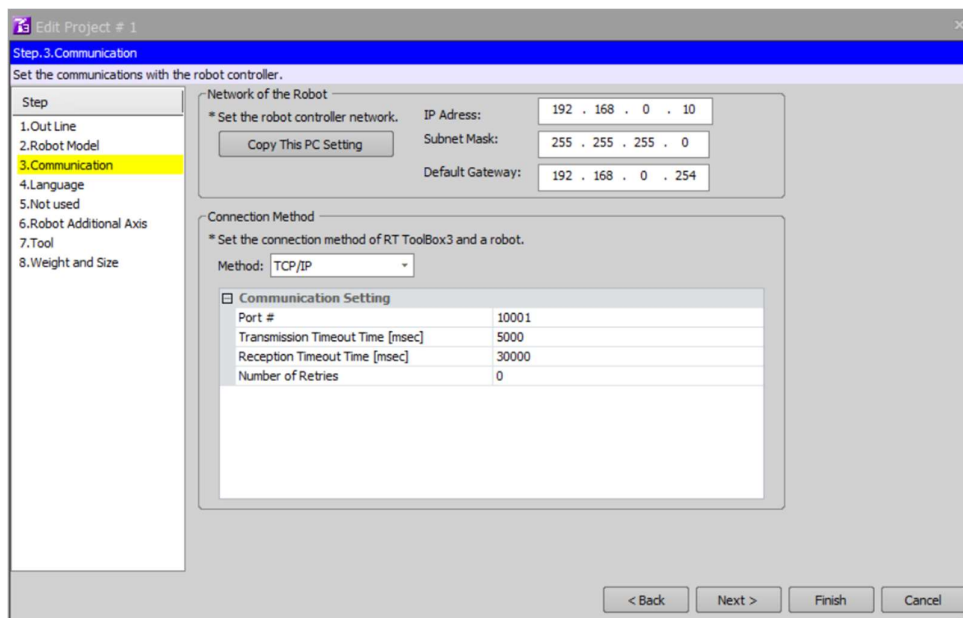
7. Gå inn på RT ToolBox3, trykk på «New». Lagre programmet på ønsket sted og skriv inn prosjektnavn.



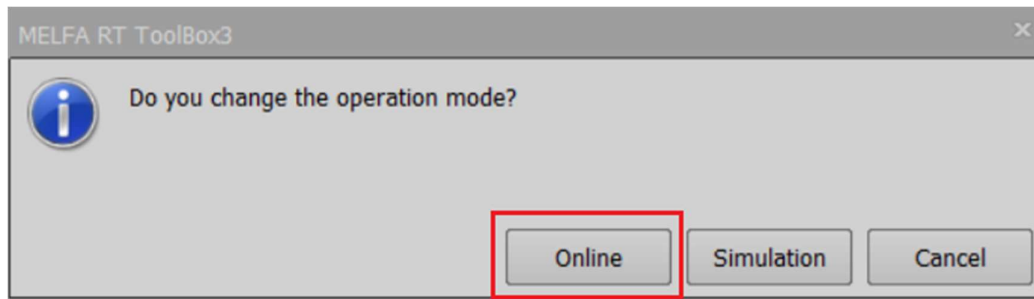
8. Velg følgende robot modell.



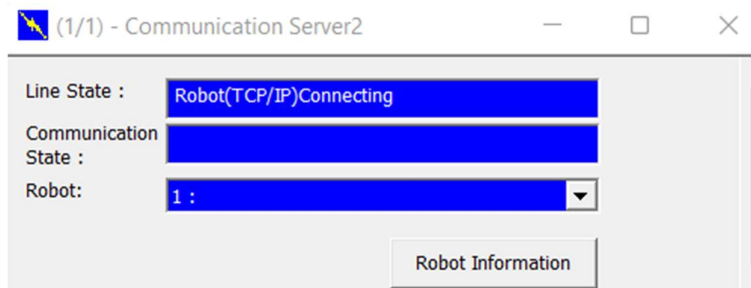
9. Gå videre til «communication». IP adressen til roboten skal være: 192.68.0.10 (OBS: Dersom denne er endret kan en gå inn på RT VisualBox og finne IP adressen der).
Velg TCP/IP som metode.



10. Forsett med «step» 4-8 om ønskelig eller trykk «Finish», deretter «online».

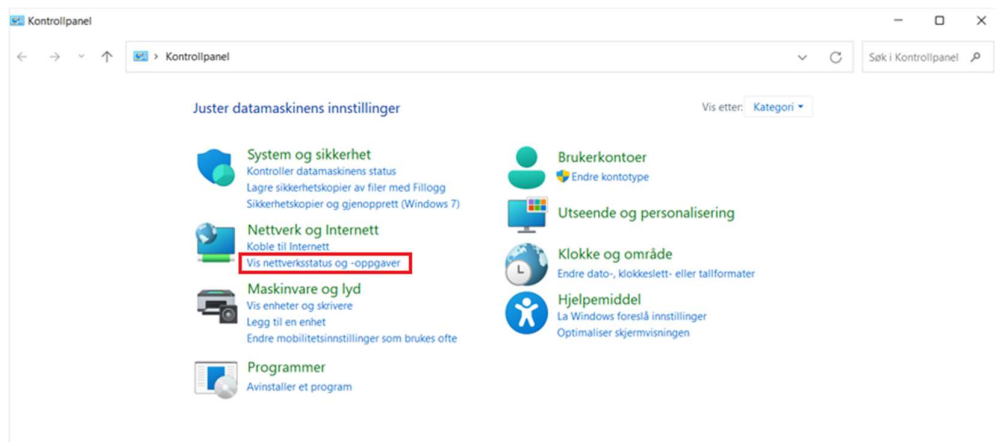


11. Kontakt med roboten skal nå være opprettet og vinduet «Communicate Server» skal ha bakgrunnsfarge blå.

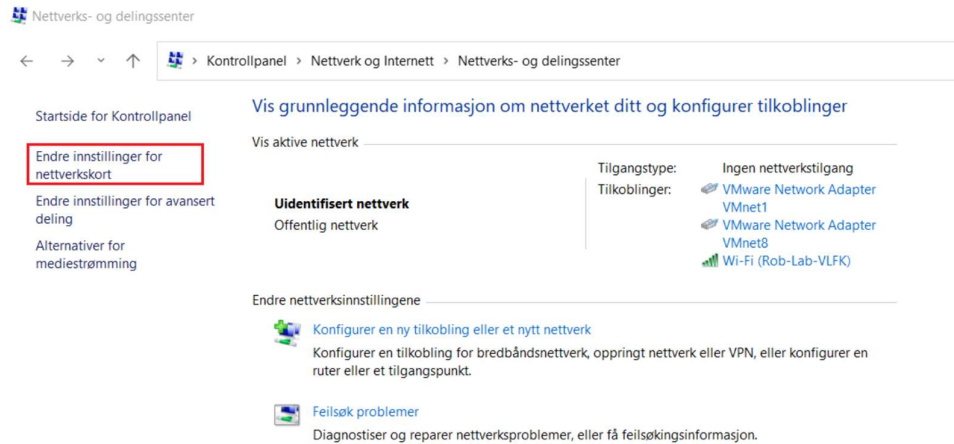


Tilkoblingsguide med router:

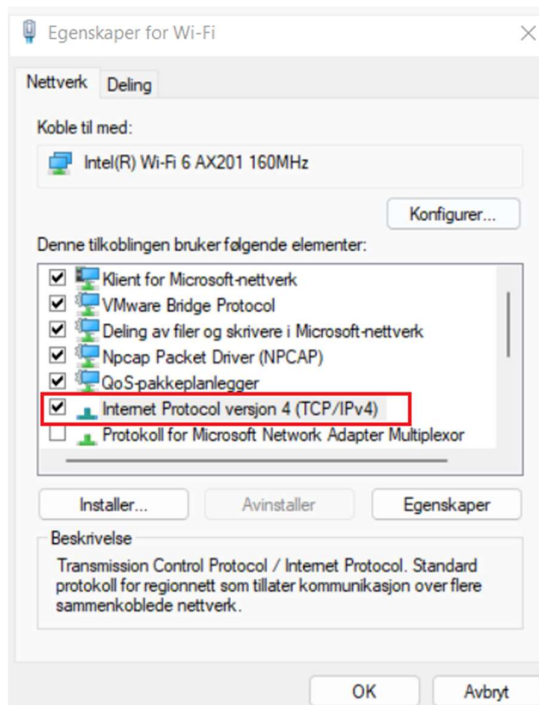
1. Sett i router og koble LAN kabel fra roboten til en av LAN-portene i router.
2. Logg inn på routeren for å konfigurere subnett masken til å være 255.255.255.0
3. Søk på kontrollpanel på din PC og trykk på «Vis nettverksstatus og -oppgaver»



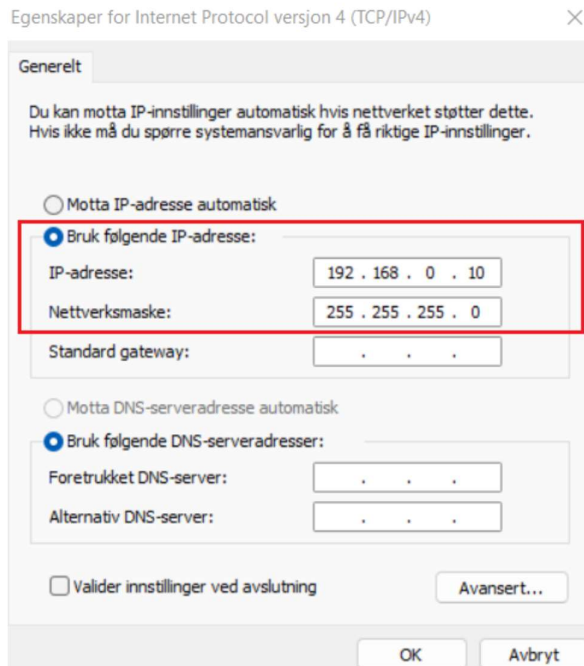
4. trykk på «Endre innstilling for nettverkskort»



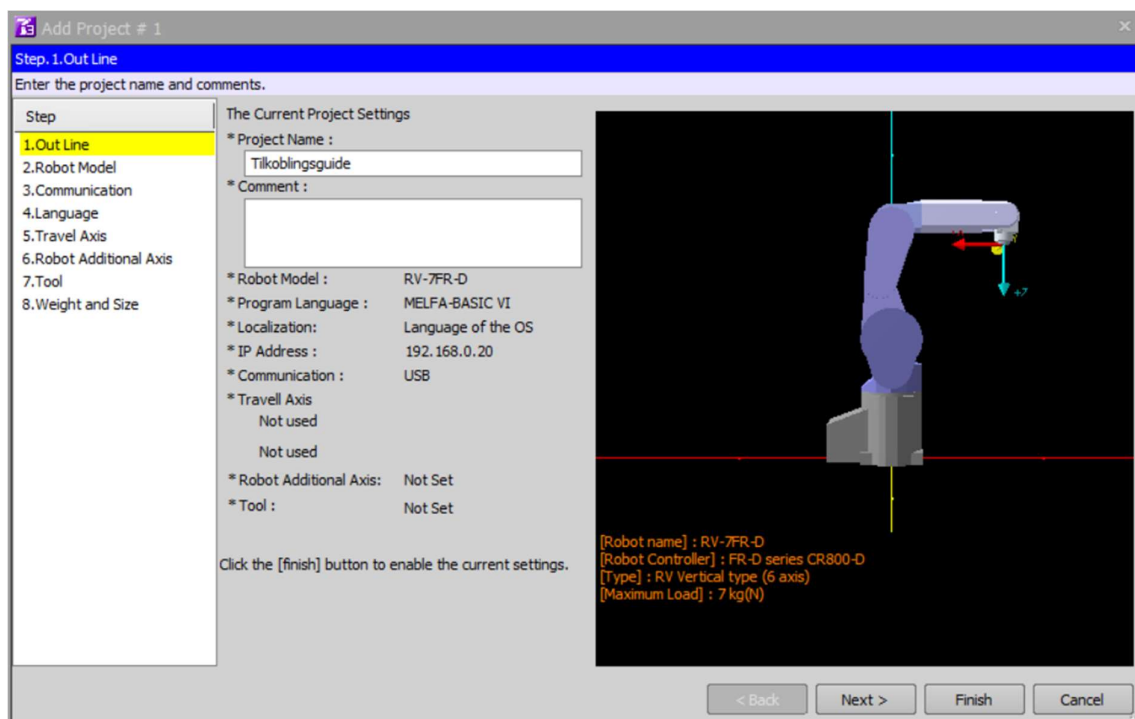
5. Gå inn på egenskapene for WI-FI og trykk på «Internet Protocol versjon 4 (TCP/IPv4)»



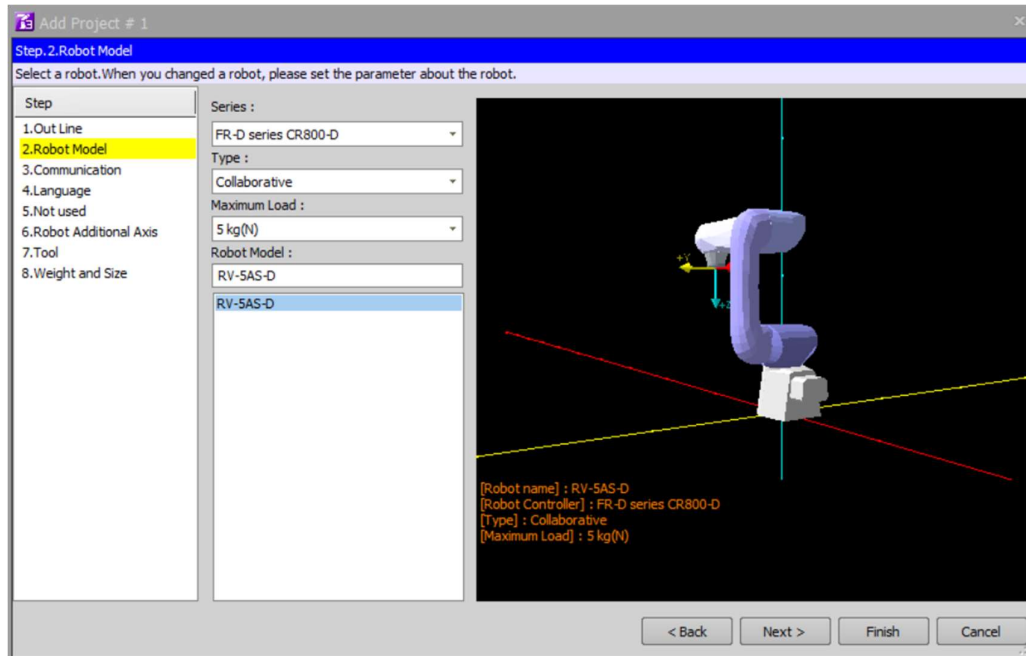
6. Bruk følgende IP-adresse (OBS: Sett tilbake til «Motta IP-adresse automatisk» når er ferdig med roboten og skal på internett igjen)



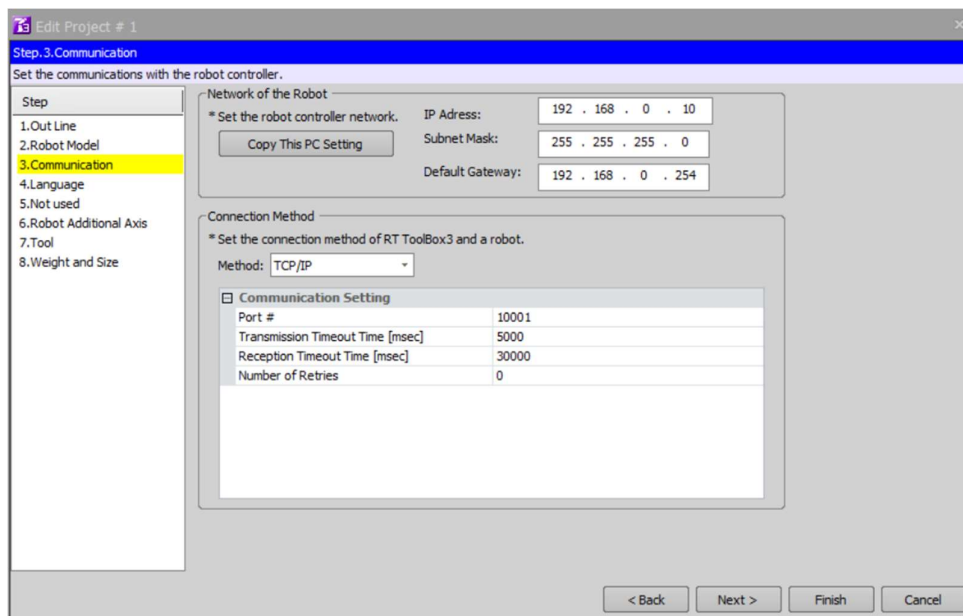
7. Gå inn på RT ToolBox3, trykk på «New». Lagre programmet på ønsket sted og skriv inn prosjektnavn.



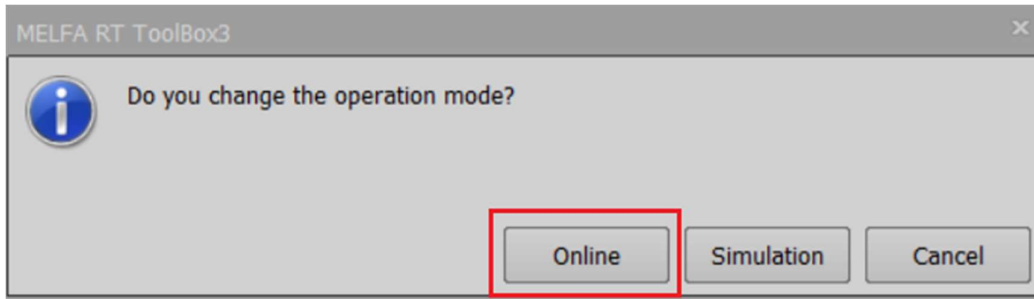
8. Velg følgende robot modell.



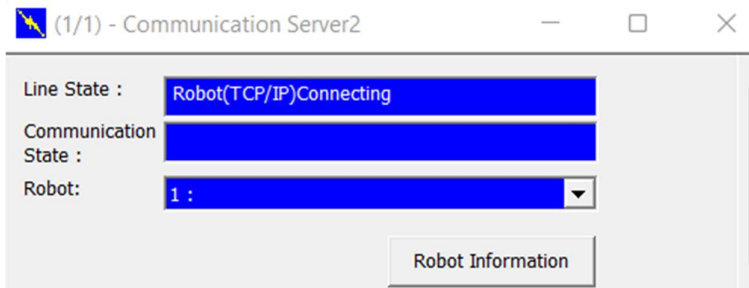
9. Gå videre til «communication». IP adressen til roboten skal være: 192.68.0.10 (OBS: Dersom denne er endret kan en gå inn på RT VisualBox og finne IP-adressen der).
Velg TCP/IP som metode.



10. Forsett med «step» 4-8 om ønskelig eller trykk «Finish», deretter «online».

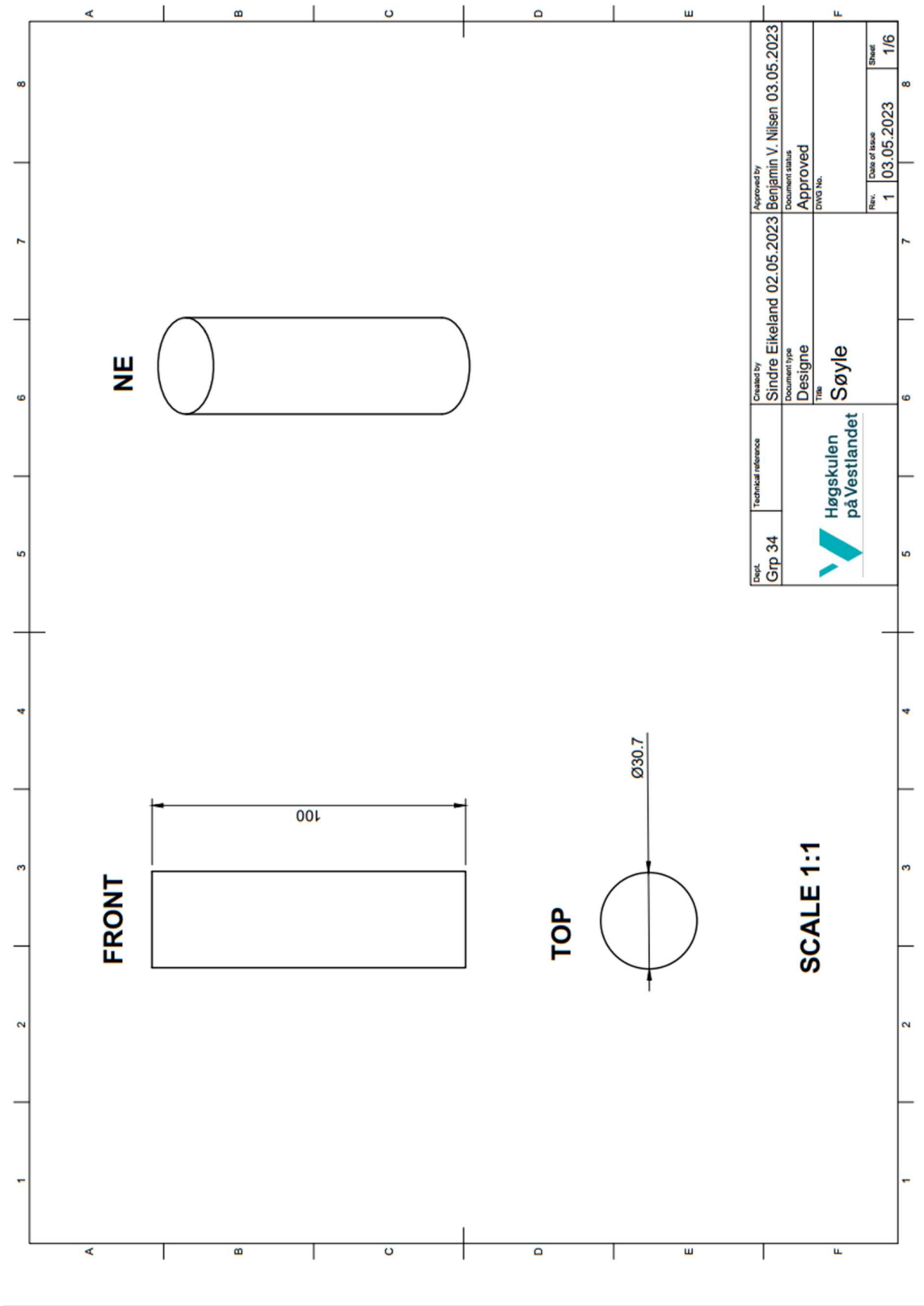


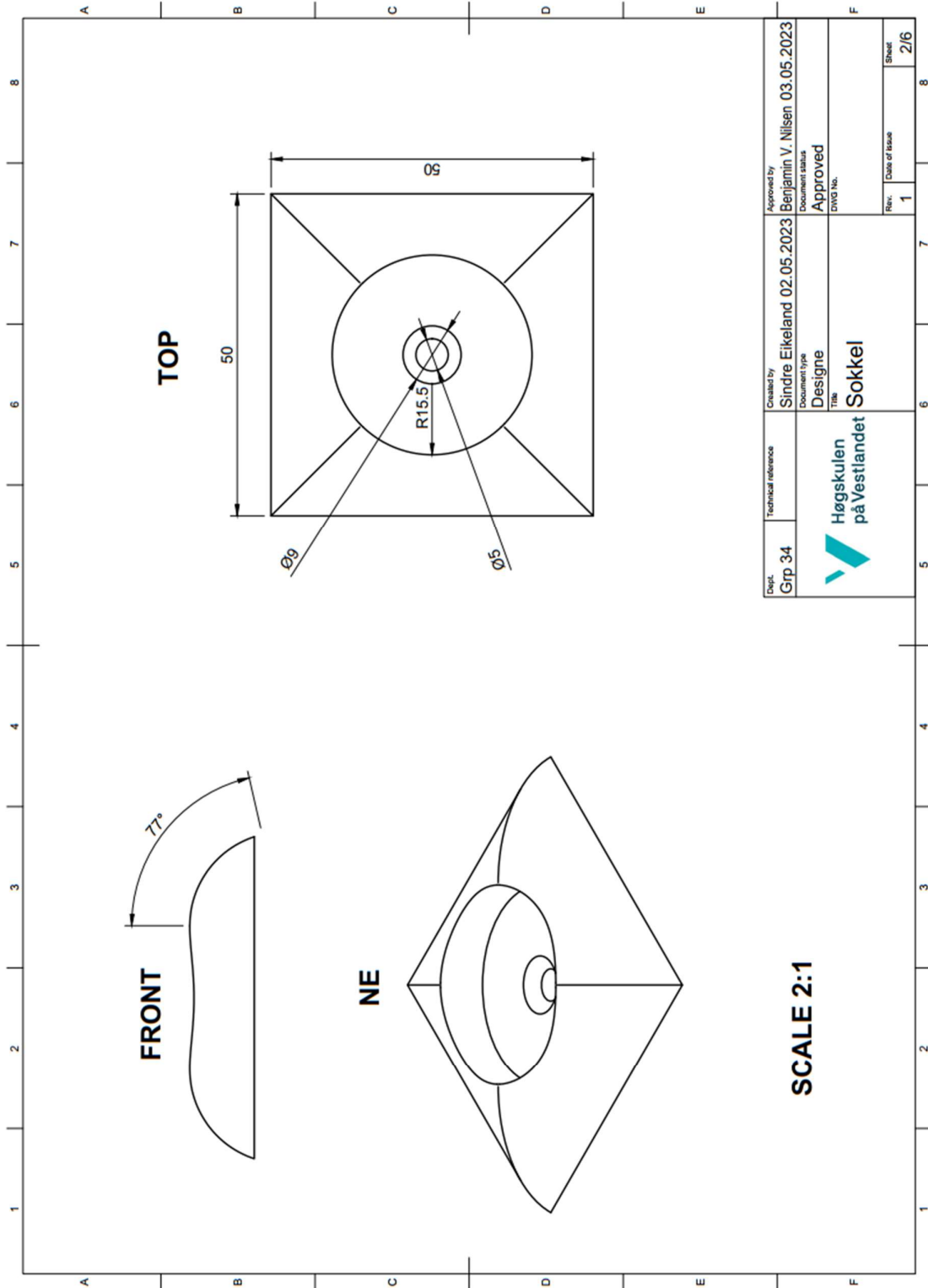
11. Kontakt med roboten skal nå være opprettet og vinduet «Communicate Server» skal ha bakgrunnsfarge blå.



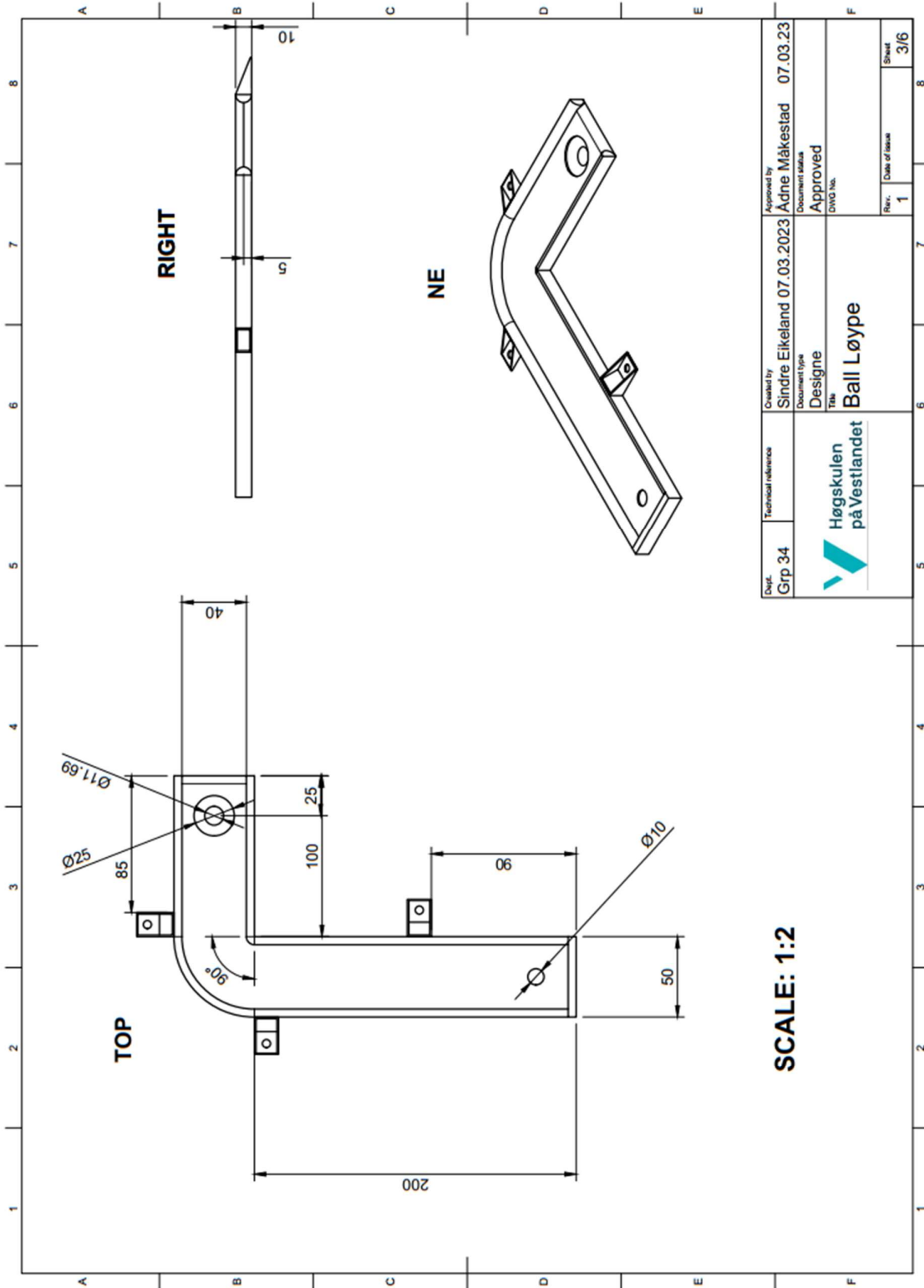
Appendiks C Tegninger

Tegninger av komponenter og modul som er laget i fusion 360, det er ikke tatt noe ut for løb-oppgave 3, da denne aldri ble laget fysisk å endelig transportband ikke var kjent.



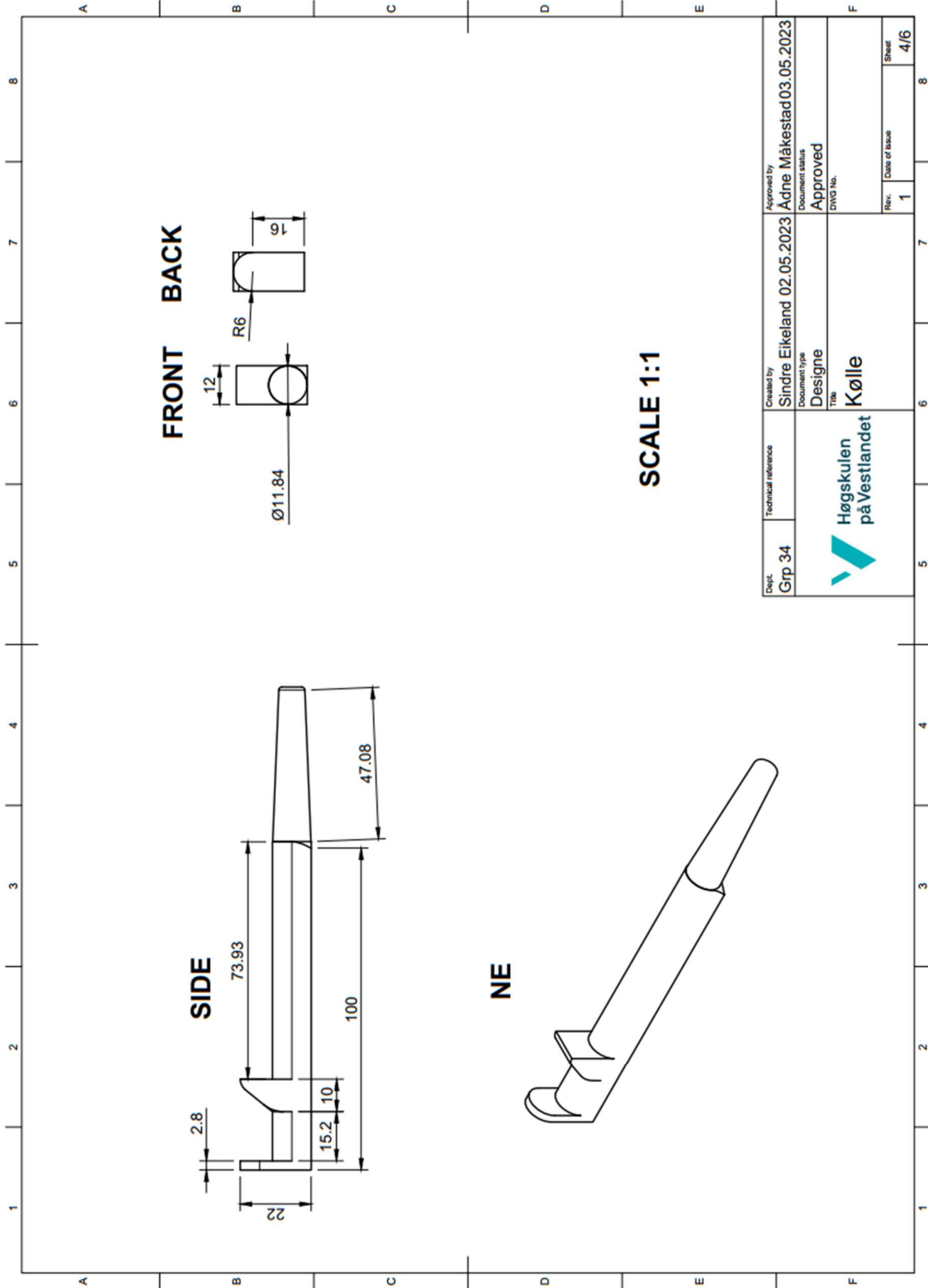


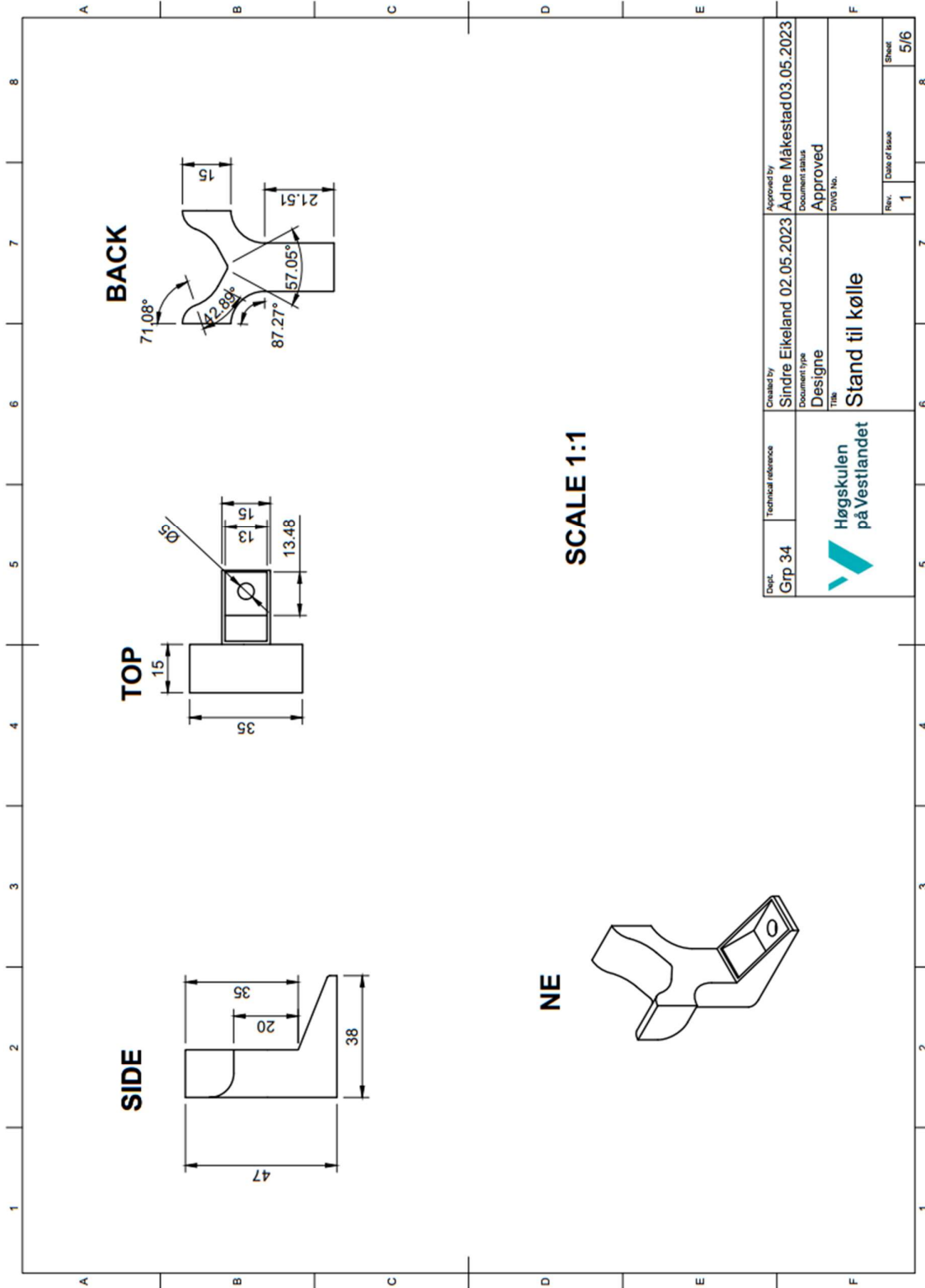
Dept. Grp. 34	Technical reference	Created by Sindre Eikeland 02.05.2023	Approved by Benjamin V. Nilsen 03.05.2023
		Document type Design	Document status Approved
		Title Sokkel	DWG No.
			Rev. 1
			Date of issue
			Sheet 2/6

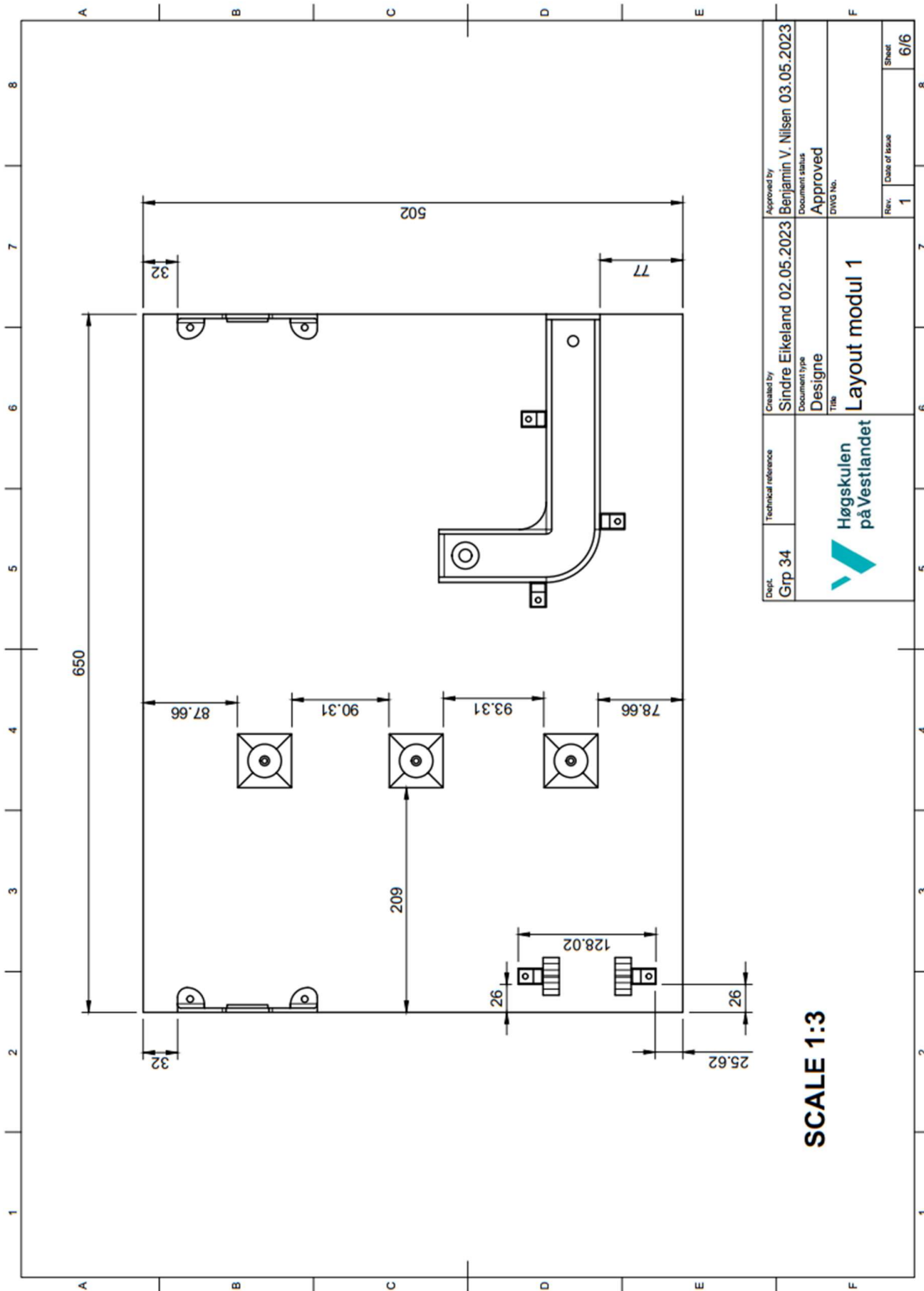


SCALE: 1:2

Dept. Grp 34	Technical reference Høgskulen på Vestlandet	Created by Sindre Eikeland 07.03.2023	Approved by Adne Måkestad 07.03.23
		Document type Designe	Document status Approved
Ball Løype		Rev. 1	Date of issue 3/6







Revisjon	Technical reference	Created by	Approved by
Girp 34	Høgskulen på Vestlandet	Sindre Eikeland 02.05.2023	Benjamin V. Nilsen 03.05.2023
		Document type	Document status
		Design	Approved
		Title	DWG No.
		Layout modul 1	
		Rev:	Date of Issue
		1	1
		Sheet	6/6

Appendiks D Oppgavesett og veiledning til underviser

D.1 Oppgavesett



Emne: (Utfylles etter behov)

Lab 1 - Slalåm

Gruppenummer:
Studenter:
Godkjent:

Tema:

- Waypoints
- Bevegelse
- Bruk av griper
- Loop
- Lineær/Ulineær

Utstyr:

- Mitsubishi Cobot RV-5AS-D
- PC
- Co-act EGP-C griper
- Modul 1 med tre søyler og en kølle (se vedlegg under)

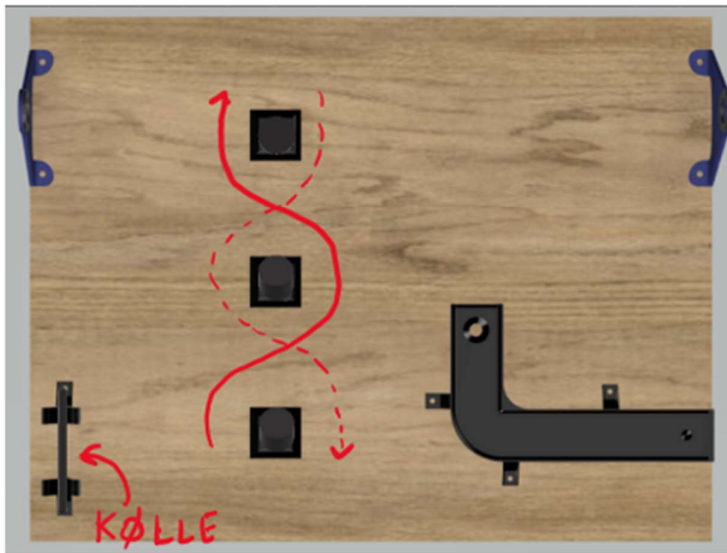
Forarbeid:

- Se veiledningsvideo

Laget av: Sindre Eikeland, Ådne Måkestad og Benjamin Våge Nilsen

Oppgave:

Plukk opp køllen med griperen og led den slalåm gjennom søylene med tuppen først, se tegningen under. En skal lede køllen tre ganger gjennom løypen fram og tilbake før en videre setter køllen på plass igjen. Dette er for å trene på å loope bevegelsen til roboten. En kan enten bruke «direct teach» funksjonen på roboten og lære den punkter for å lage løypen den skal bevege seg i, eller bruke funksjonen «Operate Robot» (dette som er anbefalt å gjøre) i programmet RT VisualBox.



Figur 1 - Oversikt over oppgaven

Spørsmål:

1. Hva er fordeler/ulempene med å bruke en samarbeidende robot (Cobot) kontra en industriell robot?
2. Hva er forskjellen på «non-straight» og «straight» bevegelse?

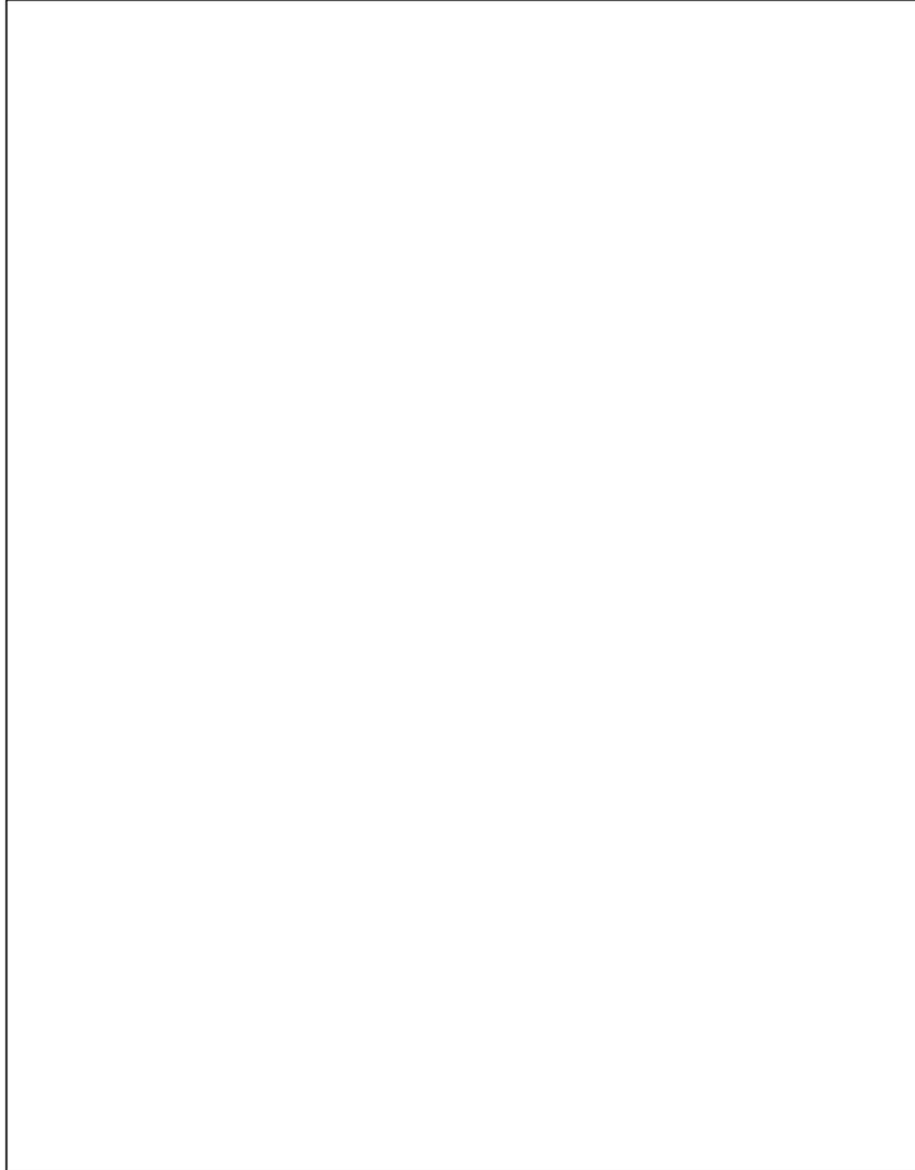
Spørsmål mot RT ToolBox 3

3. Hvordan vil du forklare funksjonen til SP-Line?
4. Beskriv kort fordelene/ulempene med RT ToolBox 3 mot RT VisualBox?

Laget av: Sindre Eikeland, Ådne Måkestad og Benjamin Våge Nilsen

Dokumentasjon:

Programutskrift av RT VisualBox eller RT ToolBox3, samt punktliste:



Laget av: Sindre Eikeland, Ådne Måkestad og Benjamin Våge Nilsen

Lab 2 - Hole in One

Gruppenummer:
Studenter:
Godkjent:

Tema:

- Sikkerhet
- Sikkerhets lysgardiner
- Safety IO
- Nøyaktighet
- Presisjon

Utstyr:

- Mitsubishi Cobot RV-5AS-D
- Co-act EGP-C griper
- PC
- Modul 1 med golfbane, golfball og kølle
- Lysgardiner

Forarbeid:

- Gjennomført lab-oppgave 1
- Se veiledningsvideo

Laget av: Sindre Eikeland, Ådne Måkestad og Benjamin Våge Nilsen

Oppgave:

Først skal lysgardinen monteres og programmeres, gardinene er koblet mot sikkerhetsinngang 4. Legg inn at armen skal stoppe eller sakke ned ved brudd på barrieren.

Programmet skal fungerer på følgende måte: Armen skal plukke opp køllen fra sin startposisjon med griperen og ved presisjon klare å støte golfballen ned i hullet på golfbanen. Videre skal en sette på plass igjen køllen og programmet skal være ferdig.



Figur 1 - Oversikt over oppgave

Laget av: Sindre Eikeland, Ådne Måkestad og Benjamin Våge Nilsen

Sikkerhet:

14.9.5. Safety Logic

Set parameters for safety logic.

14.9.5.1. Safety Input Parameter(Safety IO)

Set parameters related to the input signal of the safety function.

Double-click [Safety parameter] -> [Safety option] -> [Safety Logic] -> [Safety Input(Safety IO)] under the [Parameter] item to be edited from the project tree.

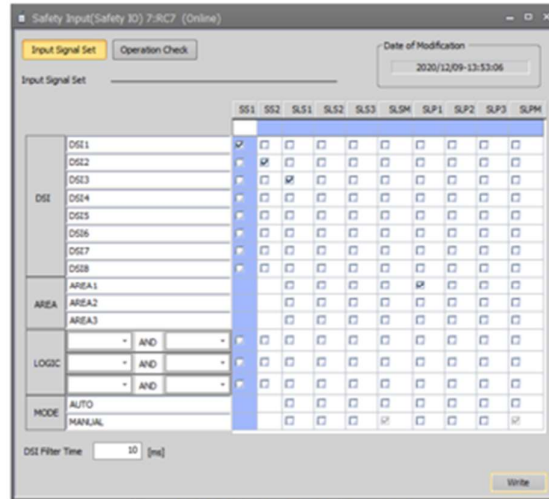


Figure 14-84 Safety Input(Safety IO) Parameter

Å forstå rekkevidden til en 6-akset-robot i bevegelse kan være vanskelig, noe som gjør det farlig å nærme seg roboten. Å utføre utstyrsvedlikehold i nærheten av en robot er risikabelt. For å minimere denne risikoen, er det nødvendig med flere sikkerhetstiltak.

Hvis en blokkerer en sikkerhets lysgardin, kan en sette roboten til total stopp eller senke farten til roboten. Sikkerhetslaserskannere kan også brukes til å oppdage tilstedeværelsen av arbeidere innenfor sikkerhetssonen.

I slike situasjoner vil robotdriften gå som normalt etter at arbeideren er utenfor sonen, når lysgardinen ikke lenger er blokkert.

Sikkerhets lysgardiner er designet for å beskytte personell mot skader, oftest i produksjonsmiljøer. I tillegg til å beskytte personell, kan sikkerhets lysgardiner også beskytte maskiner mot skader.

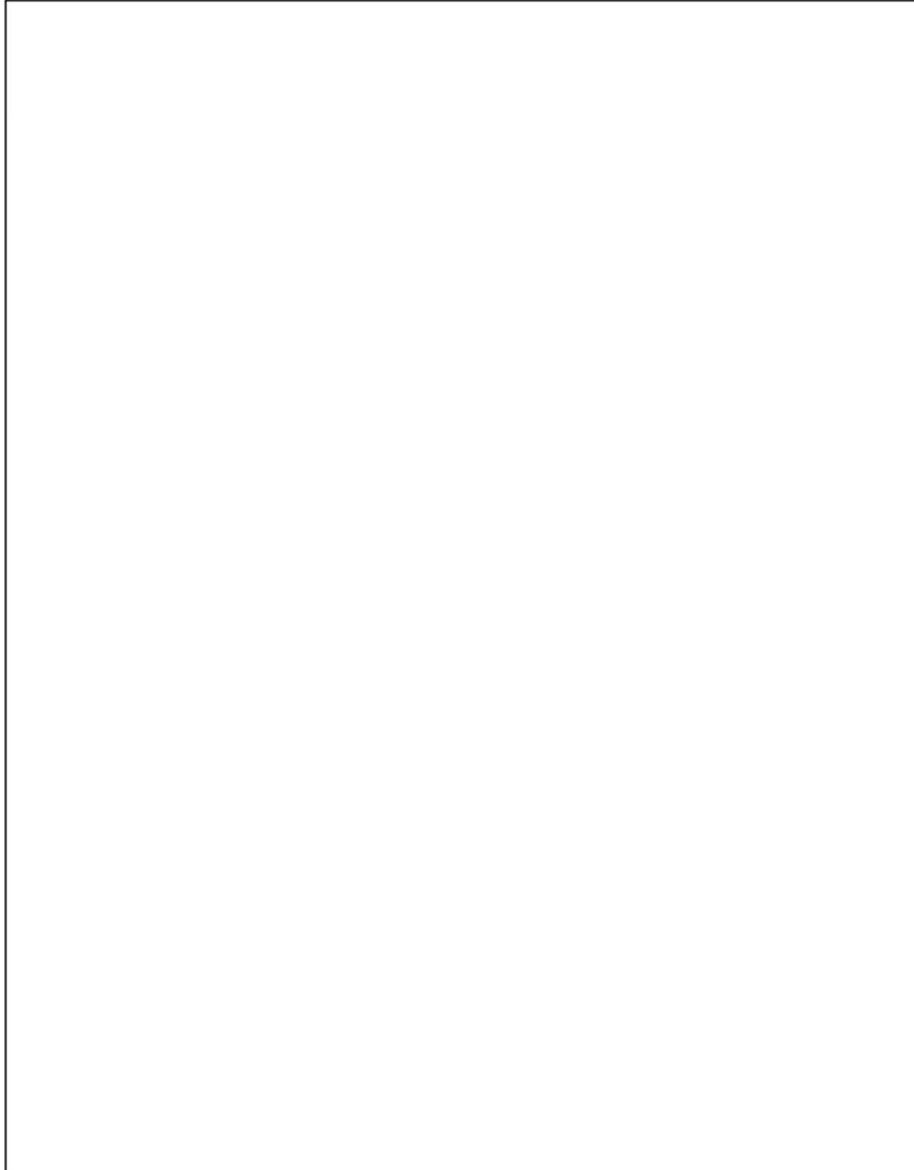
Laget av: Sindre Eikeland, Ådne Måkestad og Benjamin Våge Nilsen

Spørsmål:

1. Beskriv forskjellen mellom sikkerhets IO og vanlig IO, begrunn svaret.
2. Utdyp hva TCP er viktig for i programmeringen av en robotarm.
3. Hva er de forskjellige hastighetene i highspeed mode, standard mode og lowspeed mode, og hva er spesielt viktig å tenke på når den er i highspeed mode?

Dokumentasjon:

Programutskrift av RT VisualBox eller RT ToolBox3, samt punktliste:



Laget av: Sindre Eikeland, Ådne Måkestad og Benjamin Våge Nilsen

Lab 3 - Pallesortering

Gruppenummer:
Studenter:
Godkjent:

Tema:

- Pallesortering
- Transportbånd
- Loop
- Sikkerhet

Utstyr:

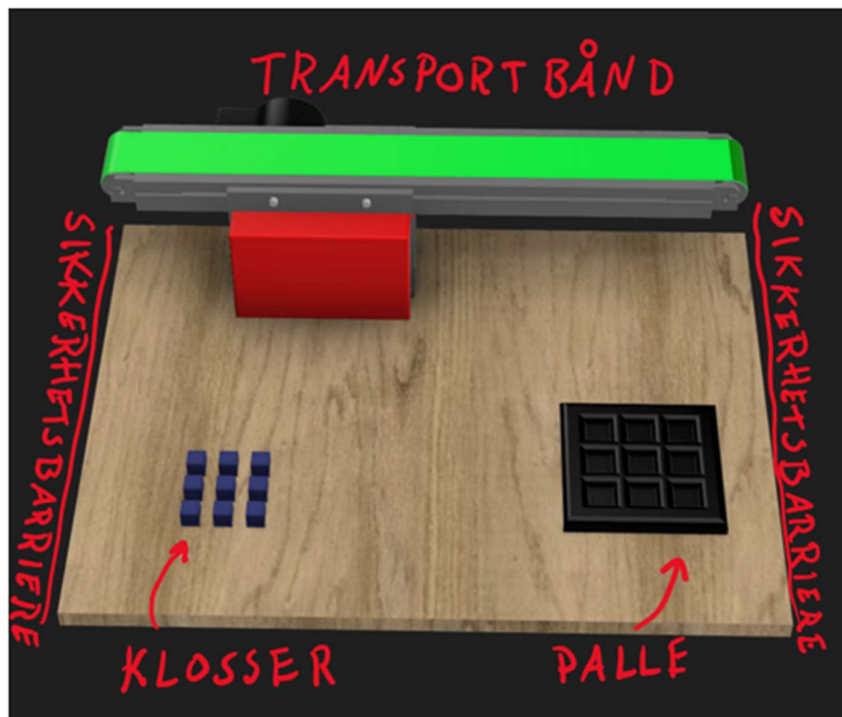
- Mitsubishi Cobot RV-5AS-D
- PC
- Co-act EGP-C griper
- Modul 2 med klosser, palle og transportbånd

Forarbeid:

- Gjennomført lab-oppgave 2

Oppgave:

Sett opp sikkerhetsbarrierer på begge sidene av modulen som roboten ikke skal krysse. Deretter skal en mate klosser manuelt på transportbåndet (mulig å lage en ordning som mater automatisk) og disse klossene skal bli detektert av en sensor som sender et signal til armen. Klossene skal videre bli plukket opp ved hjelp av griperen og plassert i pallen. Hvis en ikke har et transportbånd kan klossene mates manuelt. Videre skal roboten plassere klossene til original startposisjon når en er ferdig med å sortere alle klossene (se bilde under).

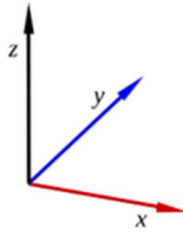


Figur 1 - Oversikt over oppgaven

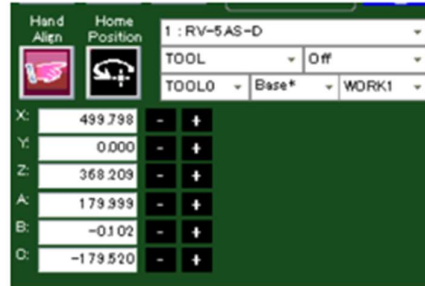
Laget av: Sindre Eikeland, Ådne Måkestad og Benjamin Våge Nilsen

Spørsmål:

1. Angi på XYZ aksen hvilken av aksene som er roll, pitch og yaw (I «operation panel» er det definert A=roll, B=Pitch og C=Yaw) angi også positiv retning med pil (hint, høyrehåndregelen)



Figur 3 - Avbildning av XYZ akse

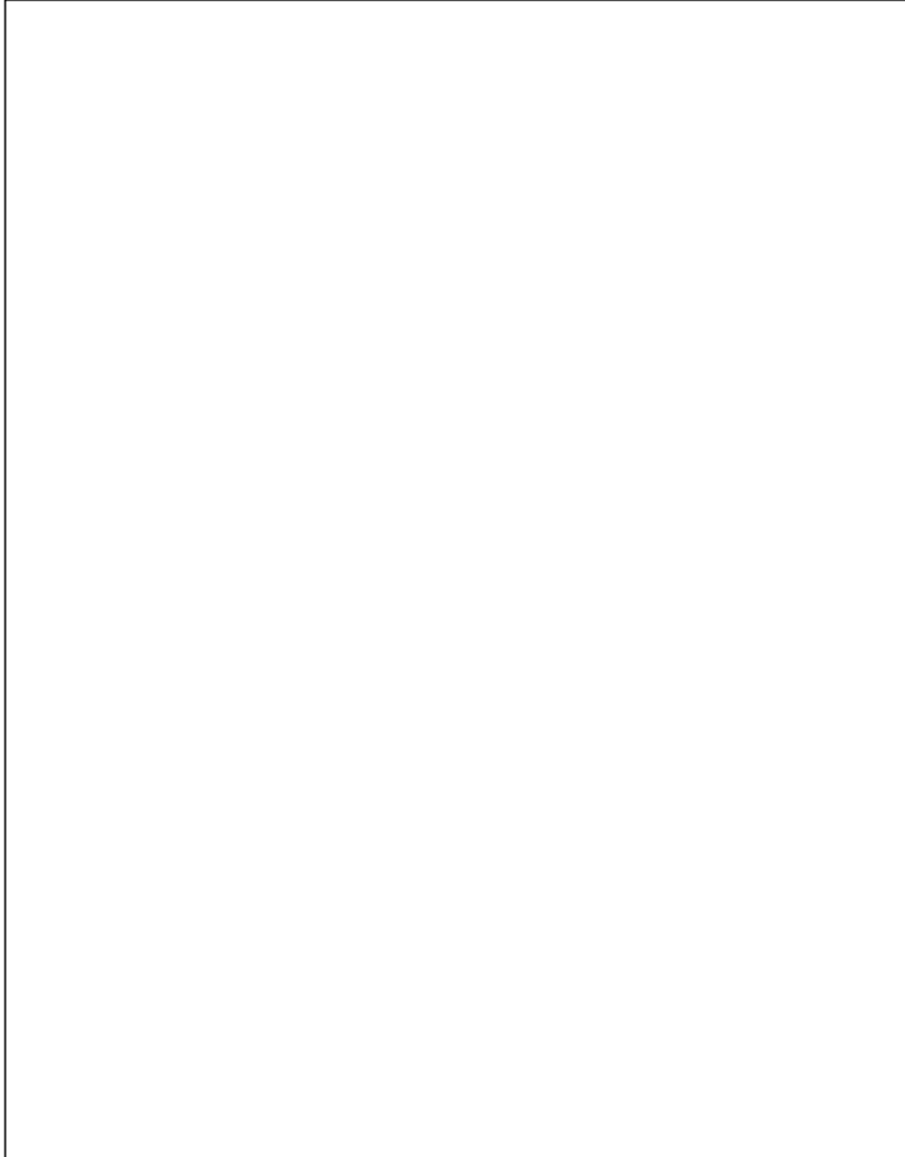


Figur 2 - Utklipp av Toolbox3 operasjons panel

2. Hvilket nivå av samarbeid vil du plassere denne oppgaven mot med tanke på IFR sine beskrivelser av samarbeid mellom industrielle roboter og menneskelige arbeidere (sameksistens, sekvensielt samarbeid, samarbeid og responsivt samarbeid). Begrunn svaret.

Dokumentasjon:

Programutskrift av RT VisualBox eller RT ToolBox3, samt punktliste:



Laget av: Sindre Eikeland, Ådne Måkestad og Benjamin Våge Nilsen

D.2 Veiledning til underviser

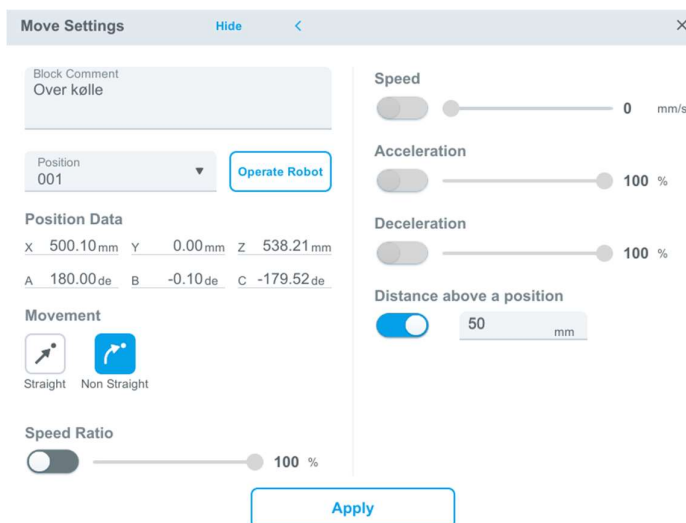
Veiledning til underviser for MELFA ASSIST Lab 1

I denne lab-oppgaven skal 2-3 studenter jobbe sammen i grupper for å løse oppgaven. Oppgavens mål er at studentene skal lage et program som skal plukke opp køllen og fører den gjennom løypen tre ganger uten å velte kjeglene, for så å legge køllen tilbake (se oppgave tekst).

Løsning med RT VisualBox:

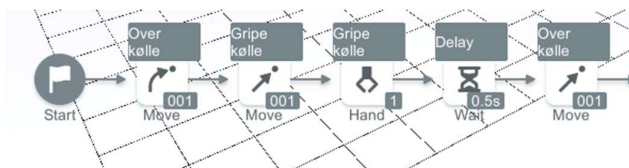
Når oppgaven skal løses i VisualBox er studentene avhengig å jobbe direkte mot roboten da det ikke er mulig å simulere dette i VisualBox. Her kan det være lurt at studentene får gjennomgang av programvaren og får mulighet å teste den ut selv før de gjennomfører lab-oppgaven.

Når programmet skal plukke opp køllen anbefales det at punktet der køllen ligger blir definert, det er mulig å lage en bevegelse til en gitt distanse over dette punktet (se bilde). I denne oppgaven kan det godt være mellom 50-100mm (dette vil avhenge av hvilken



situasjon en befinner seg i). Bevegelsen til dette punktet har ikke noe å si da roboten opererer i et fritt område, derfor bruker vi en ulineær bevegelse (Non Straight).

Resterende av operasjonen i det å plukke opp køllen kan for eksempel se slik ut:



Her ser vi at punkt 1 blir brukt gjentatte ganger, og at når vi går ned/opp og plukker køllen opp vil vi ha en lineær bevegelse.

NB! Det er viktig å punktere for studentene at når en lager programmet er det veldig lurt å kommentere hver bevegelsene de legger inn.

Videre i programmet burde en ha en bevegelse til starten av løypen for så ha gjennomkjøring av løypen i en loop der man kan definere hvor mange ganger den skal kjøre. Under gjennomkjøring skal tuppen alltid gå først mellom hindrene, her er det viktig å tenke på at den ikke slår ned noe under gjennomkjøring når roboten skal rotere på håndleddet. Dette er en typisk feil studentene kommer til å gjøre, det er her de vil måtte jobbe med feilsøking.

Typiske problemer under denne delen kan være at det ikke har tatt hensyn til hvor mye håndleddet på roboten kan rotere samt når det blir brukt lineær og ulineære bevegelse.

Løsning med RT ToolBox3:

Når oppgaven skal løses i TB3 (RT ToolBox3) kan studenten både løse den direkte mot roboten eller i simulatoren. Anbefalt vil være å la gruppene løse den i simulator for så å koble seg opp mot roboten. Deretter å laste programmene sine over på den, for å kjøre de der.

Når oppgaven løses i TB3 vil det beste resultatet oppnås ved å lage en SP Line, da vil det bli en mer flytende gjennomkjøring av hinderløypen istedenfor at den stopper opp hver gang man skal til et nytt punkt. Når man skal laste opp programmet en lager vil det være behov for å laste både programfil og eventuelle SP Liner man har definert over på roboten (nevnt i video om SP Line). Nederst på denne delen finner man programmet og SP Liner brukt under video for løsningsforslag.

Det vil være ønskelig å følge en lik teknikk for når en skal plukke opp køllen som ble gjort ved programmering i VisualBox. Nederst på siden ligger programkode som demonstrer hvordan det gjøres i TB3.

SP Line:

Det kan vise seg er at det ikke alltid er behov for så mange punkter når man lager en SP Line, men ved høyere antall punkter vil det som regel bli en mer naturlig flyt i bevegelsen, det er allikevel ingen poeng å overdrive her. Under løsningsforslaget ble det brukt ti punkter for SP Linen som gikk frem (02) og syv punkter for den som returnerte (03).

Programkode:

```

Program 1:RC1LAB1.prg [MELF... x
1 Mov p1_Start      'Bevegelse til start posisjon
2 Mov p3_KollePos,-100 '100mm over køllens posisjon
3 Ovrd 10          'Setter hastigheten til 10%
4 Mvs p3_KollePos  'Linjær bevegelse ned til køllen
5 HClose 1        'Lukker griper
6 Dly 0.5         'Pause 0.5 sekunder
7 Mvs p3_KollePos,-100 'Linjær bevegelse opp fra køllen (100mm)
8 Ovrd 100        'Setter hastigheten til 100% igjen
9 Mov p4_StartSlalom 'Beveger seg til starten på hinder løypen
10 For i=0 To 1 Step 1 'Starter For løkke som skal kjøre 2 ganger
11 Ovrd 70         'Setter hastigheten til 70%
12 MvSpl 2,250,50 'Kjører SP Line nr 2, hastighet 250mm/s,akselerasjon 50mm/s^2
13 Ovrd 100        'Setter hastigheten til 100%
14 Mov p5_Midtveis 'Mellom punkt for å rotere håndled tilbake
15 Ovrd 70         'Setter hastigheten til 70%
16 MvSpl 3,250,50 'Kjører SP Line nr 3, hastighet 250mm/s,akselerasjon 50mm/s^2
17 Next           'Slutten på For løkke
18 Ovrd 100        'Setter hastighet til 100%
19 Mov p3_KollePos,-100 'Sekvensen for å legger tilbake køllen<-
20 Mvs p3_KollePos  '----->
21 HOpen 1        '----->
22 Dly 0.5         '----->
23 Mvs p3_KollePos,-100 '----->
24 Mov p1_Start    'Kjører armen tilbake å start posisjon
25 Hlt            'Avslutter programet fulstendeig, uten at det starter på nytt

```

Punkter:

Man kan enkelt bruke punkter man allerede har definer fra en tabell ved å kopiere og lime de inn i punktlisten i et annet program. **NB! Det kan ikke være noen punkter som har likt navn.**

Punkter Lab 1									
p1_Start	7.970	-500.730	326.010	-167.570	-1.420	-178.160			7 0
p3_KollePos	289.730	-412.540	38.470	-178.140	-0.150	-178.420			7 0
p4_StartSlalom	286.980	-410.940	126.370	176.730	4.690	58.950			7 0
p5_Midtveis	46.990	-840.920	40.210	-179.960	0.640	-48.480			7 0

SP Liner:

Tabellene under er kompatibel til å kopieres for så å legges inn i TB3.

NB! Når man skal legge inn data for en ny SP Line med data fra en tabell (som under), kan man kopiere tabellen, men du er nødt til å definere like mange punkter som det er i tabellen du vil legge inn i den ny definerte SP Line'en.

SP 02:

91.190000	-329.320000	51.580000	-176.500000	1.680000	102.000000	0.000000	0.000000	7
	0	0	100	100	100	-1	-1	1
	0	0	0					
32.100000	-354.390000	51.580000	-176.500000	-1.040000	141.120000	0.000000	0.000000	7
	0	0	100	100	100	-1	-1	1
	0	0	0					
-6.460000	-438.820000	51.580000	-176.500000	-2.820000	-149.890000	0.000000	0.000000	7
	0	0	100	100	100	-1	-1	1
	0	0	0					
86.140000	-506.590000	51.580000	-176.500000	-3.310000	-89.510000	0.000000	0.000000	7
	0	0	100	100	100	-1	-1	1
	0	0	0					
177.190000	-507.620000	51.580000	-176.500000	-0.310000	-143.660000	0.000000	0.000000	7
	0	0	100	100	100	-1	-1	1
	0	0	0					
189.490000	-594.190000	51.580000	-176.500000	5.940000	143.030000	0.000000	0.000000	7
	0	0	100	100	100	-1	-1	1
	0	0	0					
81.930000	-648.630000	51.580000	-176.500000	5.670000	91.110000	0.000000	0.000000	7
	0	0	100	100	100	-1	-1	1
	0	0	0					
7.240000	-667.720000	51.580000	-176.500000	0.060000	149.900000	0.000000	0.000000	7
	0	0	100	100	100	-1	-1	1
	0	0	0					
3.400000	-742.900000	51.580000	-176.500000	-5.980000	-139.420000	0.000000	0.000000	7
	0	0	100	100	100	-1	-1	1
	0	0	0					
47.590000	-800.860000	51.580000	-176.500000	-7.500000	-108.680000	0.000000	0.000000	7
	0	0	100	100	100	-1	-1	1
	0	0	0					

SP 03:

47.590000	-800.860000	51.580000	175.350000	-6.850000	-49.260000	0.000000	0.000000	7
	0	0	100	100	100	-1	-1	1
	0	0	0					
187.460007	-700.130005	51.580000	-176.500000	-0.510000	22.280001	0.000000	0.000000	7
	0	0	100	100	100	-1	-1	1
	0	0	0					
137.360001	-651.150024	51.580000	-176.500000	-3.140000	87.370003	0.000000	0.000000	7
	0	0	100	100	100	-1	-1	1
	0	0	0					
20.190001	-627.919983	51.580000	-176.500000	-3.200000	44.150002	0.000000	0.000000	7
	0	0	100	100	100	-1	-1	1
	0	0	0					
-2.350000	-545.770020	51.580000	-176.500000	1.550000	-39.270000	0.000000	0.000000	7
	0	0	100	100	100	-1	-1	1
	0	0	0					
84.870003	-496.730011	51.580000	-176.500000	1.660000	-89.099998	0.000000	0.000000	7
	1048576	0	100	100	100	-1	-1	1
	0	0	0					
169.229996	-457.459991	51.580000	-176.500000	-1.140000	-12.310000	0.000000	0.000000	7
	0	0	100	100	100	-1	-1	1
	0	0	0					

Veiledning til underviser for MELFA ASSISTA Lab 2

I denne lab-oppgaven skal 2-3 studenter jobbe sammen i grupper for å løse oppgaven. Denne oppgaven går ut på å støte en golfball ned i hullet på løypen. Dette skal gjøres ved å plukke opp køllen å bruke den til å støte golfballen (se oppgave tekst).

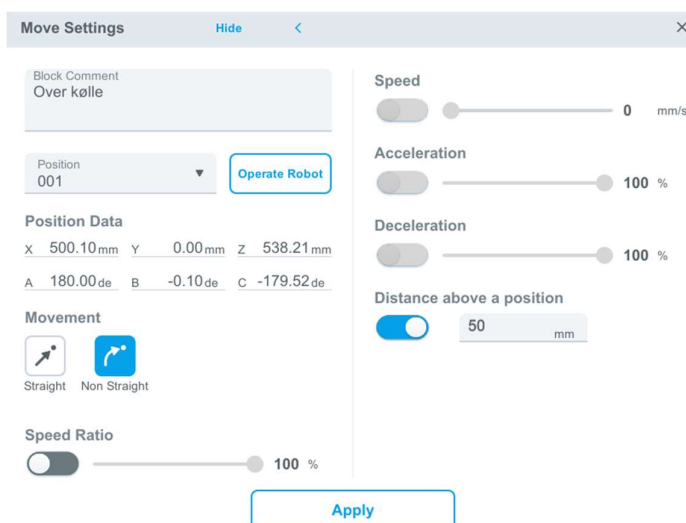
Løsning med RT VisualBox:

Når oppgaven skal løses i VisualBox er studentene avhengig å jobbe direkte mot roboten da det ikke er mulig å simulere dette i VisualBox. Her kan det være lurt at studentene får

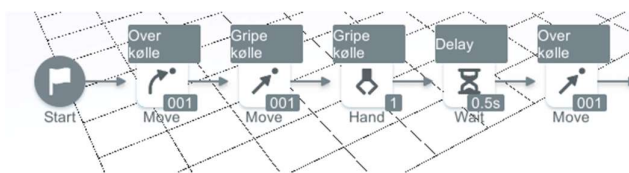
gjennomgang av programvaren og får mulighet å teste den ut selv før de gjennomfører lab-oppgaven.

Når programmet skal plukke opp køllen anbefales det at punktet der køllen ligger blir definert, det er mulig å lage en bevegelse til en gitt distanse over dette punktet (se bilde). I denne oppgaven kan det godt være mellom 50-100mm (dette vil avhenge av hvilken

situasjon en befinner seg i). Bevegelsen til dette punktet har ikke noe å si da roboten opererer i et fritt område, derfor bruker vi en ulineær bevegelse (Non Straight).



Resterende av operasjonen i det å plukke opp køllen kan for eksempel se slik ut:



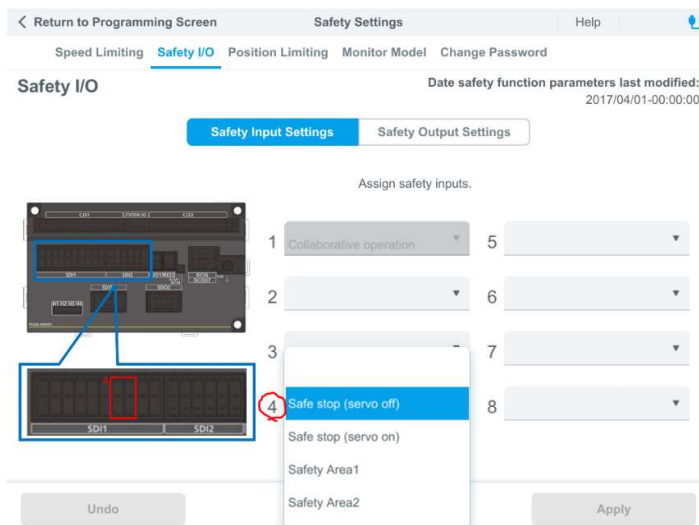
Her ser vi at punkt 1 blir brukt gjentatte ganger, og at når vi går ned/opp og plukker den opp vil vi ha en lineær bevegelse.

NB! Det er viktig å punktere for studentene at når en lager programmet er det veldig lurt å kommentere hver bevegelsene de legger inn.

Videre i programmet burde en ha en bevegelse til starten av løypen for så ha gjennomføring av bevegelsen bort til golfballen. Under gjennomkjøring av programmet er det tuppen av køllen som skal slå ballen. Utfordringen med øvingen er å finne ut hvor mye kraft en trenger for å klare å få ballen ned i hullet, det er også viktig å være presis i valg av bevegelse. Det er viktig å tenke når roboten skal rotere håndleddet at den ikke slår ned noe under gjennomkjøring. Dette er en typisk feil studentene kommer til å gjøre, samt kraften i støtet kan bli en utfordring, det er her de vil måtte jobbe med feilsøking.

Lysgardiner:

Her ser en hva safety input en skal velge når en kobler til lysgardinene. Som en ser så er det port 4 på SDI1 inngangen. Her kan en velge enten Safe stop (servo off)/(servo on). Det som er forskjellen på disse er om en vil at servoen til roboten skal være på eller av når en bryter lysgardiene.



Når en bruker lysgardiner i VisualBox vil roboten gå i full stopp når disse blir brutt.

Løsning med RT ToolBox3:

Når oppgaven skal løses i TB3 (RT ToolBox3) kan studenten både løse den direkte mot roboten eller i simulatoren. Anbefalt vil være å la gruppene løse den i simulator for så å koble seg opp mot roboten og laste programmene sine over på den for å kjøre dem der.

Når man skal laste opp programmet man lager vil det være behov for å laste programfil. I denne veiledningen finner man programmet og punkter brukt under video for løsningsforslag. Det vil være ønskelig å følge en lik teknikk for når en skal plukke opp kollen som ble gjort ved programmering i VisualBox. Under «programkode» i denne veiledningen ligger programkode som demonstrer hvordan det gjøres i TB3.

Lysgardiner:

I TB3 bruker vi SLS3 på DS14 for å at lysgardinene skal være aktive (se bilde på siden).

I TB3 kan en også styre om lysgardinene skal stoppe robotarmen eller sette ned farten når de blir brutt. Se der det står SLS3 i bilde på siden, den er satt til 50 mm/s. Dette betyr at hastigheten til armen blir satt til 50 mm/s når lysgardinene blir brutt. Dette kan studenten endre på, og de kan prøve ut andre hastigheter hvis ønskelig.

The image shows two screenshots from the RT ToolBox3 software interface.

The top screenshot is titled "Safety Input(Safety IO) 1:RCI (Simulation)". It displays a configuration table for safety signals. The columns are labeled SS1, SS2, SLS1, SLS2, SLS3, SLSM, SLP1, SLP2, SLP3, and SLP4. The rows are grouped into DS1, AREA, LOGIC, and MODE. The SLS3 column is highlighted in blue, and several cells in this column are checked, indicating that SLS3 is active for those signals.

	SS1	SS2	SLS1	SLS2	SLS3	SLSM	SLP1	SLP2	SLP3	SLP4
DS1					<input checked="" type="checkbox"/>					
DS2					<input checked="" type="checkbox"/>					
DS3					<input checked="" type="checkbox"/>					
DS4					<input checked="" type="checkbox"/>					
DS5					<input checked="" type="checkbox"/>					
DS6					<input checked="" type="checkbox"/>					
DS7					<input checked="" type="checkbox"/>					
DS8					<input checked="" type="checkbox"/>					
AREA1					<input checked="" type="checkbox"/>					
AREA2					<input checked="" type="checkbox"/>					
AREA3					<input checked="" type="checkbox"/>					
LOGIC					<input checked="" type="checkbox"/>					
MODE					<input checked="" type="checkbox"/>					

The bottom screenshot is titled "Speed 1:RCI (Simulation)". It displays speed monitoring settings. The "SLS3" column is highlighted in red. The "Comp. Speed" for SLS3 is set to 50.00 mm/s. The "Deceleration Response Time" is set to 1280 ms. The "Speed Limit OVRD" for SLS3 is set to 100.0000 [%].

Comp. Speed	SLS1	SLS2	SLS3	SLSM
1000.00	250.00	50.00	250.00	

Monitoring Speed for Each Direction	SLS1	SLS2	SLS3	SLSM
X+ :	100000.00	100000.00	100000.00	250.00
X- :	100000.00	100000.00	100000.00	250.00
Y+ :	100000.00	100000.00	100000.00	250.00
Y- :	100000.00	100000.00	100000.00	250.00
Z+ :	100000.00	100000.00	100000.00	250.00
Z- :	100000.00	100000.00	100000.00	250.00

Joint Monitoring Speed	SLS1	SLS2	SLS3	SLSM
11 :	100.00	100.00	100.00	100.00
12 :	100.00	100.00	100.00	100.00
13 :	100.00	100.00	100.00	100.00
14 :	100.00	40.00	40.00	100.00
15 :	100.00	100.00	100.00	100.00
16 :	100.00	100.00	100.00	100.00
17 :	100.00	100.00	100.00	100.00
18 :	100.00	100.00	100.00	100.00

Speed Limit OVRD	SLS1	SLS2	SLS3	SLSM
100.0000	100.0000	100.0000	100.0000	100.0000

Programkode:

```

Program 1:RC1LAB2LF.r (Simula... x
1  Mov p1_startPos      'Bevegelse til start posisjon
2  Mov p2_KollePos,-100 '100mm over køllens posisjon
3  Mvs p2_KollePos      'Lineær bevegelse ned til køllen
4  HClose 1            'Lukker igjen griperen og plukker opp køllen
5  Dly 0.5             'Pause 0.5 sek
6  Mvs p2_KollePos,-100 'Lineær bevegelse opp med køllen (100mm)
7  Mov p3_OverBall     'Beveger seg over mot golfballen
8  Mov p4_SlagPos      'Beveger seg til posisjonen bak ballen for å støte
9  Ovrd 15             'Setter hastigheten til 15% (Tilpasses etter forhold)
10 Mov p5_Slag         'Slår golfballen
11 Dly 3               'Pause på 3 sek for å la ballen trille
12 Ovrd 100           'Setter hastigheten til 100%
13 Mov p2_KollePos,-100 'Sekvensen for å legge tilbake køllen<-
14 Mvs p2_KollePos     '-----
15 HOpen 1            '-----
16 Dly 0.5             '-----
17 Mvs p2_KollePos,-100 '----->
18 Mov p1_startPos     'Kjører armen tilbake til startposisjon
19 Hlt                 'Avslutter programmet fullstendig, uten at det starter på nytt

```

Punkter:

Man kan enkelt bruke punkter man allerede har definer fra en tabell ved å kopiere og lime de inn i punkt listen i et annet program. **NB! Det kan ikke være noen punkter som har likt navn.**

Lab 2										
p5_Slag	-379.490	-426.340	29.390	-178.570	-0.100	-90.450			7	0
p4_SlagPos	-478.410	-425.570	26.920	-178.570	-0.100	-90.450			7	0
p3_OverBall	-479.030	-458.050	149.530	178.570	-0.070	-88.510			7	0
p2_KollePos	289.730	-412.540	38.470	-178.140	-0.150	-178.420			7	0
p1_startPos	5.490	-428.740	319.220	178.350	-1.420	-178.160			7	0