

Andreas Dollis Vik, Eline Berg, Lars Erik Langsåvold, Morgan Boe Johansen

Passasjerhåndteringssystem for den autonome urbane passasjerfergen milliAmpere2

Passenger handling system for the autonomous urban passenger ferry milliAmpere2

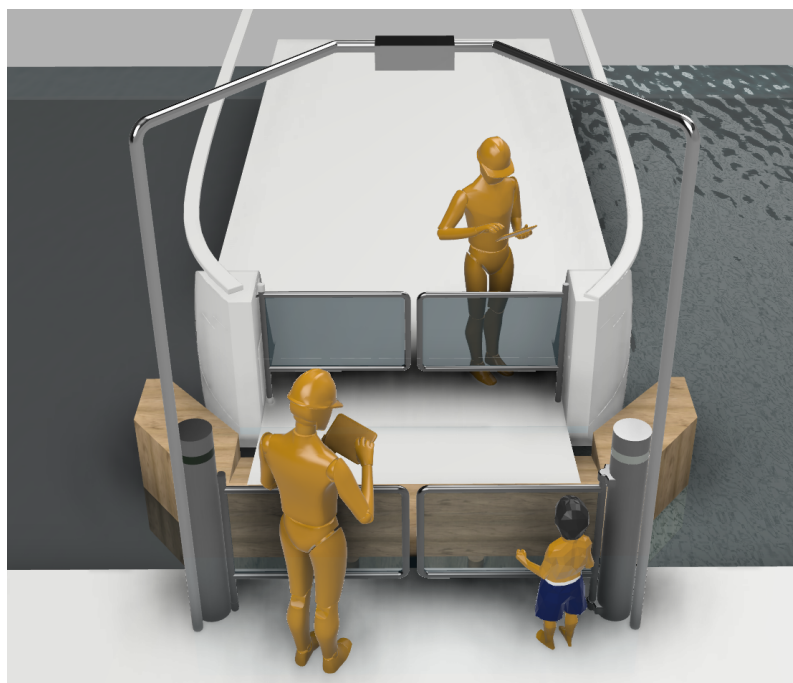
Bacheloroppgave i Automasjon og robotikk

Veileder: Morten Breivik

Medveileder: Egil Eide, Ole Andreas Alsos

Mai 2023

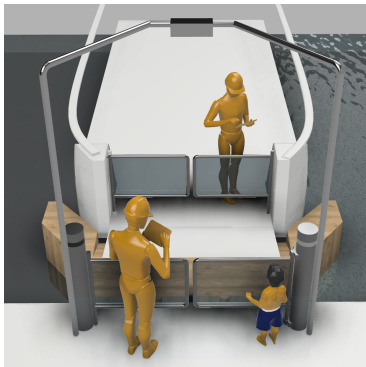
NTNU
Norges teknisk-naturvitenskapelige universitet
Fakultet for informasjonsteknologi og elektroteknikk
Institutt for teknisk kybernetikk



Andreas Dollis Vik, Eline Berg, Lars Erik Langsåvold,
Morgan Boe Johansen

Passasjerhåndteringssystem for den autonome urbane passasjerfergen milliAmpere2

Passenger handling system for the autonomous urban passenger ferry milliAmpere2



Bacheloroppgave i Automasjon og robotikk
Veileder: Morten Breivik
Medveileder: Egil Eide, Ole Andreas Alsos
Mai 2023

Norges teknisk-naturvitenskapelige universitet
Fakultet for informasjonsteknologi og elektroteknikk
Institutt for teknisk kybernetikk



Kunnskap for en bedre verden

Tittelside Bacheloroppgave BIELEKTRO

Oppgavetittel (norsk og engelsk): Passasjerhåndteringssystem for den autonome urbane passasjerfergen milliAmpere2 Passenger handling system for the autonomous urban passenger ferry milliAmpere2	
Forfattere: Andreas Dollis Vik Eline Berg Lars Erik Langsåvold Morgan Boe Johansen	Prosjektnummer: E2307
	Innleveringsdato: 22.05.2023
	Gradering: [x] åpen [] lukket
Studium: Elektroingeniør - BIELEKTRO	
Studieretning: Automatisering og robotikk	
Veileder: Morten Breivik Institutt: Institutt for teknisk kybernetikk	
Medveileder: Egil Eide Institutt: Institutt for elektroniske systemer Medveileder: Ole Andreas Alsos Institutt: Institutt for design	
Oppdragsgiver: NTNU: Institutt for teknisk kybernetikk, Trondheim Kontaktperson: Morten Breivik, morten.breivik@ntnu.no, 415 25 881	
Sammendrag Den autonome fergen «milliAmpere2» skal krysse kanalen fra Ravnkloa til Vestre Kanalkai i Trondheim, en strekning som utgjør omtrent 100 meter, uten at det er sikkerhetspersonell ombord. Fergen er en såkalt «on-demand»-ferge, som vil si at passasjerene skal kunne «bestille» fergen over til sin side ved å trykke på en knapp plassert på begge sider av fergesambandet. For at ombordstigning og overfart skal være trygg for passasjerene, er det viktig å ha på plass et automatisk passasjerhåndteringssystem som styrer porter, teller passasjerer og sørger for at ferga kun kjører når alt er klart til overfart. The autonomous ferry "milliAmpere2" is set to cross the canal from Ravnkloa to Vestre Kanalkai in Trondheim, a distance of about 100 meters, without any safety staff onboard. The ferry is an "on-demand" type, meaning that passengers can "order" the ferry to their side by pressing a button located on both sides of the ferry route. In order for boarding and crossing to be safe for passengers, it is important to have an automatic passenger handling system in place that controls gates, counts passengers, and ensures that the ferry only operates when everything is ready for crossing.	
Stikkord norsk: Automatisering, energi, elektrisitet, autonom, sikkerhet, ferge, programmering, klima, miljø, urban, passasjerer	Stikkord engelsk: Automation, energy, electricity, autonomous, safety, ferry, programming, climate, environment, urban, passengers

Sammendrag

Denne bacheloroppgaven dokumenterer utviklingen og tankegangen for et passasjerhåndteringssystem for milliAmpere2 i Autoferry-prosjektet til Norges teknisk-naturvitenskapelige universitet (NTNU). Dette omfatter design, utvikling, prototyping, programmering, systemarkitektur og planlegging. Målet med oppgaven er å lage et forslag som kan benyttes i senere tid, og som en dag kan realiseres i den virkelige verden.

Oppgaven består av totalt fire hovedkategorier, hvor hver hovedkategori deles opp i mindre kategorier. Disse kategoriene er fergeport, landport, sikkerhetssone og systemarkitektur. Designet og utviklingen til oppgaven er lagt opp for å senere kunne settes sammen med en annen bachelorgruppe. Dette er gjort slik at kommunikasjonen og kommandoer mellom land og ferge går sømløst. Den andre gruppen består av elektroingeniørstudenter med fordypning i elektronikk- og systemdesign ved NTNU.

Arbeidet i denne oppgaven er planlagt og designet etter oppgavebeskrivelsen gitt av oppdragsgiver. For gruppen innebar dette litteraturstudier, idéskaping, dialog med forskjellige samfunnsaktører og til slutt komme opp med et forslag til konsept. Innhenting av informasjon og prototyping skapte grunnlaget for mange av de valgene som er tatt. Gruppen har erfart at det aldri finnes én løsning, og at det å stille med åpent sinn og være åpen for forslag er viktig.

Med bakgrunn i oppgavebeskrivelsen og dialog med veiledere, har gruppen kommet frem til en løsning som er mulig å realisere. Mekanismen for styring av porter på fergen er tegnet i simuleringsprogrammet Autodesk Fusion 360 og 3D-printet for å teste mekanismen. Landportene er bestilt fra utlandet, og skal tåle kystklimaet som er i Trondheim. Sikkerhetssonen er programmert for å detektere objekter innenfor et gitt området, og skal forhindre skade på objektet ved bevegelse i port- eller lukemekanisme. Systemarkitekturen er ferdig programmert, og det gjenstår en test og eventuell feilsøking.

Målet gjennom hele prosjektet har vært å fremstille en prototype som kan monteres opp for testing og videre utvikling. Verkstedet ved Institutt for teknisk kybernetikk (ITK) på NTNU har vært en god støttespiller for å se på det praktiske rundt dette. På grunn av forsinkelser i beslutningsstaking og lang leveringstid har ikke alle essensielle komponenter blitt levert. En prototype av systemet ble derfor ikke bygget og koblet opp på grunn av dette.

Som en avslutning på rapporten har gruppen kommet med forslag til forbedringer og videreutvikling av konseptet.

Abstract

This bachelor's thesis documents the development and thought process behind a passenger handling system for milliAmpere2 in the Autoferry project at Norwegian University of Science and Technology (NTNU). This includes design, development, prototyping, programming, system architecture, and planning. The aim of the project is to create a proposal that can be utilized in the future, and which can one day be implemented in the real world.

The thesis consists of four main parts, each of which can be further subdivided. These parts are the ferry gate, terminal gate, the safety zone, and the system architecture. This thesis has been designed and developed to be integrated with another bachelor group. This is done so that communication and commands between the terminal and ferry flow seamlessly. The other group consists of electrical engineers specializing in electronics and system design at NTNU.

The work in this thesis has been planned and designed according to the project description given by NTNU. For the group, this involved literature studies, generating ideas, dialogue with different people and companies, and finally coming up with a concept proposal. Gathering information and prototyping formed the basis for many of the decisions made. The group has found that there is never a single solution, and that having an open mind and be receptive to suggestions is crucial.

Based on the project description and dialogue with the supervisor and advisors, the group has arrived at a feasible solution. The mechanism for controlling the ferry's gates has been designed in Autodesk Fusion and 3D printed for testing purposes. The terminal gate has been ordered from abroad and are built to withstand a coastal environment. The safety zone is programmed to detect objects within a given area and is designed to prevent objects from getting damaged due to movement in the hatch or gates.

The goal throughout the project has been to create a prototype that can be assembled for testing and further development. The Department of Engineering Cybernetics (ITK) workshop at NTNU has been a significant resource for practical considerations. Due to certain decisions and lengthy delivery times, not all the essential components have arrived. A prototype of the system was not built and tested due to delayed delivery.

As a conclusion to the report, the group has provided suggestions for improvements and further development of the concept.

Forord

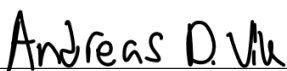
Med denne rapporten avslutter forfatterne sitt treårige bachelorstudium ved *Norges teknisk-naturvitenskapelige universitet (NTNU)* i Trondheim. Det er fire studenter ved studieretningen *Elektroingeniør* med spesialisering i *Automatisering og robotikk* som står bak verket. Oppgaven er levert av NTNU. Gruppen viser gjennom denne rapporten kunnskap de har tilegnet seg gjennom tre år på studiet.

Gruppen ønsker å benytte anledningen til å rette en stor takk alle involverte som har vært med å bistå det som til slutt viser å være vår bachelorrappport.

Vi ønsker å takke Leander Spyridon Pantelatos for digitale tegninger av fergen. Einar Hareide ved Hareide Design AS for innspill på design. Felix-Marcel Petermann for lån av MR-labben og utstyr. Robin Stokke for å simulere passasjerhåndterings-systemet vårt inn i Unity og hjelpe oss med å teste det ut i MR-labben. Glenn Angell ved Institutt for teknisk kybernetikk for hjelp til mekanisk tilvirkning. Bjørn-Olav Holtung Eriksen ved Zeabuz for innsikt i kommunikasjonsoppsettet av WAGO PLS-en ombord på milliAmpere2.

Vi ønsker å rette stor takk til våre veiledere Egil Eide, Morten Breivik og Ole Andreas Alsos for god hjelp, innspill og veiledning under oppgaven. Veilederne har bidratt med kunnskap fra hvert sitt fagfelt, og bidratt til læring for kandidatene.

Trondheim, 22.05.2023



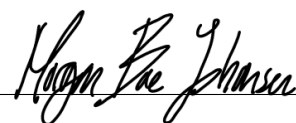
Andreas Dollis Vik



Eline Berg



Lars Erik Langsåvold



Morgan Boe Johansen

Forkortelser og ordforklaringer

Forkortelser

Ord	Forklaring
AFT-API	AutonomFergeTerminal - Application Programming Interface
AISI	American Iron and Steel Institute
ASCII	American Standard Code for Information and Interchange
AutoPAX	Auto = Automatisk, PAX = Passasjer
CAD	Computer Aided System, dataassistert konstruksjon
CPU	Central Processing Unit, prosessor eller hovedprosessor
CNN	Convolutional Neural Network
DMX	Digital Multiplex, kommunikasjonsprotokoll
GPU	Graphics Processing Unit, grafikkprosessor
IC	Integrated Circuit
IDE	Integrated Development Environment
I/O	Input/Output, inngang og utgang
MR	Mixed Reality, fysisk-virtuell verden
NMEA	National Marine Electronics Association
NTNU	Norges Teknisk-Naturvitenskapelige Universitet
PET	Polyetylentereftalat
PETG	Polyetylentereftalat-glykol
PLS	Programmerbar logisk styring
PMMA	Polymetylmetakrylat
PoE	Power over Ethernet, standard som tillater å sende strøm og data gjennom samme kabel
RS	Recommended Standard, kommunikasjonsprotokoll
TCP/IP	Transmission Control Protocol/Internet Protocol
UART	Universal Asynchronous Receiver-Transmitter, seriell kommunikasjonsprotokoll
UDP/IP	User Datagram Protocol, nettverksprotokoll
VAC	Voltage Alternating Current, Vekselstrøm
VR	Virtual Reality, virtuell verden
YOLO	You only look once, algoritme for datasyn

Ordforklaringer

Ord	Forklaring
3D-utskrift	Tredimensjonal lagvis bygging av objekter
Aktuator	En forstillingsmekanisme som kan benyttes til å styre bevegelser i en prosess
Autodesk Fusion 360	Dataprogram til 3D-design og -modellering
AutoPAX	Det automatiske passasjerhåndteringssystemet
BusLink	Programvare som konfigurerer LINAK-aktuatorer med IC
e!COCKPIT	Software til programmering av WAGO PLS
LINAK	Produsent av elektriske lineære aktuatorsystemer
milliAmpere1	NTNUs første prototype av autonome ferger for transport i urbane miljøer
milliAmpere2	En ferge utviklet og designet ved NTNU, for å forske på autonome ferger for transport i urbane miljø. Prototype nummer to
Shore control lab	Kontrollrommet til milliAmpere2 på Nyhavna
SOLIDWORKS	Et designprogram som brukes til å lage 3D-modeller og simulere mekaniske design
Spyder	Programmeringsplattform for Python-programmering.
Torghatten	Norsk rederi [1]
Ubuntu 18.04.5	Linuxbasert operativsystem, versjon 18.04.5
WAGO	PLS-produsent
Zeabuz	Spin-off firma fra det ledende forskningsmiljøet på autonome fartøy ved NTNU i Trondheim [1]

Innhold

Sammendrag	I
Abstract	II
Forord	III
Forkortelser og ordforklaringer	IV
1 Innledning	1
1.1 Bakgrunn og motivasjon	1
1.2 Oppgavebeskrivelse	4
1.2.1 Passasjerhåndteringssystemets oppgaver	5
1.3 Mål	6
1.4 Tidligere arbeid	6
1.5 Struktur - rapportens innhold	7
2 Bakgrunnsteori og metoder	8
2.1 Bakgrunnsteori til fergeportene	8
2.1.1 3D-printing	8
2.1.2 Lineær aktuatorene som roterer fergeportene	9
2.1.3 Bakgrunnsteori til lenker og ledd (kinematikk)	9
2.1.4 Tannhjul	12
2.2 Programvareverktøy brukt for å modellere systemet	14
2.2.1 Fusion 360	14
2.2.2 Spillmotoren Unity	15
2.3 Sikkerhetssone	15
2.3.1 Algoritmen YOLO	15
2.4 Systemarkitektur	16
2.4.1 Kommunikasjon mellom ferge og terminaler	16
2.4.2 Kommunikasjon på terminal	17
2.4.3 Kommunikasjon på ferge	18
2.4.4 Programvareverktøy brukt i systemarkitekturen	19
3 Systemkomponenter	20
3.1 Fergeport	20
3.1.1 LED-lys	21
3.1.2 Rammeverk	22
3.1.3 Festebrakett	23
3.1.4 Portstyring	23
3.1.5 Rotasjon av fergeport	23
3.1.6 Tannhjul	32
3.2 Landport	37
3.3 Sikkerhetssone	38
3.3.1 Deteksjon av objekter i sikkerhetssonen	39
3.4 Mixed Reality Laboratory (MR-lab)	41

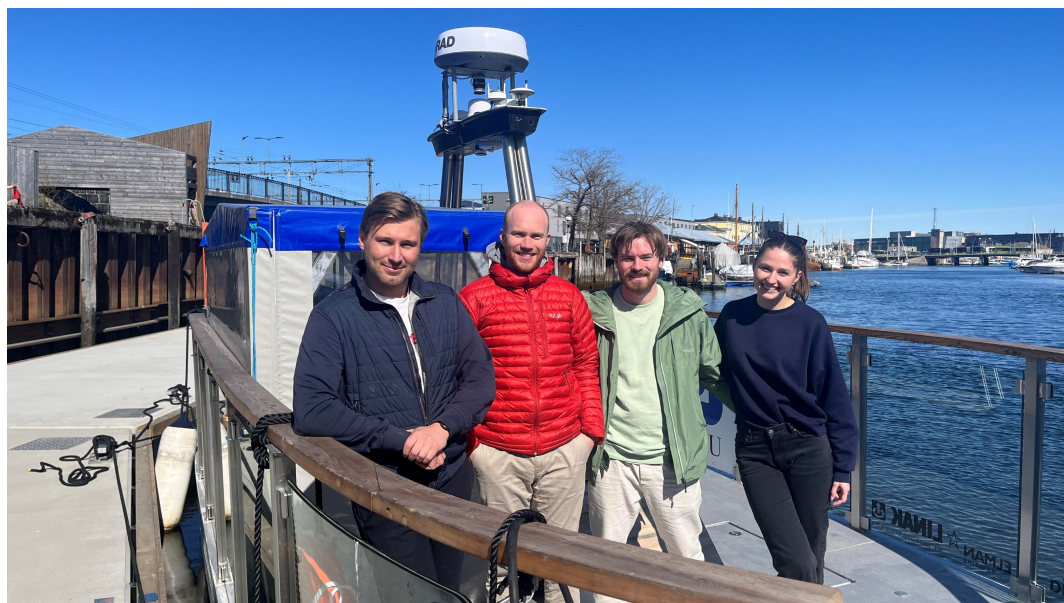
3.5	Landkontroll	41
3.5.1	Kommunikasjon mellom PC og landterminal	42
3.6	Systemarkitektur for ferge og terminaler	43
3.6.1	Fergeterminal	44
3.6.2	Ferge	46
3.6.3	Terminal	54
4	Prototype passasjerhåndteringssystem	60
4.1	Fergeport	61
4.1.1	Diskusjon	64
4.2	Landport	64
4.2.1	Diskusjon	65
4.3	Testing av systemet	65
4.3.1	Resultat ved bruk av ledd og aktuator	65
4.3.2	Resultat ved bruk av aktuator og tannhjul	67
4.4	Sikkerhetssone	68
4.4.1	Programvare	68
4.4.2	Filkonfigurering og kjøring av program	69
4.4.3	Kamera og design	71
4.4.4	Diskusjon om kommunikasjon mot ferge	72
4.4.5	Sikkerhetsbøyle	72
4.4.6	Diskusjon	72
4.5	Telling av passasjerer	73
4.5.1	VR-briller	74
4.6	Systemarkitekturen	75
4.6.1	Ferge	76
4.6.2	El-skap tilhørende terminal	80
4.7	Sekvensiell styring	80
4.7.1	Forklaring	80
4.7.2	Manuell drift	81
4.7.3	Automatisk drift	81
4.8	Risikoanalyse	82
5	Konklusjon og videre arbeid	85
5.1	Konklusjon	85
5.2	Videre arbeid	87
5.2.1	Fergeport	87
5.2.2	Landport	88
5.2.3	Montering av sikkerhetsbøyle over sikkerhetssone	88
5.2.4	El-skap	89
5.2.5	Speiling av systemet	89
5.2.6	Programmering	89
5.2.7	Testing	90
	Referanser	91
	Vedlegg	96
	A - GitHub, kode og Fusion 360-filer	96
	B - Sekvensielle skjemaer, portstyring, ankomst og avgang	97

C - Sekvensielle skjemaer, manuell styring	100
D - AFT-API grensesnitt og kommandoer	104
E - NMEA0183 kommando og statusmeldinger, lukestyring	120
F - Landport koblingsskjema	122
G - Landport seriellprotokoll	124
H - Koblingsskjema og rekkeklemmetabell for landterminal	135
I - Bestillingslister	143
J - I/O liste - WAGO I/O-maskin	147
K - Populærvitenskapelig artikkel	149

Kapittel 1

Innledning

1.1 Bakgrunn og motivasjon



Figur 1.1.1: Bachelorgruppen ombord på milliAmpere2. (Fra venstre: Lars Erik Langsåvold, Morgan Boe Johansen, Andreas Dollis Vik, Eline Berg)

På bakgrunn av Trondheims behov for en passasje mellom Ravnkloa og Fosenskaia kom førsteamanuensis ved Institutt for elektroniske styresystemer, Egil Eide, sammen med Kystlaget i 2016 opp med en helt ny idé: "Hva med å utvikle en elektrisk selvgående ferge i stedet for å bygge en bro som kunne skape trøbbel for båttrafikken?" [2].

Og med det gikk startskuddet for utbyggingen av Norges aller første selvkjørende ferge. I dag er forskningsprosjektet på prototype nummer to, milliAmpere2, som det er planer om skal forske på transport av passasjerer. For å gjøre brukeropplevelsen så trygg som mulig er fergen under konstant utvikling.

Idéen om autonome passasjerferger langs kanalen er benyttet av Trondheim Kommune som erstatning til tidligere planer om utbygging av gang- og sykkelbro. Dette skyldes at det å drive skytteltrafikk over mindre strekninger med autonome passasjerferger anses som mer miljøvennlig og fleksibelt enn det utbygging av broer ville vært. Det ga også Trondheim en mulighet til å være banebrytende innen teknologien for selvkjørende passasjerferger, en utvikling som anses som unik i verden.

Avansert sensorteknologi som lasersyn, kameraer og radar er utstyrt om bord på den autonome passasjerfergen milliAmpere2 [2]. Det er i tillegg bedre plass til passasjerer enn på forgjengeren, milliAmpere1. Videre har den en økt batterikapasitet og en kraftigere maskinvare for blant annet dynamisk posisjonering.

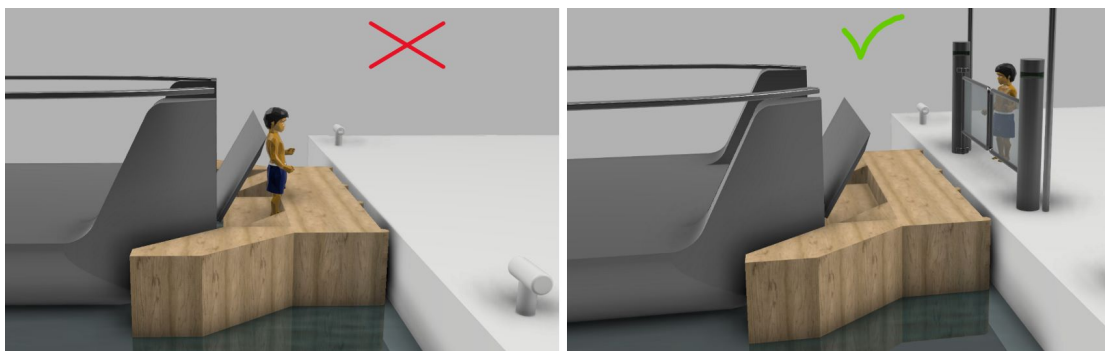
Interessen for teknologien i autonome passasjerferger i urbane strøk øker internasjonalt. Det er spinoff-selskapet Zeabuz som kommersialiserer NTNU-teknologien og skal sammen med fergerederiet Torghatten lansere en autonom passasjerferge i Stockholm i 2023 [1].

Ved bruk av autonome ferger er interaksjonen mellom mennesket og maskin sentral. Uten en sikkerhetsvert må det inkluderes flere tiltak for å ivareta sikkerheten. Reaksjoner og opplevelser fra passasjerene sitt perspektiv gir en stor indikasjon på hva som bidrar til å gjøre opplevelsen trygg eller utrygg [2].



Figur 1.1.2: Dronebilde av fergen i transport med personer om bord. Bildet er tatt under prøvedriften høsten 2022 og viser fergen som legger fra kai [3].

Målet er at den autonome fergen, milliAmpere2, skal forskes på mens den går i skytteltrafikk uten mannskap om bord, over en av Trondheims kanaler mellom Ravnkloa og Fosenkaia. På grunn av sikkerhetsmessige grunner vil det være en sikkerhetsvert på et kontrollrom som er plassert på land. Hensikten med dette kontrollrommet er å overvåke fartøyene og følge med på at ting går som det skal, og ved uønskede hendelser gripe inn.



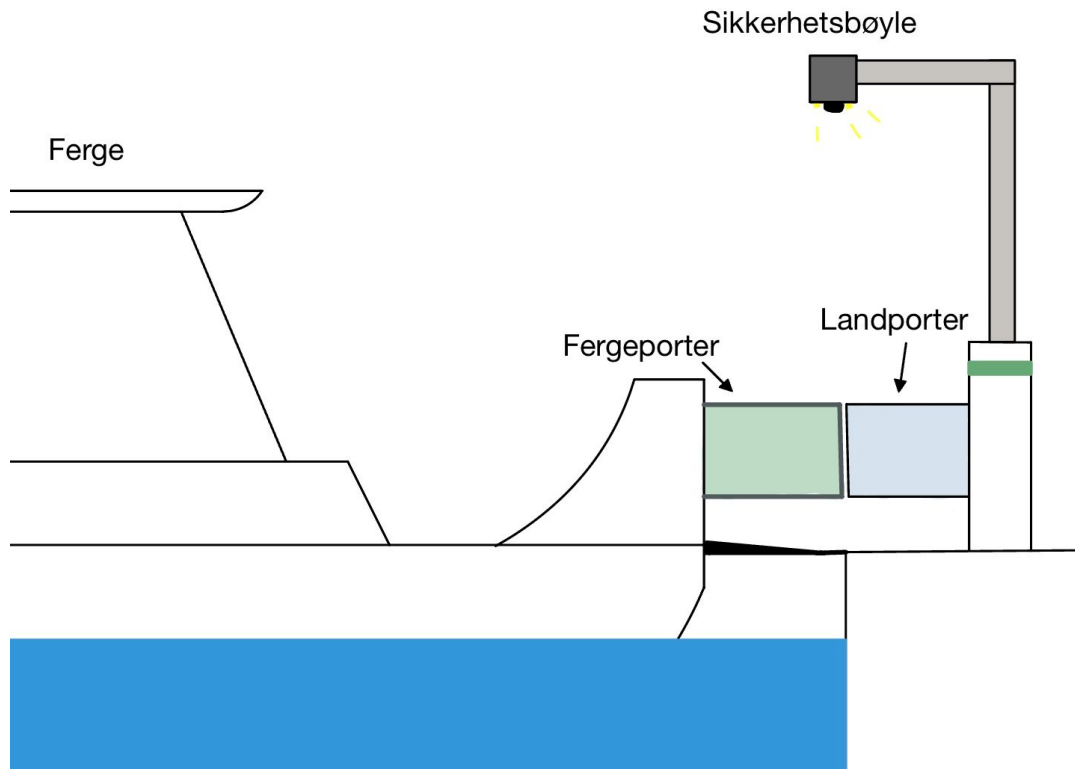
Figur 1.1.3: Eksempelfigur uten og med sikkerhetsport på land. Forhindrer at personer kommer i klem når fergen legger til kai.

For å kunne erstatte denne sikkerhetsverten om bord den autonome fergen må det legges til automatiske sikkerhetssystemer som ivaretar passasjerene. Det er i på- og avstigningsprosessen risikoen er størst. Når fergen legger til kai, legger den ned lemmen i enden av fergen. For å ivareta sikkerheten er det viktig å påse at ingen personer eller dyr befinner seg i området der fergen legger til. For å hindre uvedkommende inn i faresonen er det hensiktsmessig å installere fysiske og praktiske barrierer.

Automatiske sikkerhetsporter sammen med sensorer er et alternativ som kan sikre trygg og effektiv passasjerflyt og en unngår da uønskede situasjoner når de autonome delene skal utføre bevegelser. Kamera eller andre sensorer kan da gi beskjed til en forhåndsprogrammert kontrollenhet om det er objekter i faresonen. Kontrollenheten analyserer informasjonen og styrer porten ved hjelp av aktuatorer eller motorer basert på den gitte situasjonen.

For å tydeliggjøre når det klart for av- og påstigning, kan automatiske porter kombineres med lys- og akustiske signal som er enkle å forstå. Sammen vil dette bidra til en trygg og forutsigbar prosess for alle involverte.

1.2 Oppgavebeskrivelse



Figur 1.2.1: Konseptskisse av passasjerhåndteringssystemet.

Når fergen ankommer kaien kobler den seg fast og senker lemmen slik at passasjerer kan gå om bord. Når fergen skal legge fra vil sikkerhetsverten aktivere prosessen hvor fergen lukker lemmen og legger fra kai. Dette systemet er ikke trygt nok i dag til at fergen kan drives uten mannskap. Derfor må en sikkerhetsvert involveres for at fergen kan legge fra kai. For å få milliAmpere2 til å bli en helautonom ferge er det derfor et behov for et oppdatert passasjerhåndteringssystem som kan dekke disse sikkerhetsbehovene ved inn- og utfart, samt en knapp der passasjerene selv kan si i fra at det er klart for avreise.

Prosjektoppgaven inneholder flere problemstillinger som da til sammen representerer et passasjerhåndteringssystem for den autonome fergen milliAmpere2. Problemstillingene er som følger:

- Utvikle systemarkitektur som både beskriver funksjonell virkemåte og fysisk maskinvare.
- Systemarkitekturen skal inneholde grensesnitt mot styringssystemet om bord i fergen og mot bestillings-/billettsystem.
- Systemarkitekturen skal beskrive hvilket av systemene som utfører styring av porter, kommunikasjon, prioritering, og igangsetting av seilas med fergen.

- Det skal utføres en sårbarhets- og sikkerhetsanalyse av systemet for å sørge for at det ikke oppstår uønskede situasjoner eller fare for passasjerene.
- Det skal utvikles hardwareløsning for portene slik at disse både blir vandal-sikre og tåler miljøet de skal operere i med hensyn til temperatur, regn og fukt.
- Hvis tiden tillater det, skal det bygges prototype og teste hele eller deler av systemet.

1.2.1 Passasjerhåndteringssystemets oppgaver

Passasjerhåndteringssystemet skal utføre flere oppgaver for å gjøre fergen trygg for passasjerene. Løsningene for de ulike oppgavene skal være brukervennlige, robuste og formidle et visuelt forståelig budskap. Det skal ikke bare være trygt, det skal også føles trygt ut med og å forebygge uønskede situasjoner.

Passasjerhåndteringssystemet skal utføre følgende oppgaver:

- Styre portene på kaien og fergen direkte gjennom trykknapp-brytere, eller indirekte gjennom styringssystemet om bord på fergen, så portene kun åpner for passasjerer når fergen er fortøyd og sikret.
- Detektere mennesker som er i ferd med å gå av eller på fergen.
- Gi lyd- og lyssignal som varsler passasjerene om at overfart starter og porter stenger.

I tillegg skal det lages en trådløs forbindelse mellom portene på fergen og kaien slik at de beveger seg samtidig. Dette systemet skal ha et grensesnitt mot det allerede eksisterende styringssystemet som er på fergen:

- Innganger: Tilbakemelding på om fergen er fortøyd, klarsignal for at bommer kan åpne, signal om at overfart skal starte og at porter kan lukke.
- Utganger: Status på sikkerhetsporter, status på om folk er i ferd med å gå av eller på fergen, antall passasjerer om bord.

milliAmpere2 er en *"on demand"*-fergetjeneste. Det vil si at passasjerer har mulighet til å anmode om dens tjenester på hver side av kanalen ved hjelp av en bestillingsknapp som er plassert ved både Ravnkloa og Fosenkaia. Dette systemet skal være designet til å ha en sømløs interaksjon med et annet system, som er utviklet av en annen bachelorgruppe [4].

Systemarkitekturen til passasjerhåndteringssystemet må kunne snakke med det allerede eksisterende systemet samt den andre bachelorgruppens landterminaler. For å oppnå optimal funksjonalitet mellom systemene må jobbe parallelt og samarbeide.

1.3 Mål

Dette ønsker gruppen å oppnå gjennom prosjektet:

- Lage et pålitelig system.
- Lage prototype av porter på land og ferge.
- Lage funksjonelle fysiske porter på land og ferge.
- Utveksle informasjon med den landbaserte løsningen.
- Utveksle informasjon med det eksisterende autonome systemet.
- Utveksle informasjon gjennom lys og lyd til passasjerer.
- Lage et system som ikke setter passasjerer i fare. Lem går ikke opp om det står mennesker på den.
- Lage en virtuell testløsning av portene.
- Designe et brukervennlig og intuitivt portsystem.
- Lage en løsning som er robust.

Målene er senere benyttet som referansepunkter for evaluering og diskusjon i resultat- og konklusjonsdelen i rapporten.

1.4 Tidligere arbeid

Denne seksjonen tar for seg tidligere forskning og arbeid som har inspirert måten gruppen løser ulike problemstillinger som har oppstått på, samt lignende prosjekter som har relevans for denne rapporten.

Det er skrevet en bachelorrappport som omhandler design av et automatisk dokking-system for en mindre autonom passasjerferge av to studenter ved Institutt for maskinteknikk og produksjon i 2019. Dette er en bacheloroppgave som har gitt inspirasjon til denne rapporten og som er relevant for eventuell videreutbygging av passasjerhåndteringssystemet [5].

Designet til sikkerhetsbøylene er laget med inspirasjon fra masteroppgaven om "Design av brukerreise og brygger til autonom passasjerferge" [6]. Gruppen ønsket å utvikle en bøyler som detekterer sikkerhetssonen og som kan belyse område hvor passasjerene stiger av og på fergen.

Sikkerhetsbøylens utseende og virkemåte er inspirert av AutoPASS-skannere som finnes på fergekaier. De er montert i buen over kjørefeltet og skanner AutoPASS-brikkene til bilene under passasje [7].

Når det gjelder sikkerhetsporter i forbindelse med ferge- og landporten er det hentet inspirasjon fra dagligvarebutikker. Når en går inn i en butikk blir man som regel møtt av en automatisk svingedørspport. I butikkene har denne en sensor som registrerer at noen skal inn i butikken. Gruppen har derfor vært i dialog med de samme leverandørene som supplerer norske dagligvarekjeder med disse portene for

å undersøke muligheten til å bestille en komplett port som kan benyttes utendørs [8].

For videre utbygging er det relevant å se på Zeams lansering av verdens første kommersielle autonome passasjerferge mellom Kungsholmen og Södermalm i Sverige. Denne utbyggingen gjøres i samarbeid med Zeabuz som er verdensledende innen autonom teknologi for maritim sektor [1]. Zeam er eid av det norske rederiet Torghatten.

1.5 Struktur - rapportens innhold

Denne rapporten er inndelt i fem kapitler. Hvert kapittel inneholder egne seksjoner, som igjen har underseksjoner.

Kapittel 1 - Innledning tar for seg bakgrunnen for denne rapporten, spesifikt hva prosjektoppgaven går ut på og målene gruppen har laget for å kunne utfylle prosjektoppgavens krav. I tillegg er tidligere arbeid som er relevant for oppgaven presentert her.

Kapittel 2 - Bakgrunnsteori og metoder gir det nødvendige grunnlaget for å få en god forståelse av rapporten. Kapitlet går ikke i dybden på de ulike teamene som drøftes, da det ikke er nødvendig for den overordnede forståelsen av rapporten.

Kapittel 3 - Systemkomponenter går i detalj på hvorledes gruppen kommer frem til de ulike komponentene som til sammen skal utgjøre passasjerhåndterings-systemet.

Kapittel 4 - Prototype av passasjerhåndteringssystem presenterer resultatene gruppen får under utvikling av prototypene. Hver prototype er fordelt inn i egne seksjoner. Disse inneholder en diskusjonsdel som drøfter ulike valg som er tatt, og hva slags informasjon resultatene inneholder.

Kapittel 5 - Konklusjon og videre arbeid tar først for seg en kort oppsummering av hovedfunn og resultat. De viktigste resultatene er drøftet, samt hva slags påvirkning de har for videre arbeid.

Videre arbeid gir leseren innblikk i hva som kan gjøres videre basert på funn og konklusjoner i rapporten. Det er i denne seksjonen det blir presentert anbefalinger på hvordan leseren kan forbedre og fortsette på arbeidet.

Kapittel 2

Bakgrunnsteori og metoder

Dette kapitlet tar for seg den relevante teorien som skal gi leseren de grunnleggende bakgrunnskunnskapene for god forståelse og læring videre i rapporten, samt en god oversikt over den allerede eksisterende kunnskapen på feltet. Kapitlet går ikke i dybden på de ulike teamene som drøftes, da det ikke er nødvendig for den overordnede forståelsen av rapporten.

Målet med dette kapitlet er å gi leseren en god forståelse for hvilket teoretisk grunnlag prosjektet bygger på og se rapporten sitt tema i en større sammenheng.

2.1 Bakgrunnsteori til fergeportene

2.1.1 3D-printing

3D-printing, også kalt additiv produksjon, er en nyere produksjonsteknikk hvor en lag på lag bygger tredimensjonale objekter. Metoden benytter digitale modeller som utgangspunkt. For å skrive ut objekter trenger en et såkalt filament. Dette smeltes ofte på en temperatur mellom $200 - 300^{\circ}C$, og påføres ved hjelp av et oppvarmet skriverhode. Det er et stort utvalg av materialer en kan benytte i 3D-utskrift. Hvilket materiale en ønsker å benytte avhenger av sluttproduktets bruksområder [9].

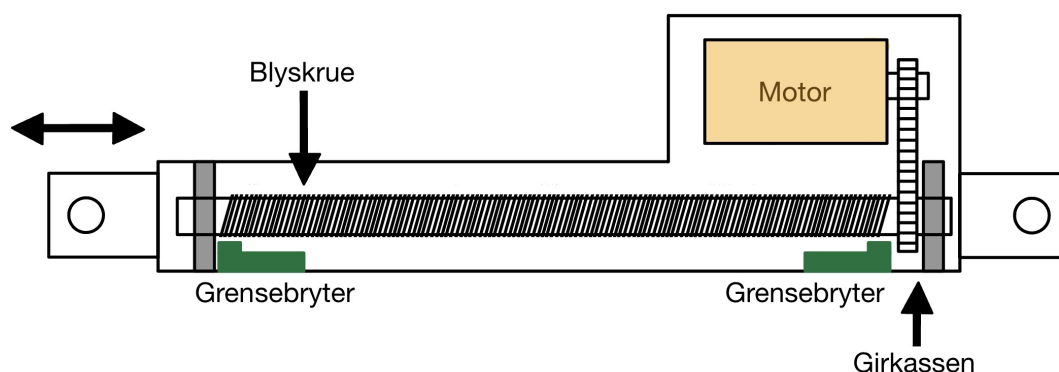
Fordelen med 3D-utskrift er at det er kostnadseffektivt og materialbesparende i form av at det kun bruker den nødvendige mengde materiale for å oppnå ønsket geometrisk produkt. De tradisjonelle maskineringsmetodene baserer seg på å fjerne råmateriale gradvis for å oppnå ønsket produkt.

Ulempen er at de utskrevne objektene kan ha noe lavere styrke og holdbarhet kontra de tradisjonelt maskinerte. Dette kan forklares med at egenskapene til det valgte filamentet kan ha flere svakheter sammenlignet med de tradisjonelle maskineringsmetodene [10].

2.1.1.1 Utskriftmaterialet polyetyltereftalat-glykol

Polyetyltereftalat-glykol (PETG) er termoplast som er kjent for å være et brukervennlig og sterkt 3D-filament [11]. Stoffet utgjør en sammensetning av polyetyltereftalat og glykol. Polyetyltereftalat (PET) er oftest kjent som gjennom-siktige plastflasker og -beholdere. PET er alene veldig sprøtt, og derfor tilsettes glykol for å øke fleksibilitet og styrke. PETG egner seg godt i forskjellige omgivelser på grunn av motstandsdyktigheten, men har likevel noen svakheter. Materialet er blant annet fuktighetsabsorberende som kan bryte ned materialet over tid hvis det ligger under vann [12].

2.1.2 Lineær aktuatorene som roterer fergeportene



Figur 2.1.1: Tegning av lineæraktuator [13].

En lineær aktuator er en mekanisk enhet som utfører en lineær bevegelse ved å konvertere energi og signaler som går inn i systemet over til en mekanisk bevegelse. Lineær aktuatorer kan være drevet av elektronikk, pneumatikk eller hydraulikk.

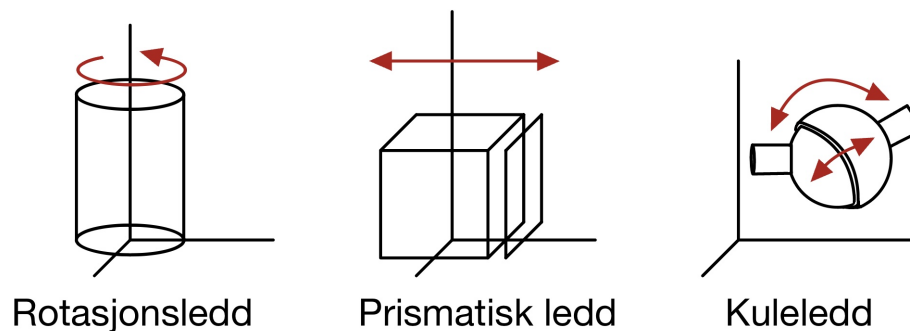
Når DC-motoren kobles til en positiv DC-forsyning, roterer motoren med klokken som da får giret til å rotere. Giret bremser ned den høye hastigheten til motoren og blyskruen roterer med klokken som får slagstangen til å bevege seg lineært vekk fra aktuatoren. Om DC-motoren er koblet til en negativ DC-strømforsyning skjer samme hendelsesforløp bare i motsatt retning som vil føre til at slagstangen trekker seg innover i aktuatoren [13].

Det er montert en mutter på blyskruen som beveger seg langs gjengene til skruen. Når mutteren beveger seg trekker eller skyver den slagstangen, som da går ut fra aktuatoren, i en lineær retning [14]. Når slagstangen går til største eller minste slaglengde vil grensebryteren bli trigget og da stoppe motoren som stopper bevegelsen til blyskruen.

Se figur 2.1.1 for å se de ulike delene til aktuatoren og hvordan de er satt sammen.

2.1.3 Bakgrunnsteori til lenker og ledd (kinematikk)

2.1.3.1 Kinematiske kjeder

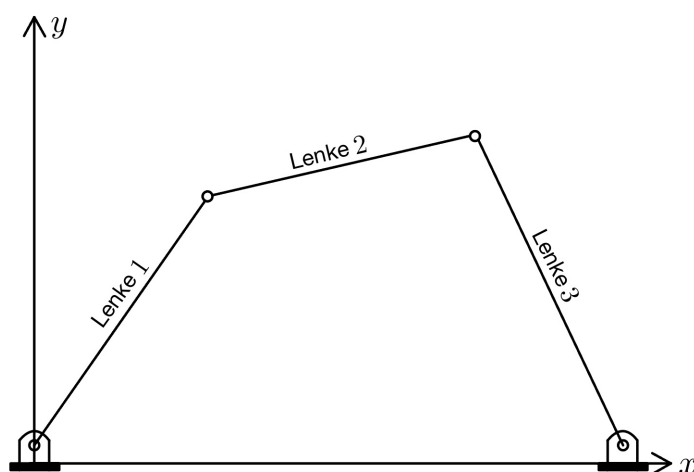


Figur 2.1.2: Ulike leddtyper.

Kinematikk er studiet av bevegelser i et system hvor en ikke tar hensyn til kreftene som forårsaker bevegelsene. Kinematikk fokuserer på å beskrive bevegelsene til systemet gjennom matematiske og grafiske beskrivelser. En kinematisk kjede er et system av sammenkoblede lenker, og ledd som kan utføre bevegelser på en bestemt måte. Ved å kontrollere bevegelsen i en eller flere ledd kan en styre bevegelsen til hele systemet.

Ledd kan være simple, som prismetisk- og rotasjonsledd, eller mer kompliserte som kuleledd, se figur 2.1.2. Forskjellen på de to leddtypene er hvor mange frihetsgrader de har. Prismatiske og rotasjonsledd har en frihetsgrad, mens kuleledd har to [15].

2.1.3.2 Lukket kinematisk kjede



Figur 2.1.3: En lukket kinematisk kjede. Bevegelsen til ledd 1 vil påvirke bevegelsen til ledd 3.

En lukket kinematisk kjede er et system der leddene er koblet sammen som en løkke, som vil si at en leddbevegelse vil påvirke de andre leddene sine bevegelser. Ettersom systemet er en lukket løkke vil bevegelsene være begrenset av systemets ledd, og en trenger da ikke ekstern kraft eller input for å kontrollere systemet.

Ettersom en lukket kinematisk kjede bare er avhengig av egne bevegelser er den mer stabil, forutsigbar og lettere å kontrollere enn det en åpen kinematisk kjede er. Endelenkene til den åpne kinematisk kjeden er ikke koblet sammen med den første lenken i systemet og en kan få flere ulike resultat [15].

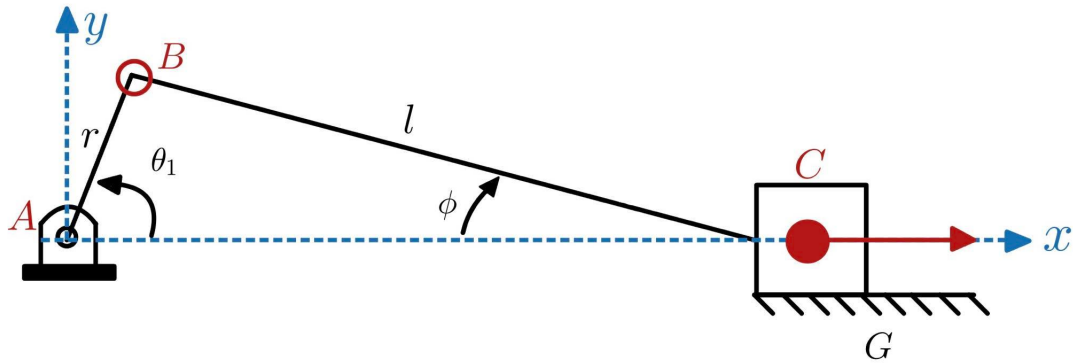
2.1.3.3 Inverskinematikk

Inverskinematikk uttrykker hvilke leddkonfigurasjoner roboten må ha for at end-effektor skal kunne oppnå ønsket posisjon (x, y, z) , og leddvinkel (θ) . For å finne posisjonen og vinklingen til systemets ledd, er en av metodene å bruke geometriske tilnærminger. Det er krevende å finne en generell løsning til systemer ved bruk av inverskinematikk ettersom systemene ofte er ulineære. Eventuell løsning er heller ikke unik [15].

2.1.3.4 Overaktuert system

Når et system er overaktuert har det flere fkontrollinnganger enn nødvendig for å oppnå et resultat. Det vil si at systemet har flere måter å kontrollere oppførselen på, enn hva som er nødvendig for å oppnå ønsket resultat. Når et system er overaktuert kan en få et overkomplisert system som er krevende å kontrollere. Det kan også føre til at flere inputs påvirker hverandre og det oppstår uønskede bevegelser [15].

2.1.3.5 Veivaksel-mekanisme



Figur 2.1.4: Skisse av veivaksel-mekanisme.

En veivaksel-mekansime, på engelsk Single Crank Mechanism, er en sammensetning av mekaniske deler som skal konvertere en rotasjons bevegelse til en lineær bevegelse eller omvendt. Denne mekanismen består av tre deler, en aksel (A), en koblingsstang (B) og et stempel (C).

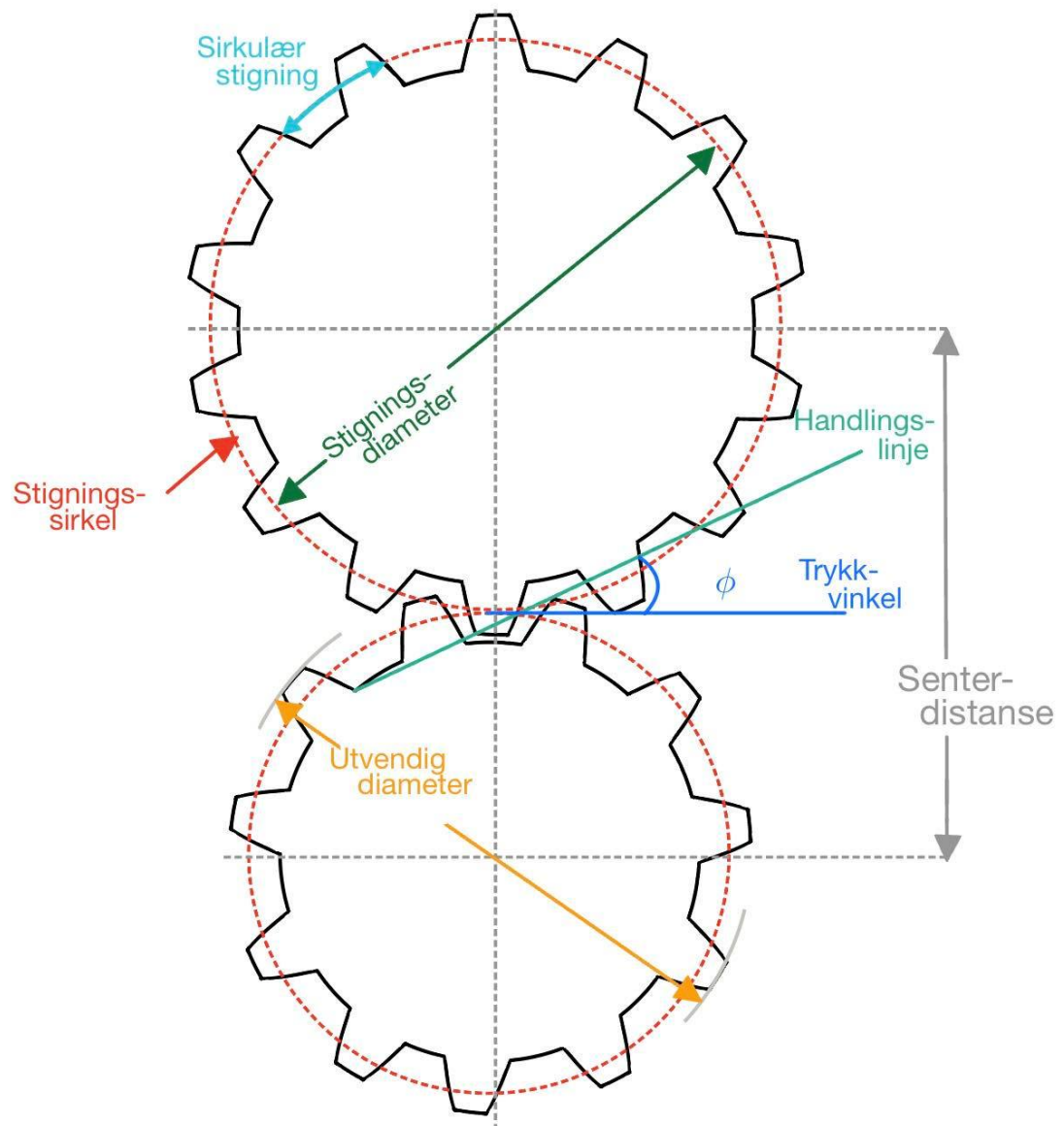
Veivakselen er et ledd som er koblet til en roterende aksel med vinkelen θ_1 , og er koblet til stempelet ved hjelp av en koblingsstang med fast lengde l . Stempelet har en bevegelse langs G som er avhengig av verdien til θ_1 . Om θ_1 øker vil stempelet, C , bevege seg langs x-aksen mot origo (A). Dersom θ_1 minker vil C bevege seg langs x-aksen i retning fra origo. r er avstanden mellom veivakselen, A , og koblingsstangen, B [16]. Se figur 2.1.4 for å se skisse av veivaksel-mekanismen.

Sammenhengen mellom vinklene i systemet, θ_1 og ϕ , og de konstante lengdene, r og l , kan brukes for å finne posisjonen til C [17]

$$\phi = \frac{\sin(\theta_1) \cdot r}{l} \quad (2.1)$$

$$C = r \cdot \cos(\theta_1) + l \cdot \cos(\phi). \quad (2.2)$$

2.1.4 Tannhjul



Figur 2.1.5: Skisse av to tannhjul. Se tabell 2.1.1 for ordforklaring.

Norsk navn	Engelsk navn	Symbol	Beskrivelse
Girutveksling	Gear ratio	m_G	Forholdet mellom tannhjulene
Handlingslinjen	Line of action	L_a	Linjen som krysser kontaktpunktene mellom tennene til tannhjulene. Strekker seg fra første kontaktpunkt til siste
Senterdistanse	Center distance	C	Avstanden mellom de to tannhjulene sine sentre
Sirkulær stigning	Circular pitch	p	Tykkelsen til tennene og mellomrommet
Stigningsdiameter	Pitch diameter	d	Diameteren til den teoretiske sirkelen som er hvor kontaktpunktet mellom tannhjulene sine tenner skal være
Trykkvinkel	Pressure angle	ϕ	Vinkelen mellom handlingslinjen og linjen som er vinkelrett på senterlinjen

Tabell 2.1.1: Ordforklaring til figur 2.1.5 [18] og [19].

Tannhjul er en girtype som består av en sylinder eller disk med tenner rundt egen aksel. I industrien er den brukt som et industrielt verktøy for å overføre mekaniske bevegelser samt styre hastighet, kraft og dreiemoment. Tannhjul er økonomisk gode, langvarige og pålitelige.

Designet til tannhjulet er enkelt og består av rette, parallelle tenner plassert rundt den sylindriske kroppen med et hull i midten. I dette hullet blir det koblet til en aksling som tannhjulet kan rotere rundt. Ved å sammenkoble en serie med tannhjul kan en øke eller redusere hastigheten til enheten eller multiplisere dreiemomentet og kraften fra en aksling til et annet tannhjul.

2.1.4.1 Sammenkobling av tannhjul i serie

Hensikten ved å sammenkoble tannhjul er å endre hastigheten og dreiemomentet. Størrelsen på hastigheten og dreiemomentet er bestemt av girutvekslingen mellom tannhjulene. Girutvekslingen er forholdet mellom tannhjulene.

Når en sammenkobler tannhjul er det flere faktorer en tar hensyn til for å unngå slitasje og skade på tennene. Tannhjulene må justeres rett i forhold til hverandre for å sikre både jevn og effektiv kraftoverføring mellom seg. Avstanden mellom origoen til tannhjulene, også kjent som senterdistanse, se tabell 2.1.1, må være plassert rett, så tennene som møtes treffer hverandre rett. For å bevare at tannhjulene opererer pålitelig og effektivt er det viktig at tannhjulene som er i systemet har kompatibel stigning, trykkvinkel og modul eller diametral stigning [20].

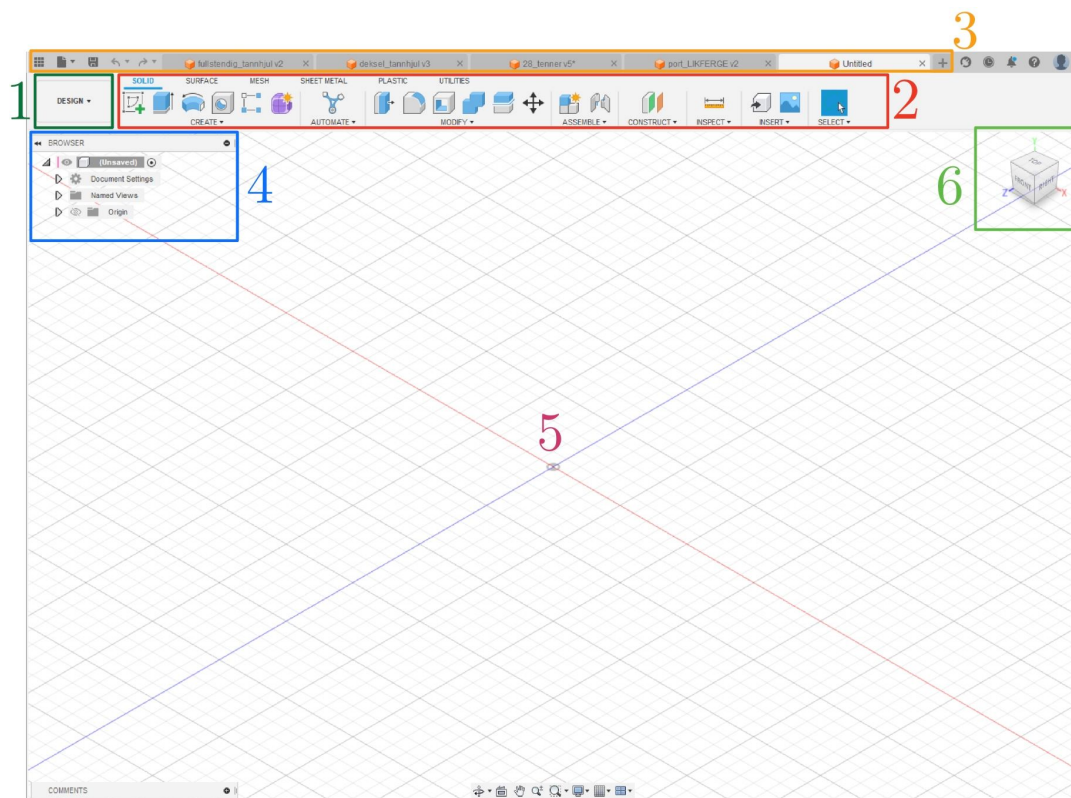
2.2 Programvareverktøy brukt for å modellere systemet

2.2.1 Fusion 360

Autodesk utviklet Fusion 360, et 3D dataassistert konstruksjonsprogram [21]. Når en omtaler dataassistert konstruksjon bruker en forkortelsen CAD, som står for det engelske ordet Computer-Aided Design.

Autodesk lanserte Fusion 360 for første gang i 2013 og programmet har siden opplevd en enorm vekst i popularitet. Programmet har omfattende verktøy som kan brukes til å lage 3D-modeller og simulere ulike design av komplekse modeller og prototyper. Som student er dette et program som er gratis å bruke for alle studenter.

2.2.1.1 Oppsett



Figur 2.2.1: Oppsett i Fusion 360 i arbeidsområdet DESIGN"[21].

Fusion 360 har flere egenskaper som blir fordelt innenfor ulike arbeidsområder (1). Hovedverktøylinjen (2) er der en finner essensielle verktøy som det å lage skisser, ekstrudere, avrunde hjørner, splitte kropper eller slå sammen kropper, ta mål av deler, koble komponenter og det å få komponenter til å utføre bevegelser. I applikasjonen (3) endrer en arbeidsområdet, altså bytte fil, lagre filer, lage eller åpne filer, eksportere filer og angre eller utfør likevel på handlinger.

Nedtrekksmeny (4) er en flytende meny som lister opp all data som designet ditt inneholder. Den viser alle komponenter, kropper, skisser, egenskaper og konstruksjons planene som er i designet. Arbeidsområdet hvor en ser modellen kalles for lerretet (5). For å endre aksen eller vinkelen en ser modellen fra, roterer en om 3D-modellen av lerretet (6).

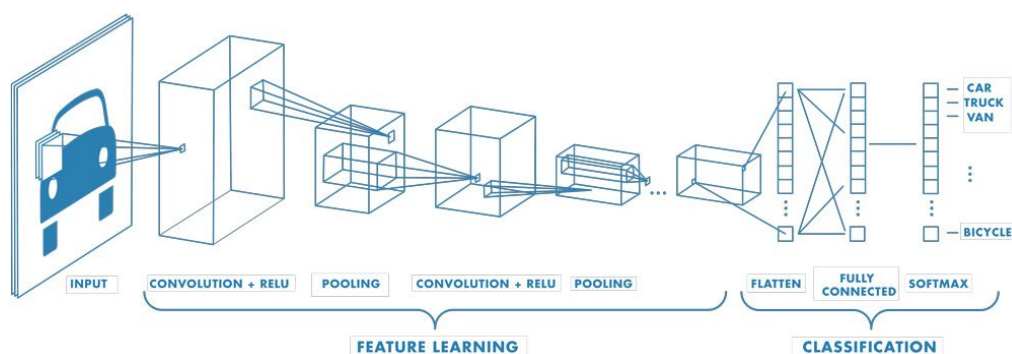
Kropper er en samling av overflater som er kombinert til å danne et enkelt objekt. Dette kan gjøres ved å skissere noe for så å ekstrudere det. Komponenter er en samling av kropper. Hver komponent har egen orientering og kan flyttes uavhengig av andre komponenter eller kropper [22].

2.2.2 Spillmotoren Unity

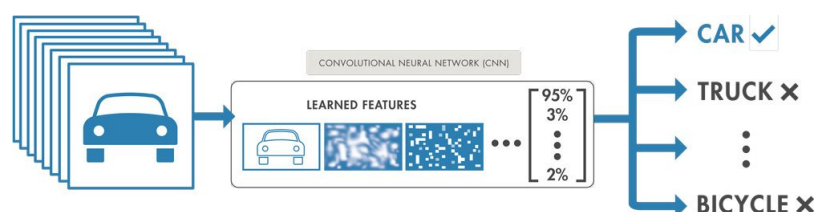
Unity en spillmotor og utviklingsplattform som kan bli brukt til å lage spill, 3D-simuleringer og interaktive opplevelser. Unity Technologies er utvikleren av Unity og utga det for første gang i 2005 [23].

2.3 Sikkerhetssone

2.3.1 Algoritmen YOLO



Figur 2.3.1: Fundamental struktur og oppbygning av et Convolutional Neural Network (CNN) [24].



Figur 2.3.2: Eksempel på klassifiseringen gjort av det neurale nettverket [24].

You Only Look Once (YOLO) er en algoritme for å detektere objekter i forskjellige klasser [25]. En klasse i denne sammenhengen er en kategorisering av objekter, som for eksempel mennesker, hunder, ryggsekker og båter. Det eksisterer uendelig mange klasser, og disse kan trenes for å detektere det en ønsker selv. Det er viktig å skille mellom de forskjellige typene datagjenkjenning. Det skilles ofte mellom tre typer: objektgjenkjenning, objekteteksjon og objektsegmentering.

YOLO bygger på prinsippet om Convolutional Neural Network (CNN), og er den maskinlæringsalgoritmen som blir brukt for bilder. Algoritmen er også utviklet basert på åpenhets-prinsippet, slik at hvem som helst kan være med på å utvikle og forbedre den [26].

CNN er en maskinlæringsalgoritme hovedsakelig brukt til å klassifisere bilder for å determinere objekter. CNN deler bildet inn i forskjellige lag, med forskjellige funksjoner. Noen modeller kan ha få lag i det skjulte laget ("hidden layer"), og andre kan ha alt fra ti til hundre lag.

Lagene som ligger i det konvolusjonelle laget har som hovedoppgave å skille mellom kanter og kontraster. De konvolusjonelle lagene lærer grovt hvordan å detektere kanter, som i mange tilfeller i denne oppgaven kommer til å være mennesker. Det neste laget lærer enda dypere og kan detektere flere kompliserte kanter og former. Totalt finnes det 24 lag i YOLO v8 modellen som blir brukt i dette prosjektet.

En kunne valgt å bruke mindre kompliserte metoder for å detektere objekter, men jo færre konvolusjonelle lag som blir brukt, desto mindre er sjansen for en korrekt detektering.

«Pooling» er mest brukt til å redusere dimensjonen av bildet. Denne prosessen betyr at alle neuronene fra de konvolusjonelle lagene blir samlet til, kall det et nytt lag. Hovedfunksjon er dermed å komprimere store mengder informasjon til mindre informasjon, og så videre. Til slutt vil en ha komprimert og dratt ut nok informasjon om den spesifikke delen i bildet, til å kjøre et neuralt nettverk på delen.

2.4 Systemarkitektur

Systemarkitekturen forklarer teorien bak forskjellige kommunikasjonsprotokoller og nevner enkelte programvarer som er brukt i oppgaven.

2.4.1 Kommunikasjon mellom ferge og terminaler

2.4.1.1 Kommunikasjonsprotokoll mellom ferge og terminaler (AFT-API)

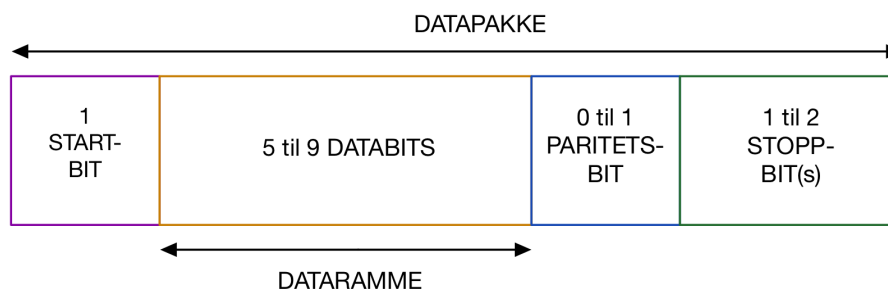
Kommunikasjonen mellom milliAmpere2 og land-terminalene er designet og bygget av en gruppe bachelorstudenter ved Institutt for Elektroniske Systemer. Denne gruppen har laget et Application Programming Interface (API), slik at datapakker som sendes mellom fergen og land-terminal går på standardisert måte. API-et heter AutonomFergeTerminal-API (AFT-API) og er utviklet sammen med denne gruppen.

2.4.1.2 Kommunikasjonsprotokoll RS-422

Recommended standard 422 (RS-422), er en kommunikasjonsprotokoll for full-dupleks kommunikasjon. Full-dupleks vil si en kommunikasjonsform hvor sender og mottaker ikke blir påvirket av hverandre. RS-422 gir mulighet for opp til 10 noder og kabellengder opp til 1200 m. En kan se på full-dupleks som telefonsamtale,

hvor partene kan snakke i munnen på hverandre, men informasjonen blir fanget opp [27].

2.4.1.3 Kommunikasjonsprotokollen UART over RS-422



Figur 2.4.1: Grafisk fremstilling av en UART-datapakke [28].

Universal Asynchronous Receiver-Transmitter (UART), er en seriell kommunikasjonsprotokoll for bitvis dataoverføring mellom to enheter. UART har to parallelle signaler: Tx som sender og Rx som mottar signalet. RS-422-standarden brukes for å oppnå robust dataoverføring i støyete miljøer. En pakke består av en startbit, dataramme, paritetsbit og stoppbits. Bitsene sendes på standardiserte transmissjonshastigheter, ofte 9600 baud eller 115200 baud. Se figur 2.4.1 for inndelingen av pakkestrukturen til UART [28].

2.4.2 Kommunikasjon på terminal

2.4.2.1 Kommunikasjonsstandarden RS-485

Recommended Standard 485 (RS-485) er en kommunikasjonsstandard som kun definerer de fysiske egenskapene til elektriske drivere og mottakere for bruk i seriell kommunikasjon. Den er på grunn av et differensielt signal mindre mottagelig for indusert støy og er derfor ofte brukt i industri. RS-485 brukes gjerne opp mot flere mottakere i type busskonfigurasjon og trenger kun en termineringsmotstand i enden på omtrent 120 ohm mellom datapinnene. Kommunikasjonen er half-duplex ettersom kun en driver kan kommunisere omgangen [29].

2.4.2.2 Nettverksprotokollen UDP/IP

User Datagram Protocol/Internet Protocol (UDP/IP) er en nettverksprotokoll for tidssensitiv data, Det er en minimalistisk metode for å overføre data uten å formelt etablere kontakt. Typiske bruksområde er video og dataspill, grunnet hastigheten den oppnår ved å ikke forsøke på ettersende gamle datapakker sammenlignet med en annen mye brukt protokoll, Transmission Control Protocol/Internet Protocol (TCP/IP) [30].

2.4.2.3 Standaren PoE

Power over Ethernet (PoE) er en standard som tillater å sende strøm og data over samme kabel. Dette muliggjør det å montere elektriske nettverkskomponenter på

steder der det ikke er elektriske strømuttak. Typiske enheter er for eksempel IP-kamera og trådløse aksesspunkt [31].

2.4.3 Kommunikasjon på ferge

2.4.3.1 Lysstyrings-standarden DMX512

Digital Multiplex (DMX) 512 er en standard kommunikasjonsprotokoll som er mest brukt for å styre scenebelysning. Navnet betyr at en kabel kan styre 512 digitale kanaler hvor hver kanal kan ha en verdi mellom 0 – 255. Hver kanal er gjerne knyttet til farger som rød, grønn og blå som igjen gjengir intensiteten på kanalen. Protokollen bygger på den fysiske standarden RS-485 for maskinvarespesifikasjonene sine [32].

2.4.3.2 Kommunikasjonsprotokollen NMEA 0183 over UDP/IP

National Marine Electronics Association 0183 (NMEA 0183) er en kombinert standard av elektrisk- og dataspesifikasjon kommunikasjon for maritimt utstyr, som Global Positioning System (GPS), sonar, kompass, radar, plotter og mer.

NMEA 0183 består av en serie med standardiserte setninger som inneholder data om ulike navigasjonsparametere. Disse setningene brukes American Standard Code (ASCII) karakterer for å representere dataen som er sendt mellom enhetene. ASCII-karakterer er brukt fordi de er lette å tolke og overføre over serielle datakabler.

NMEA 0183 er en seriell protokoll som vil si at det er flere enheter som lytter"og snakker", hvor en snakkerkan prate til mange lytteresamtidig. Kommunikasjonsstandarden er proprietær og definerer RS-422 som standard, men har utbredt seg en offentlig akseptert standard hvor en setning pakkes inn i en UDP datagram og sendes over et IP-nettverk [33].

Se tabell 2.4.1 for hvordan ASCII-setningene struktureres.

Start tegn \$ eller !	ID tag, Data, Felt separator (,)	Checksum separator (*)	Checksum	Slutt tegn <CR><LF>
--------------------------	-------------------------------------	---------------------------	----------	------------------------

Tabell 2.4.1: Oversikt over ASCII setningsstruktur.

Eksempelvis kan en ASCII-setning se slik ut: \$ID,A,0.10,0*14<CR><LF>. Den forteller at en setning startes med (\$), og sender IDen sin (ID,A,0.10,0). Deretter separeres sjekksummen og ID (*), og sier hva sjekksummen er (14) for deretter å avslutte med (<CR><LF>).

ASCII	Hex	Dec	Use
<CR>	0x0d	13	Carriage return
<LF>	0x0a	10	Line feed, end delimiter
!	0x21	33	Start of encapsulation sentence delimiter
\$	0x24	36	Start delimiter
*	0x2a	42	Checksum delimiter
,	0x2c	44	Field delimiter
\	0x5c	92	TAG block delimiter
^	0x5e	94	Code delimiter for HEX representation of ISO/IEC 8859-1 (ASCII) characters
~	0x7e	126	Reserved

Tabell 2.4.2: ASCII-tegn med tilhørende representasjon i heksadesimal og desimal, samt beskrivelse av funksjonaliteten [33].

2.4.4 Programvareverktøy brukt i systemarkitekturen

2.4.4.1 WAGOs programmeringsmiljø e!COCKPIT

e!COCKPIT er WAGO sitt integrerte utviklertmiljø for programmering, maskin-konfigurering og mer. Det kan lastes ned fra <https://www.wago.com> og krever lisens, men en 30 dagers prøvelisens kan anskaffes kostnadsfritt.

2.4.4.2 Konfigureringsverktøyet BusLink for LINAKs aktuatorer

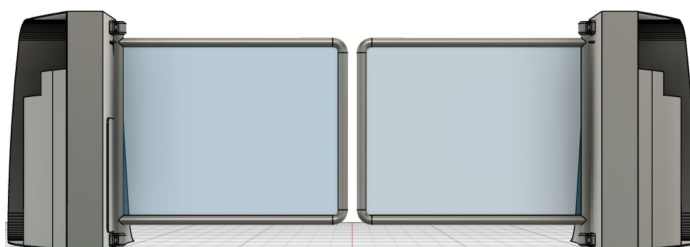
BusLink er programvaren LINAK bruker for å kontrollere aktuatorene sine med integrert kretser (IC). Her kan parametere enkelt justeres og virtuelle endestopp settes. Programvaren er gratis og kan lastes ned fra <https://www.linak.no>. En trenger egen separat adapter for å koble til aktuatorene.

Kapittel 3

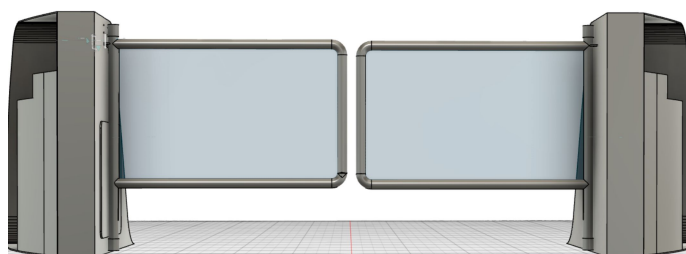
Systemkomponenter

I dette kapittelet er de ulike komponentene som til sammen utgjør passasjerhåndteringssystemet presentert. Som leser får en et innblikk i hvordan de er laget, hvorfor de endte opp akkurat slik og hva slags prosess som er gjort for å få det ferdige resultatet. Det drøftes hva som har fungert, og hva som ikke har fungert. Leseren vil få en forståelse av arbeidsprosessen frem mot resultatene.

3.1 Fergeport



Figur 3.1.1: Eksempelforslag på fergeport fra Leander.



Figur 3.1.2: Fergeport med høyere bunn.

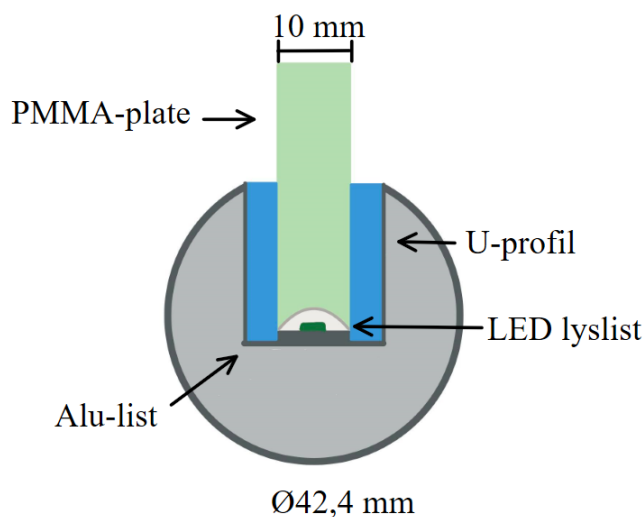
I forbindelse med fergeporten tok gruppen utgangspunkt i den ferdiglagde CAD-tegningen, vist i figur 3.1.1, av fergeporten som er designet av stipendiat Leander Spyridon Pantelatos ved Institutt for design på NTNU. Tegningen er laget i SOLIDWORKS og inkluderer baugkasse og porter. Under prosjektet gikk gruppen ut ifra denne tegningen, men gjorde noen endringer underveis for at det skulle passe visjonen til gruppen og oppdragsgivere om et sikkert passasjerhåndteringssystem.

Når en skal bygge en fergeport er det flere faktorer som det er tatt hensyn til, blant annet sikkerhet, design og teknisk utførelse.

For å sikre et langvarig produkt som tilfredsstillter oppdragsgivers visjon, har gruppen vært i jevnlig kommunikasjon med veiledere. Etter samtale med Egil Eide konkluderte gruppen at porten lå for nært dekket. En port som er for langt nede kan

føre til at porten ikke klarer å åpne eller lukke mye snø. Det helhetlige designet er også penere med en høyere port. Det ble derfor gjort designendringer i CAD-filen til Leander, hvor endringene ga porten en større avstand til dekk. Se figur 3.1.1 for å se hvordan fergeporten så ut før, og se figur 3.1.2 for å se fergeporten etter at gruppen har gjort endringer.

3.1.1 LED-lys

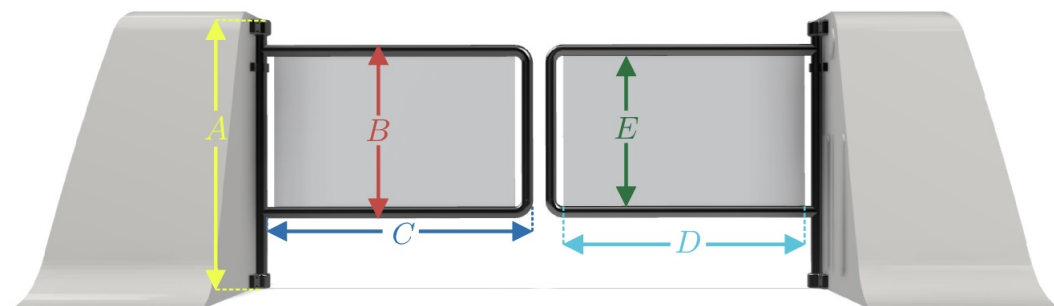


Figur 3.1.3: Rammeverk med PMMA-plate og LED-list.

På oppstartsmøtet 23.02.23 introduserte vår veileder Ole Andreas Alsos et designforslag av fergeporten. Designforslaget inkluderte statusoppdateringer fra fergen gjennom LED-lyssignal i en plastplate som har til hensikt å fordele lyset i hele porten. Dette inspirerte gruppen til at porten ikke bare skal være funksjonell, men også skulle også gi passasjerene informasjon visuelt. Se figur 3.1.3 for sammensetningen til rammeverket med rør, LED og PMMA-plate.

Gruppen valgte å benytte plastplate av typen PMMA i satinert opal med 50% transmisjon, etter anbefalinger fra plastplateleverandøren [34]. LED-listen lyser ulike farger etter hva statusen på fergen er. Ved åpen port i forbindelse med på- eller avstigning lyser porten grønt. Når porten er i bevegelse vil den blinke rødt.

3.1.2 Rammeverk



Figur 3.1.4: Fergeport med mål som er nærmere beskrevet i tabell 3.1.1.

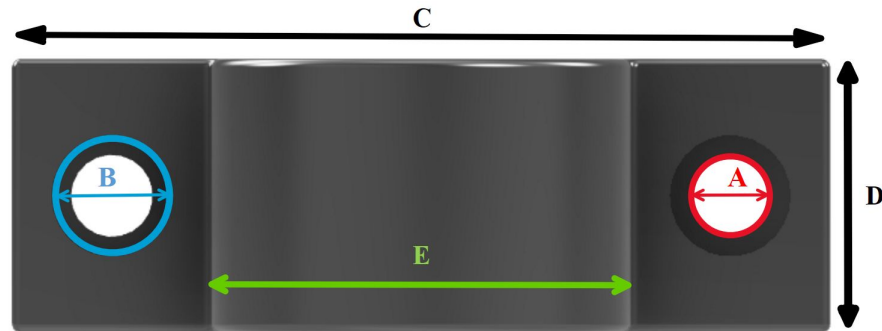
Strekning	Lengde
A	980.00 [mm]
B	60.00 [mm]
C	990.00 [mm]
D	920.00 [mm]
E	36.00 [mm]

Tabell 3.1.1: Størrelser på fergeport, se figur 3.1.4.

Ettersom gruppen ønsket LED-lister i rammen på porten er det valgt å benytte et stålrør med nedfelt U-profil, se figur 3.1.3. Det er forventet noe sjøsprøyt på porten, noe som kan føre til korrosivitet av stålet. Det er derfor valgt å bruke det rustfrie stålet av typen AISI 316. Den Amerikanske jern- og stålföreningen American Iron and Steel (AISI) er standard innen jern og stål produksjon [35]. 316-stålet har en sammensetning som gjør det mer korrosjonsdyktig enn andre ståltyper som er ofte brukt i industrien. Dette skyldes særlig dets høyere krominnholdet som da fungerer som en beskyttelsesfilm på ståloverflaten og beskytter dermed stålet mot korrosjon i tøffere miljø [36].

Både stålrøret med U-profil og det runde stålrøret er levert seks meter langt fra leverandør. Gruppen samarbeidet med ITK-verkstedet for å kappe opp, og deretter sveise sammen rørene. Rørene er kappet etter mål opplyst i tabell 3.1.1.

3.1.3 Festebrakett



Figur 3.1.5: Festebrakett med definisjoner, se tabell 3.1.2 for mål.

Strekning	Lengde
A	Ø9.00 [mm]
B	Ø13.30 [mm]
C	90.00 [mm]
D	30.00 [mm]
E	Ø43.40 [mm]

Tabell 3.1.2: Størrelser på festebrakett, se figur 3.1.4.

Det er flere forskjellige måter å feste fergeporten fast i baugkassen. En metode er å benytte aksling og kulelager. En sveiser da akslingen fast i toppen og bunnen av porten, og fester de tilhørende lagrene i baugkassen. En annen mulighet er å benytte hengsler. Dette fant gruppen mer utfordrende i form av at det krever god presisjon å sveise hengslene lineært i forhold til hverandre. Det disse to metodene har til felles er at de er mer tidkrevende å gjennomføre, i tillegg til at de krever mer hjelp fra ITK-verkstedet til mekanisk tilvirkning. Gruppen valgte derfor å 3D-printe festebraketten i stoffet PETG.

3.1.4 Portstyring

Fergeporten med dens LED-list og aktuator er styrt fra en WAGO PLS om bord på fergen. Det er Zeabuz som står bak programmeringen av autonomisystemet på fergen. Derfor arrangerte gruppen møter med dem for å finne ut hvorledes fergeportene skal implementeres til det eksisterende PLS-programmet.

3.1.5 Rotasjon av fergeport

I startfasen anskaffet prosjektgruppen CAD-tegninger av baugkassene, porter og aktuator til fergen laget av Leander. Leander lagde CAD-tegningene i SOLIDWORKS, et program som ikke er gratis for ITK-studenter. Dette resulterte i bruk av et annet CAD-program, Fusion 360, slik at gruppen kunne jobbe videre med CAD-tegningene. Overføringen fra en STEP-fil, SOLIDWORKS sitt filformat, til en f3d-fil, Fusion 360 sitt filformat, førte til at bevegelsesanimasjonene som er med i systemet forsvant.

Startet med å åpne baugkassen i arbeidsområdet *Design*", og omgjorde objektene som skulle utføre bevegelser fra kroppertil komponenter". Lagde så et mekanisk relasjon mellom komponentene som skulle utføre bevegelser ved bruk av verktøyet "joint".

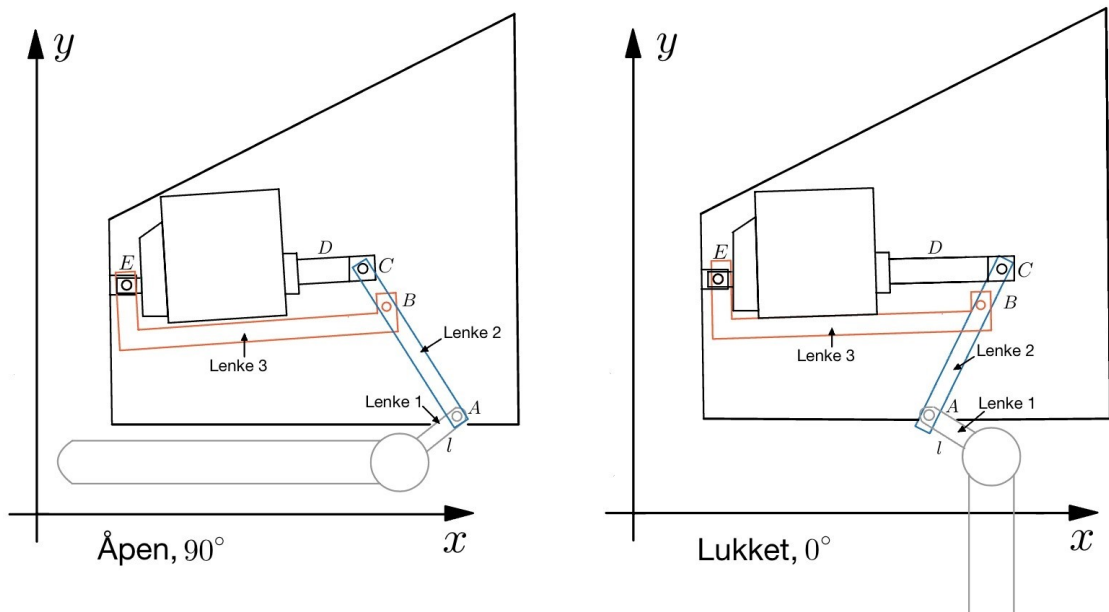
Komponent 1	Komponent 2	Bevegelse	Beskrivelse
Port	Brakett oppe og nede	Rotasjon	Port roterer om z -aksen
Endepunkt aktuator	Koblingspunkt på vegg	Rotasjon	Aktuator roter om koblingspunktet i z -aksen
Stang ut fra aktuator	Aktuator	Prismatisk	Stangen har en lineær bevegelse ut og inn fra aktuatoren langs z -aksen
Koblingsstang fra port	Stang ut fra aktuator	Rotasjon	Kobler den roterende porten og den prismatiske aktuatoren sammen

Tabell 3.1.3: Hvordan lenker og ledd er koblet sammen.

Det ble en konflikt mellom baugkassen og port ved testing av CAD-tegningen med bevegelsene fra tabell 3.1.3. Når aktuatoren hadde full slaglengde gikk koblingsstangen fra porten inn i nødstoppen som er på baugkassen. Slaglenden til aktuatoren i tegningen er 50 mm lengre enn den faktiske slaglengden til aktuatoren som skulle brukes. Gruppen kontaktet derfor Leander om disse problemene og satt i gang problemløsning.

Når det gjelder problemløsning er hovedfokuset å lage løsninger som er robust og som ikke skaper uønskede situasjoner. Testet ut flere løsninger, men aktuator og koblingslenke klarte ikke å rotere porten smidig 90° alene. Det måtte derfor tilføres nye komponenter til systemet.

3.1.5.1 Lenker og ledd



Figur 3.1.6: Porten sine posisjoner med 90° og 0° .

Det er brukt inverskinematikk for å finne en teoretisk løsning hvor aktuatoren klarer å rotere portene fra 90° til 0° . Tok hensyn til restriksjoner som at arbeidsområdet systemet opererte på er begrenset, det er bevegelsen fra aktuatoren som får porten til å rotere og slaglengden til aktuatoren er 100 mm.

Deretter ble det skissert opp en tegning av hvor lenkene og leddene burde ligge for å løse problemstillingen med bruk av inverskinematikk. Videre ble det benyttet inverskinematikk for å finne ut hvilken vinkling de burde ha for at den prismatiske bevegelsen fra aktuatoren skulle klare å rotere end-effektor 90° . Se skisse i figur 3.1.6.

Ledd A , B , C og D er roterende ledd som er koblet fast til lenker. Leddene roterte om z -aksen. Ledd C har to frihetsgrader: Et roterende ledd og et prismatisk ledd med en slaglengde på 0 – 100 mm. Det er totalt fire ledd i systemet, A , B , C og D , se posisjonen i figur 3.1.6.

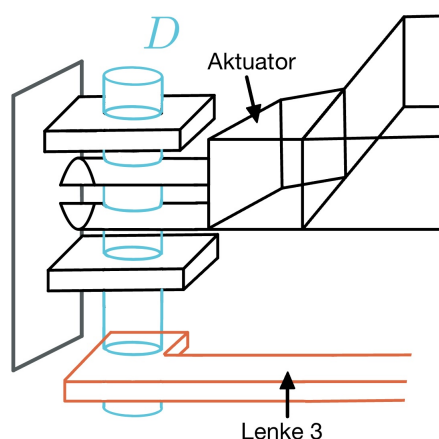
Lenke 1 er stangen som er koblet fast til porten. Lenke 2 er koblingsleddet mellom aktuator og lenke 1. Lenke 3 er leddet som er koblet til samme ledd som aktuatoren. I figur 3.1.6 vil disse fargene gjelde for de ulike lenkene, lenke 1 er grå, lenke 2 er blå og lenke 3 er oransje.

Har brukt teorien bak *enkel aksel-mekanisme* så la derfor til en koblingslenke, lenke 2, mellom lenke 1 og aktuatoren sitt prismatiske ledd, ledd D . Dette gjorde at den lineære bevegelsen kunne omgjøres til en rotasjonsbevegelse.

Ledd B , som både er et roterende- og prismatisk ledd, roterte i z -aksen samtidig som den sklei langs x -aksen, og har derfor en frihetsgrad for mye som gjør systemet over-aktuert. La derfor til lenke 3 som vil holde lenke 2 fra å skli langs x -aksen, og da heller rotere om x -aksen. Når lenke 2 roterte ville lenke 1 følge etter og porten

ville da rotere, se figur 2.1.4. Lenke 3 gjorde systemet til en lukket kinematisk kjede som betydde at kraftpåvirkningen kom fra aktuatoren som da også styrte systemet.

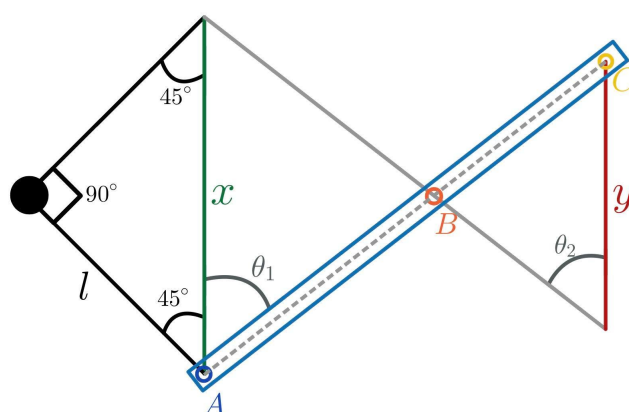
3.1.5.2 Kobling til ledd D



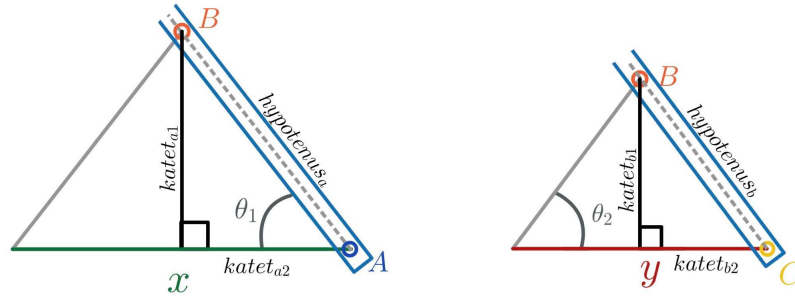
Figur 3.1.7: Aktuator og lenke 3 koblet til ledd D .

Ledd D er et rotasjonsledd som da gjorde det mulig for aktuatoren å bevege seg langs x -aksen med ledd A . Når porten lukket seg skulle ledd D og A øke avstanden fra x -aksen til porten sto 45° lukket. Deretter, når porten beveget seg fra 45° til 0° , trakk leddene seg mot x -aksen. Aktuatoren måtte samarbeide med lenke 3, og de roterte derfor sammen. Se figur 3.1.7, for å se hvordan aktuator og ledd 3 er koblet til ledd D .

3.1.5.3 Utregning av lengder og vinkler



Figur 3.1.8: Arbeidsområdet til lenke 2 som da roterer langs x og y linjene.



Figur 3.1.9: Likebeinte trekanter som er delt i to. Bruker Pytagoras' læresetning for å finne lengdene til sidene.

Lengden til lenke 1 er $l = 69.77$ mm. Vinklingen til lenke 1 på porten er 45° når porten er åpen og 45° når den er lukket. Totalt skaper dette arbeidsområdet en likebeint trekant, figur 3.1.8.

Bruker trigonometri, Pytagoras' læresetning, og finner ut hvor lang x -linjen er:

$$l^2 + l^2 = x^2 \quad (3.1)$$

$$x = l \cdot \sqrt{2}. \quad (3.2)$$

Ved å sette inn verdien til l får man da at $x = 98.67$ mm lang. I denne sammenhengen er x en linje som ledd A sklir langs. Hvor på x -linjen ledd A ligger er avhengig av hva slags kraft den får fra aktuatoren.

Lenke 1 er koblet til lenke 2 som står $45^\circ + \theta_1$ grader på hverandre når porten er åpen. For at ledd A skal klare å skli langs x -linjen når porten åpner eller lukker seg, er det tre koblingsledd, A , B og C . Når ledd C , altså slaglengden, øker skal ledd A skli oppover x -linjen for å rotere porten.

De ulike posisjonene til leddene på lenke 2 skaper to likebeinte trekanter, figur 3.1.8. Bruker derfor Pytagoras' læresetning for å finne ut hvor de ulike leddene må ligge på lenke 2 for at porten skal klare å gå fra 90° til 0° . Må også finne ut hva slaglengden til linje y er.

Bestemmer forholdet mellom de to likebeinte trekantene, a og b og fikk da ut konstanten $\frac{5}{3}$.

Trekantene, a og b , er identiske, men har ulik skalering. Konstanten $\frac{a}{b}$ er forholdet mellom de to trekantene sine to like sider, og forholdet mellom grunnlinjene, x og y er $\frac{x}{y}$. Uttrykkte dette som

$$\frac{a}{b} = \frac{x}{y}. \quad (3.3)$$

Setter inn verdiene til a , b og x og bruker dette forholdet til å finne verdien til y som er 59.22 mm.

Trekker en linje tvers gjennom de likebeinte trekantene fra ledd B og til grunnlinjene x og y , se figur 3.1.9. Den totale avstanden på denne linjen er satt på 100 mm.

Bruker Pytagoras' læresetning for å finne ut hva avstanden mellom punkt A og B er i trekant a , se figur 3.1.9. Finner først ut hva verdien til $Katet_{a1}$ (K_{a1}) er med bruk av uttrykket

$$K_{a1} = a \cdot 100. \quad (3.4)$$

Setter inn verdi a og får at K_{a1} er 62.5 mm lang.

Finner ut hva verdien til $Katet_{a2}$ (K_{a2}) med bruk av uttrykket,

$$K_{a2} = \frac{x}{2} \quad (3.5)$$

Setter inn x sin verdi og får at K_{a2} er 49.35 mm lang.

Setter opp uttrykket for $Hypotenus_a$ (H_a),

$$H_a = \sqrt{K_{a1}^2 + K_{a2}^2}. \quad (3.6)$$

Setter inn verdiene til K_{a1} og K_{a2} og får ut at H_a , avstanden mellom ledd A og B , er 79.6 mm lang.

Bruker den inverse trigonometriske funksjonen \tan^{-1} for å finne vinklingen lenke 2 skal ha på lenke 1, θ_1 ,

$$\theta_1 = \tan^{-1}\left(\frac{K_{a1}}{K_{a2}}\right). \quad (3.7)$$

Vinkelen lenke 2 skal ha på lenke 1, θ_1 , er $51.7^\circ + 45^\circ$ som er 96.7° .

Bruker også her Pytagoras' læresetning for å finne avstanden mellom ledd B og C i trekant b , se figur 3.1.9. Finner først ut hva verdien til $Katet_{b1}$ (K_{b1}) er med bruk av uttrykket

$$K_{b1} = b \cdot 100. \quad (3.8)$$

Setter inn b sin verdi og får at K_{b1} er 37.5 mm lang.

Finner ut hva verdien til $Katet_{b2}$ (K_{b2}) med bruk av uttrykket,

$$Katet_{b2} = \frac{y}{2}. \quad (3.9)$$

Setter inn y sin verdi og får at $Katet_{b2}$ er 29.61 mm lang.

Setter opp uttrykket for *Hypotenus_b* (H_b),

$$H_b = \sqrt{K_{b1}^2 + K_{b2}^2}. \quad (3.10)$$

Setter inn verdiene til K_{b1} og K_{b2} og får ut at H_b , avstanden mellom ledd B og C , er 47.78 mm.

Bruker den inverse trigonometriske funksjonen \tan^{-1} for å finne vinklingen lenke 2 skal ha på aktuatoren, θ_2 ,

$$\theta_2 = \tan^{-1}\left(\frac{K_{b1}}{K_{b2}}\right). \quad (3.11)$$

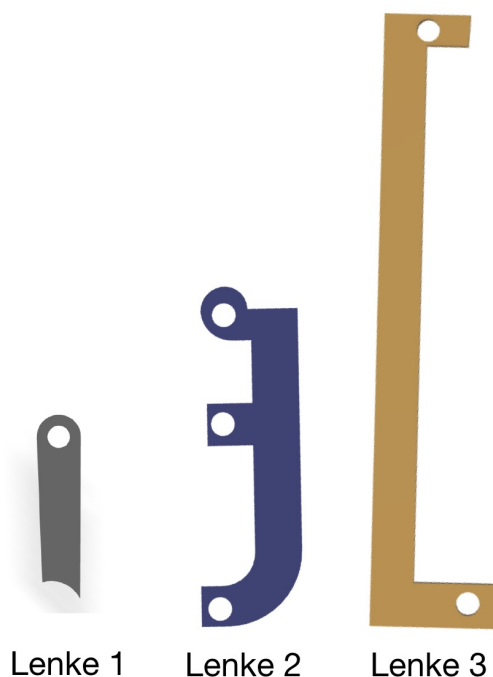
Vinklingen lenke 2 skal ha på aktuatoren, θ_2 , er $51.7^\circ + 45^\circ$ som er 96.7° .

Se tabell 3.1.4 for oppsummering av hvordan de ulike leddene skal være plassert og hva slags vinkling den må ha på lenkene den er koblet sammen med.

Fra ledd	Til ledd	Lengde	Vinkelen
A	B	79.6 [mm]	På lenke 1, 96.7°
B	C	47.78 [mm]	På aktuator, 96.7°

Tabell 3.1.4: Lenke 2 sine mål for avstand mellom ledd samt vinkling den må ha på lenkene den er koblet til.

3.1.5.4 Fusion 360-simulering



Figur 3.1.10: Lenke 1, 2 og 3 laget i Fusion 360.

Lenke 1 eksisterte allerede i tegningen som Leander hadde laget, og det eneste som måtte endres var lengden til 69.77 mm. Gjorde dette med bruk av verktøyet *ekstrudere*". Lenke 2 og 3 eksisterte ikke fra før av så bitene måtte først skissere opp etter utregnede mål og ekstrudert bitene for å gjøre skissene om til kropper. Se figur 3.1.6 for skisse av lenkene. Omgjorde lenkene fra kropper til komponenter slik at de kan utføre bevegelser.

I figur 3.1.10 er lenkene markert i samme farge som figur 3.1.6 for å lettere kunne observere sammenheng mellom delene i Fusion 360 og de teoretiske skissen.

Lenke 2 er ikke helt lik det som var skissert opp, men fungerer på samme måte grunnet plassering av hvor leddene skal være, hull. For at ledd B skal kunne rotere uten å krasje med lenke 2 må enden ha et rundt skaft.

Leddene som er på lenkene er sylindriske og går tvers gjennom lenkene via de opp-tegnede hullene. Det er et lite mellomrom fra lenken til sylindrene slik at de kunne rotere fritt. Sylindrene sine kropper er festet inn i samme komponent som leddet det hørte til. Dette gjør at kroppene henger og beveger seg sammen. Sylindrene må rotere om egen z -akse og har derfor blitt gjort om til egne komponenter inni komponentene, altså under-komponenter.

Komponent	Kropp	Under-komponent
1.1	Lenke 1	Ingen
1.2	Lenke 2	Ledd (sylinder) A
1.3	Lenke 3	Ledd (sylinder) B og D
1.4	Slagarm aktuator	Ledd (sylinder) B

Tabell 3.1.5: Komponenters innhold av lenker og ledd.

For å teste ut om aktuatorene klarte å rotere porten 90° måtte de leddene som skulle bevege seg kobles sammen. Sammenkoblingen tar utgangspunkt i posisjonen til porten når den er åpen, 90° . Ved sammenkobling flytter lenkene på seg, så er viktig å passe på at vinklingen mellom lenke 1 og 2 er 96.7° . For å teste ut om det teoretiske fungerer i praksis via simuleringer er det viktig å være konsekvent på at alt må gjøres etter skisseringen.

En bruker verktøyet *"joint"* for å koble lenkene sammen. Koblingen på de tre første sammenkoblingene er lik første gang systemet var koblet sammen, figur 3.1.3.

Komponent 1	Komponent 2	Bevegelse	Beskrivelse
Port	Brakett oppe og nede	Rotasjon	Port roterer om z -aksen
Endepunkt til aktuator	Koblingspunkt til vegg	Rotasjon	Aktuator roter om koblingspunktet i z -aksen
Stang ut fra aktuator	Selve aktuatoren	Prismatisk	Stangen har en lineær bevegelse ut og inn fra aktuatoren langs z -aksen
Ledd D	Endepunkt til aktuator	Rotasjon	Ledd D roterer sammen med aktuator
Lenke 2	Ledd C	Rotasjon	Lenke 2 roterer om ledd C
Ledd C	Stangen til aktuatoren	Rotasjon	Ledd C roterer på stangen til aktuatoren
Lenke 2	Ledd A	Rotasjon	Lenke 2 roterer om ledd A
Ledd A	Lenke 1	Rotasjon	Ledd A roterer om lenke 1
Ledd B	Lenke 2	Rotasjon	Ledd B roterer om lenke 2
Ledd B	Lenke 3	Rotasjon	Ledd B roterer om lenke 3

Tabell 3.1.6: Hvordan ulike lenker og ledd er koblet sammen med beskrivelse.

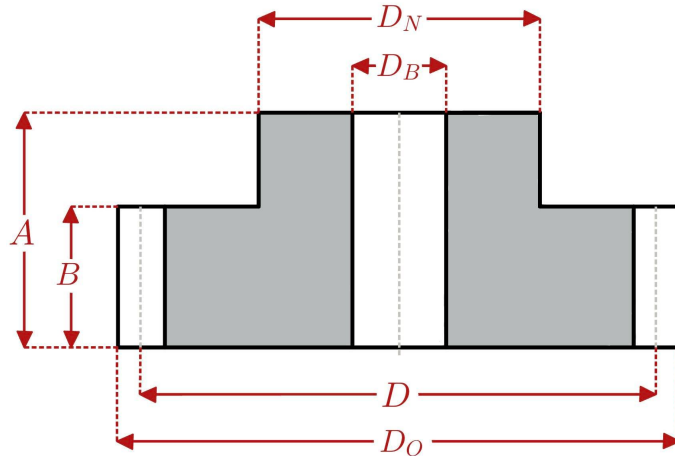
Ledd D er under-komponent til lenke 3 som vil si at når ledd D roterer så vil lenke 3 følge etter. Meningen bak denne koblingen er å få lenke 3 og ledd D til å rotere i sammen med aktuatoren.

Lenkenes rotasjon er bestemt av leddenes utslag. De sylindrerformede leddene er plassert på hullene som tidligere har blitt utregnet. For å kunne få lenken til å rotere etter leddene sine bevegelser må sylindrene rotere om egen akse, derav "*Ledd (...) roterer på lenke (...)*".

3.1.5.5 3D-utskrift

Printet ut 3D-objekter som var laget i Fusion 360. Overførte Fusion 360-filene til en 3D-print-programvare og printet ut ønskede biter. Satt så sammen bitene med aktuatoren testet det fysiske systemet.

3.1.6 Tannhjul



Figur 3.1.11: Tegning av tannhjul sammen med mål. Se tabell 3.1.7 for størrelser.

Tannhjul	Antall tenner (Z)	D_N	D	D_N	D_B
1	28	45.0 mm	42.0 mm	30.0 mm	12.0 mm
2	60	93.0 mm	90.0 mm	60.0 mm	15.0 mm

Tabell 3.1.7: Tannhulets mål fra firmaet Jens S. Transmisjoner AS sine dokumenter [37].

Den andre løsningen for å få portene til å rotere var å bruke tannhjul. Gruppen valgte å bruke standard tannhjul i håp om at de da ville ankomme hurtigere. Tannhjul 1, er montert fast i porten for å få porten til å rotere. Rotasjonen til tannhjul 1 er påført av kraften til det andre tannhjulet, tannhjul 2. Tannhjul 2 fikk den påførte kraften fra aktuatoren ettersom de to leddene er montert fast til hverandre.

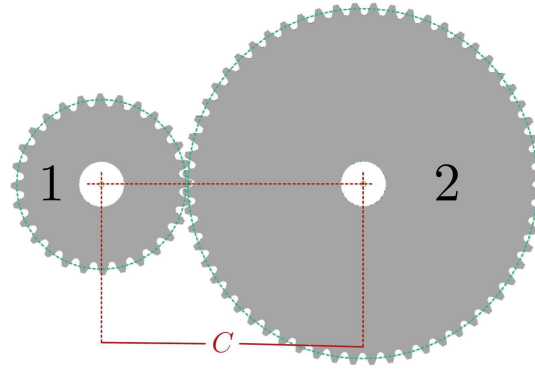
Tegningen i figur 3.1.11 er laget med inspirasjon fra firmaet Jens S. Transmisjoner AS sine sylindriske tannhjul manual [37]. Begge tannhjulene er fra modul 1 og hadde en tykkelse på 30 mm, A . Navet, $A - B$, på tannhjulet er 13 mm tykt og tannhjulet uten nav, B , er 17 mm tykt.

3.1.6.1 Plassering av tannhjul

Det er viktig å sette tannhjulene rett i forhold til hverandre for å unngå slitasje og eventuell tidlig ødeleggelse på tannhjulene. Laget først begge tannhjul om samme akse og trakk de fra hverandre med senterdistansen, C , som er 69.79 mm. Se figur 3.1.12 for hvordan tannhjulene treffer hverandre og overfører kraft.

For å finne forholdet mellom de to tannhjulene deler man tannhjul 2 sitt antall tenner (Z_2) på tannhjul 1 sitt antall tenner (Z_1)

$$\frac{Z_2}{Z_1} \quad (3.12)$$



Figur 3.1.12: Tannhjul 1 og 2.

Deretter setter inn antall tenner hvert tannhjul har, Z_1 og Z_2 og får at forholdet mellom de to tannhjulene er $1 : 2.143$. Kraften som er påført tannhjulene er konstant som gjør at tannhjul 1 rotere mer enn dobbelt så fort som tannhjul 2. Dette vil si at tannhjul 2 må rotere 42° for at tannhjul 1 skal rotere 90° . Samme gjaldt for hastighet. Tannhjul 2 roterte dobbelt så fort som tannhjul 1, men da i motsatt retning.

For å bestemme rotasjonen som et tannhjul må ha for å treffe tannmellomrommet til det andre tannhjulet brukes stigningsdiameter

$$\frac{Z}{Modul} \quad (3.13)$$

Tannhjulene er fra samme modul, modul 1, men har ulikt antall tenner (Z). Tannhjul 1 har en stigningsdiameter på 60 d og tannhjul 2 har en stigningsdiameter på 28 d.

Trenger bare den relative forskjellen i omkretsen mellom tennene, og bruker derfor forskjellen i stigningsdiameteren til de to tannhjulene. Den er på 32 d.

Konverterer denne forskjellen om til rotasjonsgrader ettersom en vil ha ut hvor mange grader tannhjulet skal rotere. For å konvertere må en finne ut den sirkulære stigningen, p

$$\frac{\pi}{Modul} \quad (3.14)$$

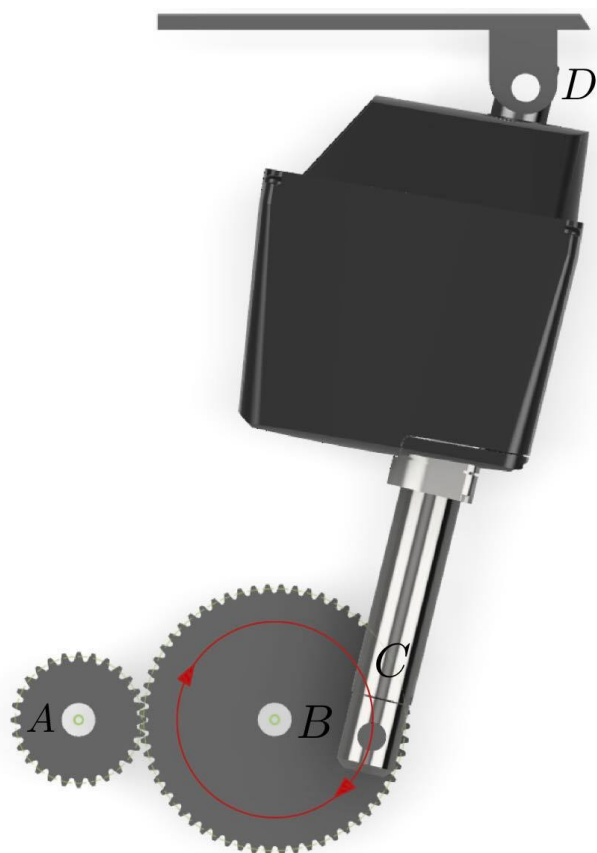
Siden tannhjulene er fra modul 1 blir den sirkulære stigningen på π p.

En kan nå regne ut hvor mange rotasjonsgrader tannhjul 1 må rotere for å treffe mellomrommet til tannhjul 2. Tar da å deler på forskjellen i stigningsdiameter (d) og den sirkulære stigningen (p)

$$\frac{d}{p}. \quad (3.15)$$

Får da ut at tannhjul 1 må rotere 10.1836° for å treffe mellomrommet til tannhjul 2.

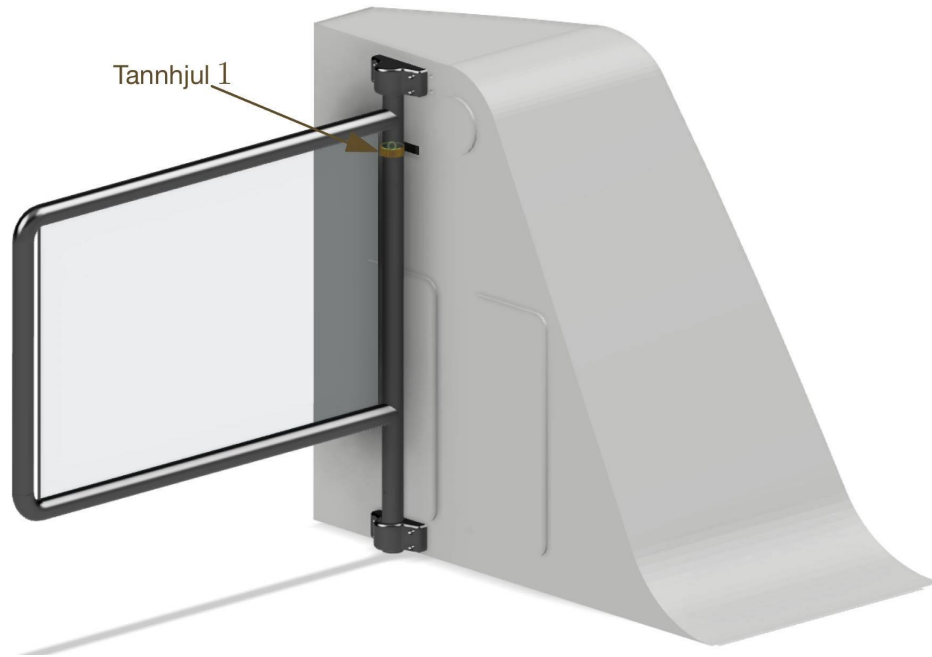
3.1.6.2 Sammenkobling til aktuator



Figur 3.1.13: Bevegelsen til ledd C på tannhjul 2.

Ledd C , leddet som bestemmer slaglengden til aktuatoren, er et prismatisk ledd. Ledd D , leddet som aktuatoren var koblet til, er et rotasjonsledd, figur 3.1.13. De to leddene hadde prismatisk- og rotasjonsbevegelse som gjorde det mulig for ledd C å følge den røde banen som fulgte rotasjonen til ledd B om z -aksen. Radiusen til den røde sirkelen var 34.91 mm.

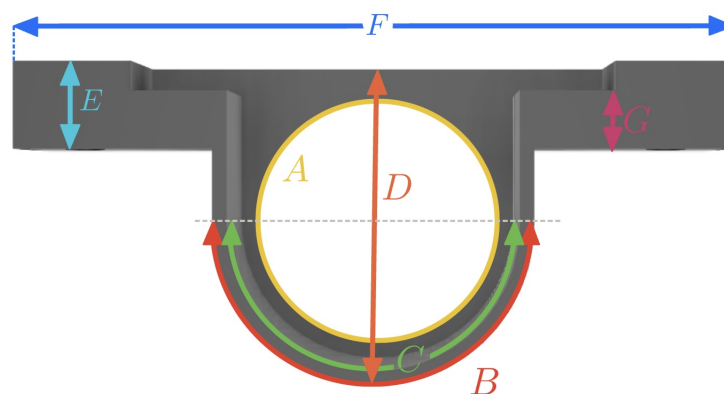
Ledd C er koblet til en stang som er fastmontert i tannhjul 2 og når aktuatoren økte slaglengden roterte tannhjul 2 om ledd B med en radius på 34.91 mm.



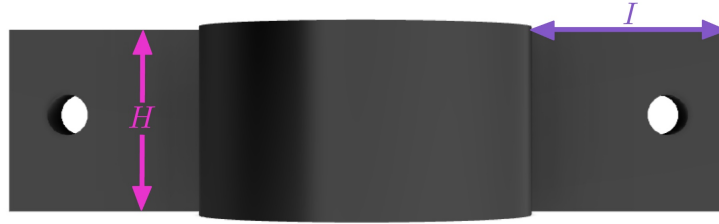
Figur 3.1.14: Tannhjul 1 montert fast i port.

Tannhjul 1 er koblet til porten slik at porten følger tannhjulets rotasjoner. Se sammenkoblingen av port og tannhjul 1 i figur 3.1.14. På grunn av dette forholdet roterte porten når aktuatoren, ledd C , økte slaglengden som da fikk tannhjul 2 til å rotere. Når slaglengden økte lukket porten seg og når aktuatoren trakk til seg slaglengden åpnet porten seg. Ledd C trengte en slaglengde på 22.54 mm for å rotere tannhjul 2 42° og tannhjul 1 90° .

3.1.6.3 Deksel



Figur 3.1.15: Dekselet sett fra undersiden, for mål se tabell 3.1.8.



Figur 3.1.16: Deksel foran, for mål se tabell 3.1.8.

Strekning	Lengde/radius
A	22.00 [mm]
B	27.50 [mm]
C	25.00 [mm]
D	57.50 [mm]
E	17.14 [mm]
F	123.00 [mm]
G	10.00 [mm]
H	31.00 [mm]
I	34.00 [mm]

Tabell 3.1.8: Mål til deksel.

For at det skal være kontakt mellom de to tannhjulene måtte tannhjul 2 stikke litt ut fra baugkassen. Åpningen som da skulle kuttes ut i baugkassen ville vært på høyde med nødstoppen og med en bredde på 32 mm.

Roterende tannhjul utstikkende fra baugkassen fører til en stor risiko for klemfare. Kan derfor ikke bruke denne løsningen uten deksel eller beskyttelse. Lagde derfor et deksel som dekket hele tannhjul 1 og den delen av tannhjul 2 som stakk ut fra baugkassen.

Designet dekselet til å stå fast i selve baugkassen slik at porten kunne rotere fritt.

Ettersom tannhjulene kommer til å rotere på en flate vil oppleve friksjon mellom flaten og undersiden til tannhjulet. For å redusere denne friksjonen, og da øke levetiden til tannhjulet, må tannhjulene smøres jevnlig med olje eller fett.

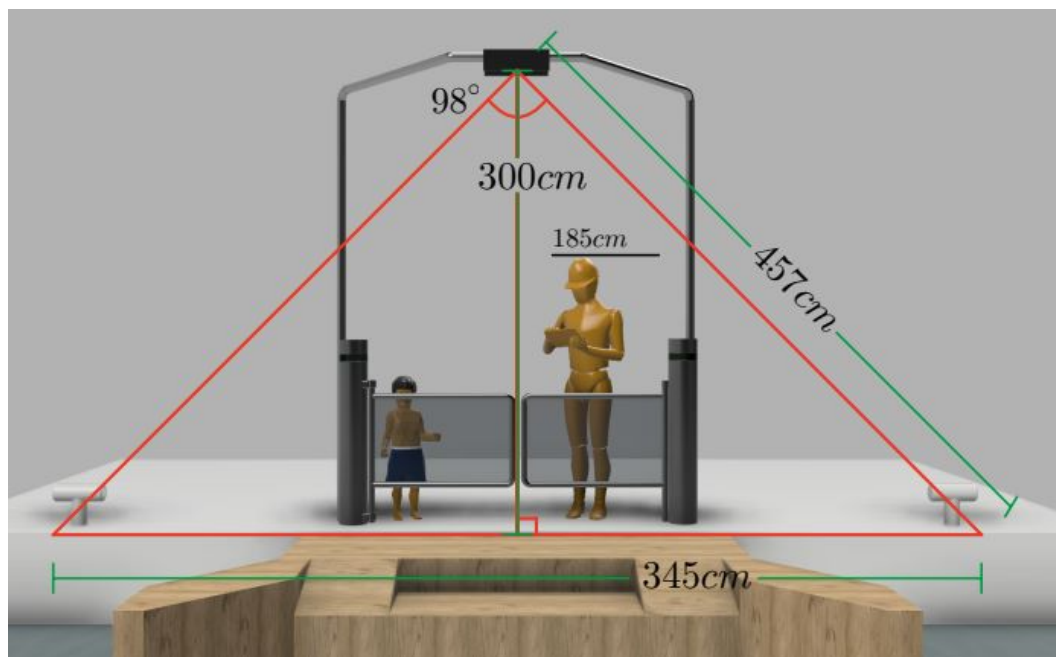
3.2 Landport



Figur 3.2.1: Landport bestilt fra Kina. Bildet er sendt fra produsent. Grønt lys indikerer at porten er åpen og det er klart for å gå av eller på fergen [8].

Landporten er bestilt ferdig bygd fra leverandør [8]. Under planleggingen av fergeporten erfarte gruppen at det er for tidskrevende å bygge enda en port fra bunnen av. Kombinasjonen av planlegging, materialinnkjøp og konstruering av port tar forholdsmessig lang tid. Bestilte derfor en ferdig bygd port fra leverandør, og tiden som ellers hadde blitt brukt på land porten kunne brukes på andre områder. Landporten kan styres enten gjennom 24 VDC eller RS-485. Dette gjøres fra PC-en i el-skabet på land.

3.3 Sikkerhetszone

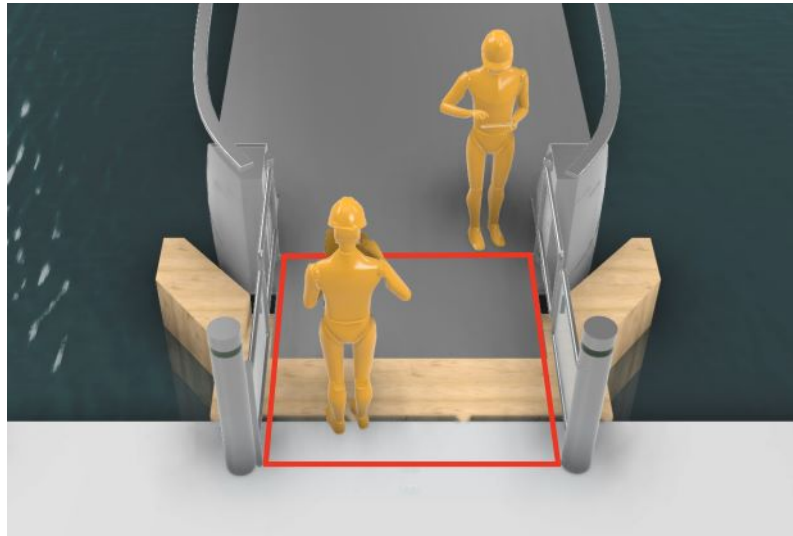


Figur 3.3.1: Beregning av høyde for sikkerhetsbøyle.

For å detektere objekter og mennesker i sikkerhetssonen er det benyttet et kuppelkamera. Kameraet har en vidvinkel linse som gjør det ideelt for bred observasjon. Med et deteksjonsområde på 98° må kameraet stå en viss avstand over plattformen, for å fange opp området, det passer uansett for å få et fint visuelt inntrykk. Slik som bildet i figur 3.3.1 viser, kan en observere beregningen av synsvinkelen og en fornuftig høyde innsatt med en person på 185 cm for å demonstrere.

Etter samtale med designer og konseptutvikler Einar Johan Hareide [38], fikk gruppen bekreftet at denne måte å løse bøylene på var intuitiv og en fin tilnærming. Grunnen til at stolpen går helt over, er det visuelle inntrykket av "inngang til ferge". Stolpen har også fleksibilitet til å eventuelt henge opp en skjerm, og fungere som en informasjonstavle for passasjerer.

3.3.1 Deteksjon av objekter i sikkerhetssonen



Figur 3.3.2: Eksempelbilde av sikkerhetssonen markert med rødt.

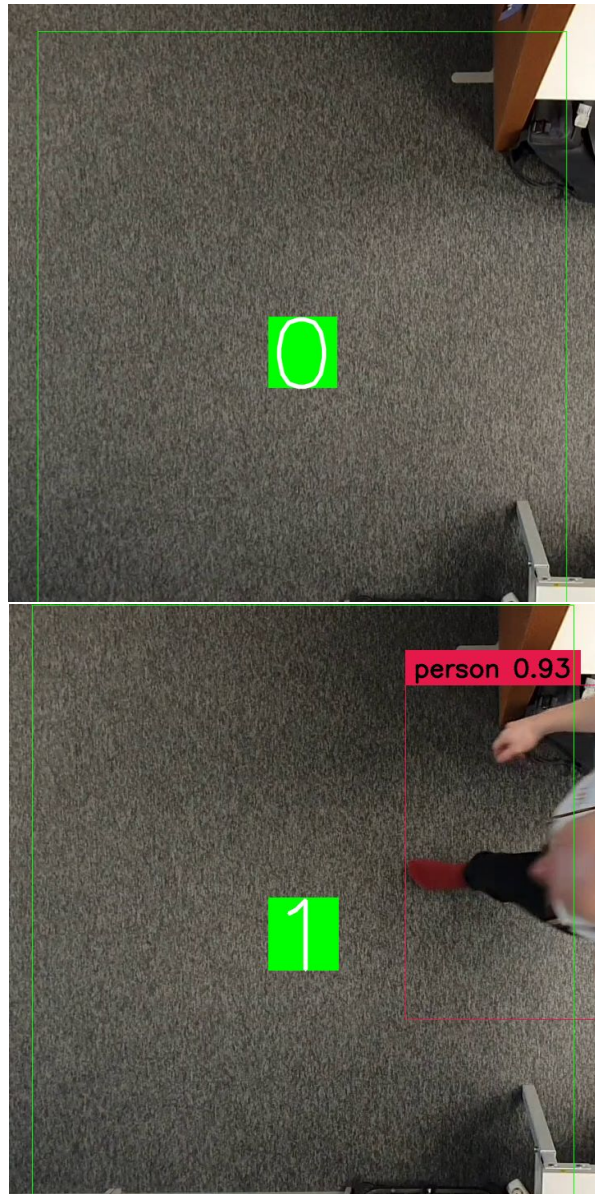
Det ønskes å unngå situasjoner hvor det er stor klemfare og fare for at objekter ligger mellom kai og ferge. I denne oppgaven er sikkerhetssonen definert som området mellom portene på båten, og portene på land, som vist i figur 3.3.2.

Denne sonen er definert sammen med oppdragsgiver for å gi et best mulig estimat av hvor den skal være.

For å kunne detektere personer eller objekter som måtte befinne seg i området har YOLOv8 blitt implementert som deteksjonsalgoritme. Algoritmen i seg selv har som oppgave å detektere objekter, og merke disse med spesifikk klasse. YOLOv8 finnes i fem forskjellige typer, ekstra stor, stor, medium, liten og nano. Disse er med på å bestemme nøyaktighet og hvor hurtig den klarer å gjenkjenne objekter. Ekstra stor er for eksempel mer nøyaktig enn nano, men går bruker mer datakraft og er tregere. Nano er da raskere, men mindre nøyaktig.

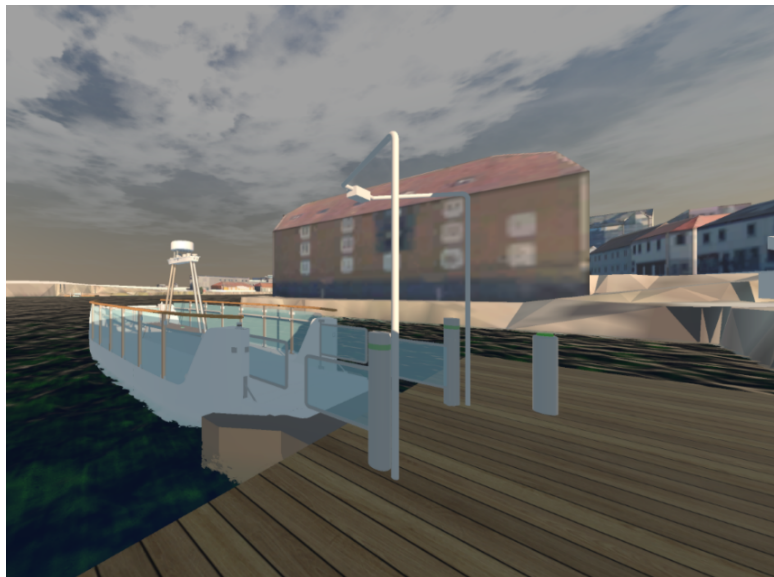
Ettersom soneinndelingen blir løst direkte på DSBOX-NX2 [39], er det ikke nødvendig å bruke egne pakker for dette. Informasjonen kan da hentes ut ved hjelp av python bindings [40].

Områdedeteksjonen gir tilbakemelding på hvor mange som er innenfor spesifisert sone. Denne sonen blir opprett med et numpy array, og er kort forklart en matrise av hvilke punkter i xy-aksene som skal brukes. Et eksempel på hvordan denne kan løses kan sees i figur 3.3.3. I eksempelet er det satt et vilkårlig område, ment for å teste ut deteksjonen.



Figur 3.3.3: Eksempelbilde av deteksjon uten objekt, og deteksjon med 1 person i sone.

3.4 Mixed Reality Laboratory (MR-lab)



Figur 3.4.1: Hva man ser i VR-verdenen når man skal gå om bord i fergen.

NTNU stilte til rådighet med en MR-lab som gruppen brukte for å teste ut brukervennligheten til systemet. Den hybride testingen bestod av en Virtual Reality (VR) verden av milliAmpere2 som driver med skytteltrafikk mellom Ravnkloa og Fosenkaia. Her var det implementert lyd, måker, kajaker og sjø for å gi brukere en reell opplevelse. I tillegg til VR-verden var det en 1 : 1 fysisk versjon av milliAmpere2 i trevirke. Som bruker kan en da få den visuelle og fysiske opplevelsen av å ta fergen, og det er nesten som å ta den virkelige fergen fra Ravnkloa og til Fosenkaia.

Passasjerhåndteringssystemet er testet ut i denne MR-labben slik at gruppen kunne få en oversikt over hvordan det var å være passasjer på fergen med de nye komponentene. Robin Stokke, ansatt i Zeabuz som autonomingeniør, har vært med på å lage den originale VR-verdenen som en del av hans sommerjobb hos NTNU i 2022. Robin overførte de nye komponentene til den originale VR-verdenen samt la ved bevegelser til de nye delene. Se figur 3.4.1 for å observere hvordan fergen ser ut med de nye komponentene i VR-verdenen.

3.5 Landkontroll

Styring og kontroll på landside blir gjort ved hjelp av en PC, type DSBOX-NX2 fra Forecr.io [39]. En slik PC er designet og utviklet for å arbeide under krevende omstendigheter og skal tåle bruken den blir utsatt for i denne oppgaven. Til styring av portene og generelle utgangssignaler benyttes en I/O-modul fra Advantech, USB-5830 [41]. Denne I/O-modulen tar også inn inngangssignaler, i dette tilfellet fra hurtigstoppen i overgangen mellom portene og knapper for manuell styring av porter og lem. For å programmere denne har leverandøren en rekke kompatible programmeringsspråk, og derfor er Python benyttet. For enkelhets skyld er det

programmert en logikk for å konvertere det heksadesimale sifferet som blir lest av fra I/O-kortet, om til et desimaltall fra 0 – 15 som symboliserer inngangen eller utgangen. Dette har kosmetisk ingen praktisk betydning, men er relativt mye lettere å forholde seg til under programmering.

PC-en som drifter dette systemet er bygget opp på et Linux-basert operativsystem som heter Ubuntu 18.04.5 og utviklet for kunstig intelligens. Valget av denne typen falt naturlig da denne er spesielt designet og utviklet for å kjøre raske analyser av videoer og bilder. Leverandøren av denne PC-en har et bibliotek med manualer som gjør det enkelt å sette opp for denne oppgaven [39]. Standard oppsettsmetode gir passord og brukernavn som vist i tabell 3.5.1.

Brukernavn	Passord
nvidia	nvidia

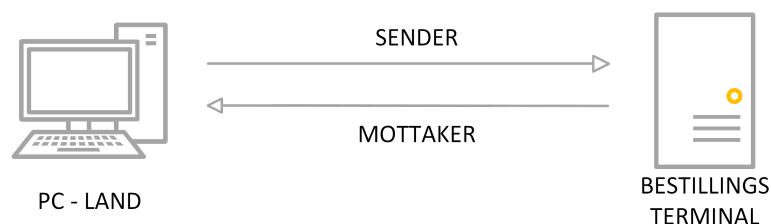
Tabell 3.5.1: Brukernavn og passord for PC på land.

I dette prosjektet har Python blitt benyttet, ettersom installasjonen av pakker og Integrated Development Environment (IDE) på PC er enkelt å bruke. Det har blitt installert en del biblioteker for å gjøre programmeringen sømløs. Bruk av eksisterende biblioteker sparer en for mye tid da en slipper å utvikle kommandoer og koder som allerede finnes.

Forfatter	Pakke	Kilde
Advantech	Automation.BDaq	[41]
Ultralytics	YOLOv8	[25]
Ukjent	PySerial	[42]

Tabell 3.5.2: Python-bibliotek.

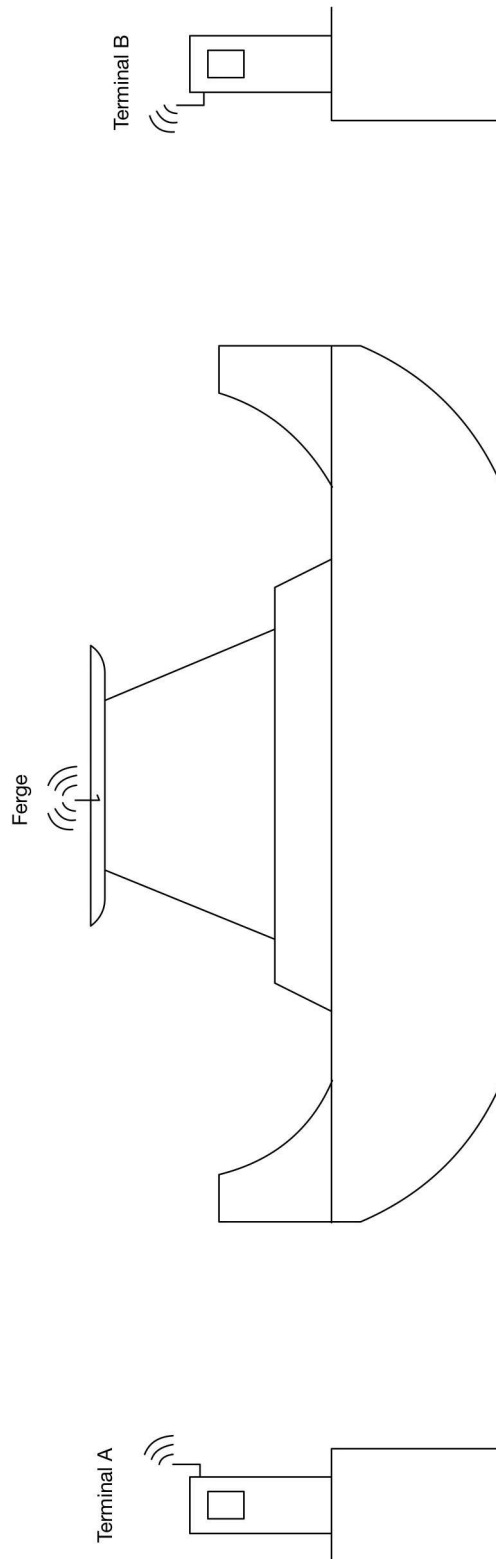
3.5.1 Kommunikasjon mellom PC og landterminal



Figur 3.5.1: Eksempel på kommunikasjon mellom land-PC og bestillingsterminalen.

Kommunikasjon mellom PC på land og bestillingsterminal har blitt bestemt å kjøre på seriell kommunikasjon. Det vil si at det går kabler fra PC til bestillingsterminal, som gjør det mulig å kommunisere mellom stasjonene i sanntid. For å programmere dette på PC-en er PySerial [42] benyttet. PySerial benytter seg av RS-422-kommunikasjon.

3.6 Systemarkitektur for ferge og terminaler



Figur 3.6.1: Konseptskisse for kommunikasjon mellom ferge og terminaler.

Denne seksjonen gir oversikt over de forskjellige kommunikasjonselementene i passasjerhåndteringssystemet, hvordan de er satt sammen og hva som skal til for å få sendt meldinger gjennom de. Systemet er satt sammen av tre hovedmoduler: fergen, terminalene og kommunikasjonen mellom dem, se figur 3.6.1. I tabell 3.6.1 ligger de mest sentrale systemkomponentene.

Ferge:

Utstyr	Type	Datablad
AutoPax-PC	DSBOX-N2	[43]
I/O-maskin	750-8212	[44]
Aktuator	LA25	[45]
DMX 512-driver	SLC S26005	[46]
Høytaler	FR 8 WP	[47]
Forsterker	EW2-MS	[48]
Eksternt lydkort	PLAY!3	[49]

Terminal:

Utstyr	Type	Datablad
Porter	Mairs MT311	[50]
PC	DSBOX-NX2	[51]
I/O-kort	USB-5830	[52]
Svitj	Ha-VIS eCon 2000	[53]
Skumringsrelé	IC2000	[54]
Kamera	TVIP44510	[55]

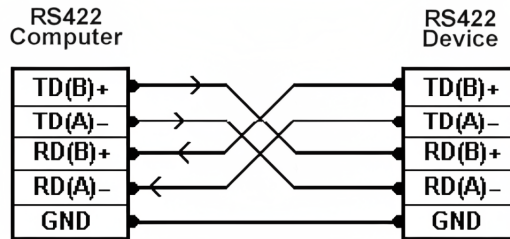
Tabell 3.6.1: Enkel komponentliste for ferge og terminal.

3.6.1 Fergeterminal

I parallel med passasjerhåndteringssystemet utviklet en gruppe ved Institutt for Elektroniske Systemer (IES) et trådløst kommunikasjonssystem, Trådløs terminal for «on-demand» bestilling av autonom ferge", for enkelhetens skyld er denne omtalt videre i oppgaven som Autonom fergeterminal (AFT). Oppgaven deres er transport av data mellom fergen og landterminalene, se deres bacheloroppgave for mer informasjon [4]. For å kunne bruke systemet er det opprettet et Application Programming Interface (API) eller programmeringsgrensesnitt med forhåndsbestemte kodeord for bestemte handlinger. Til sammen omtales dette AFT-API [4].

3.6.1.1 Tilkobling

Kommunikasjonssystemet AFT-API bruker kabling og signal standarden RS-422 for oppkobling og signalutveksling med andre enheter i full-duplex. Den tar i bruk UART protokollen som transportør for meldingene. Koblingen av RS-422 utføres i henhold til figur 3.6.2.



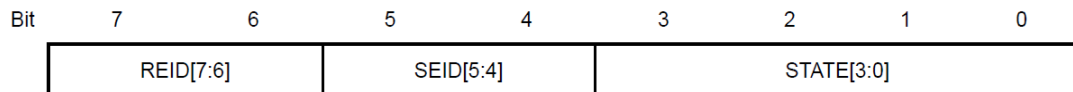
Figur 3.6.2: Oppkobling av RS-422 [56].

For at to enheter skal kunne snakke sammen med bruk av UART må de være enige om hvordan en melding utveksles, som når den starter og slutter. Hvordan AFT-API-et forventer å motta meldinger er beskrevet i tabell 3.6.2.

Navn	Data	Type
Data hastighet	115200	Baudrate
Data bits	8	Bits
Data kontroll	N	Parity
Data slutt	1	Stop bits

Tabell 3.6.2: AFT-API konfigurering.

3.6.1.2 Meldingsutveksling



Figur 3.6.3: Eksempel på AFT-API dataramme 5.2.7.

Meldingene som sendes gjennom AFT-API er alle innpakket i en UART-pakke, disse er definert av AFT-API-et til å ha en dataramme på 1 Byte. Datarammen som er vist i figur 3.6.3, er da strukturert slik at de 4 Most Significant Bit (MSB) er identifikatorer brukt for å indikere sender og mottaker. De 4 Least Significant Bitsene (LSB) indikerer handlingen som er etterspurt, utført eller, alternativt indikerer en tilstand.

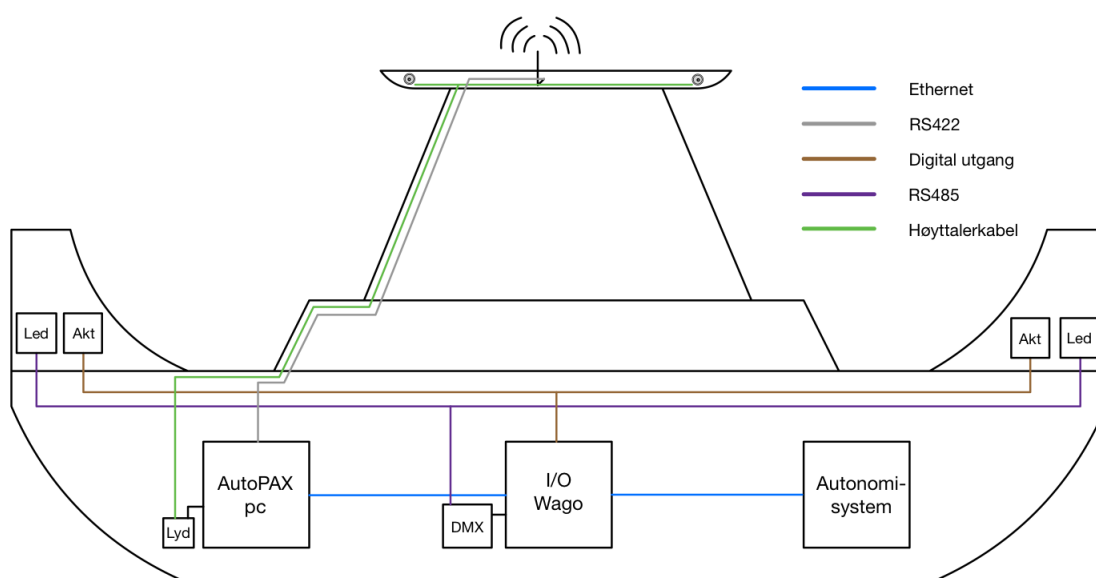
Meldingene som sendes på AFT-API-et er forhåndsbestemt i henhold til dokumentasjonen utarbeidet av IES-gruppen, dette er gjort i samsvar med tilbakemeldinger fra denne oppgaven og hele dokumentet ligger i vedlegg **D - AFT-API grensesnitt og kommandoer**. Meldingsflyten er laget på konseptet etterspørsel/respons, hvor for eksempel fergen etterspør tilstanden på landportene, deretter svarer terminalen med statusen. Noen av meldingene vil på grunn av kapasitet begrensning kun agere etter egen etterspørsel, som for eksempel en trykk knapp for manuell styring av portene, og krever derfor ingen respons. Et utdrag på mulige responser på status av landporter er vist i figur 3.6.4. Om porten er åpen og sendt fra terminal A til fergen vil dette akkumulere til Byte-en 0xD2 representert på formen heksadesimal.

Bit 3:0 - STATE[3:0] Tilstand på port
 Dette bit-feltet definerer tilstanden på port ved avsender-ID.

Verdi	Navn	Beskrivelse
0x1	TIMEOUT	Gate timeout
0x2	OPEN	Gate open
0x3	CLOSE	Gate closed
0x4	PAUSE	Gate pause

Figur 3.6.4: AFT-API portstatus eksempel 5.2.7.

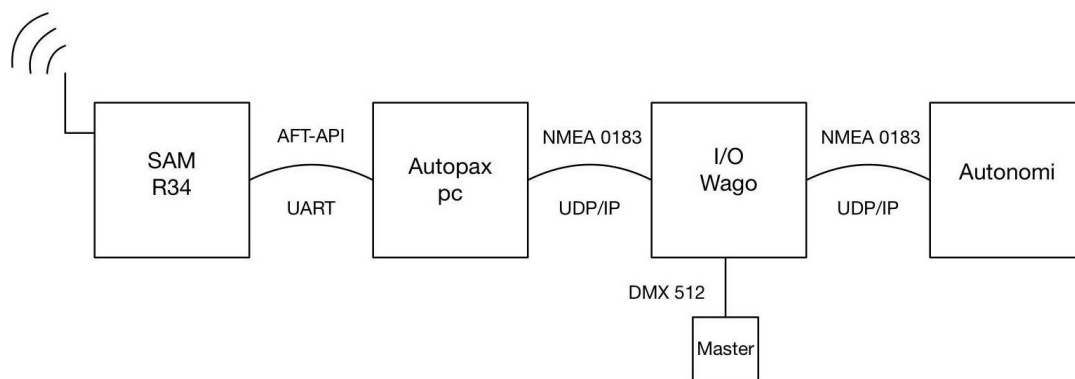
3.6.2 Ferge



Figur 3.6.5: Systemarkitektur for ferge.

Ombord i fergen legger passasjerhåndteringssystemet tilrette for at systemet skal integreres på en separat PC kalt AutoPAX-PC-en. Maskinen som i dag kjører fergen, autonomisystemet, befinner seg ombord i fergen. Figur 3.6.5 gir en grafisk fremstilling av hva som inngår i passasjerhåndteringssystemene på fergen.

I masten er antennen til AFT-kommunikasjonen og et 4-punkt høyttalersystem. For å kunne styre sikkerhetsportene og systemet rundt brukes en ferdig montert programmerbar I/O-maskin. I/O-maskinen utfører sekvenslogikken, mottar kommandoer direkte fra autonomimaskinen og videresender nødvendige beskjeder videre til AutoPAX-PC-en.



Figur 3.6.6: Kommunikasjonsflyt for ferge.

Meldingsflyten mellom de forskjellige enhetene ombord er illustrert i figur 3.6.6. Her vises det hvordan autonomimaskinen kan sende ut en melding til I/O-maskinen over standarden NMEA 0183 som en UDP-pakke med IP protokollen og Ethernet som transportmediet. Dette signalet går via de forskjellige enhetene i forskjellig innpakning og sendes over til terminalene via mikrokontrolleren SAM R34.

3.6.2.1 AutoPAX-PC



Figur 3.6.7: Grensesnitt til PC [57].

Funksjonen til AutoPAX-PC-en inn mot passasjerhåndteringssystemene er å være enheten ombord som håndterer de interaktive systemene mot passasjerene. Foreløpig er den kun benyttet for å rute data mellom AFT-API-et og I/O-maskinen. AutoPAX-PC-en har også som formål å formidle auditiv informasjon gjennom et Public Address (PA)-system via et eksternt lydkort. PC-en er en Jetson DSNBOX-N2 [43] med 4 GB ram og 250 GB minne, med en integrert serielldriver for RS-422 og I/O-pinner for håndtering av digitale signaler. Den benytter fergens 24 VDC kraftforsyning til å driftes og kan enkelt monteres på DIN-skinne med et allerede installert feste.

3.6.2.2 Autonomimaskinen

Hjernen til milliAmpere2 er autonomimaskinen. Den tar avgjørelsene som angår fergens drift og det vil være denne maskinen som etterspør AutoPAX-systemet om å åpne/lukke sikkerhetsportene når fergen har lagt til eller fra terminalen.

Autonomisystemet benytter allerede en meldingsløsning med I/O-maskinen i dag for å styre lukene som heter NMEA 0183 over UDP. Meldingsstrukturen fra autonomimaskinen er definert som i figur 3.6.8. En komplett oversikt over meningsutvekslingene finnes i vedlegg **E - NMEA0183 kommando og status meldinger lukestyring**.

`$WAGOHATCH,<ID>,<command>*ch`

Figur 3.6.8: NMEA kommando melding fra autonomimaskinen.

3.6.2.3 WAGO I/O-maskin

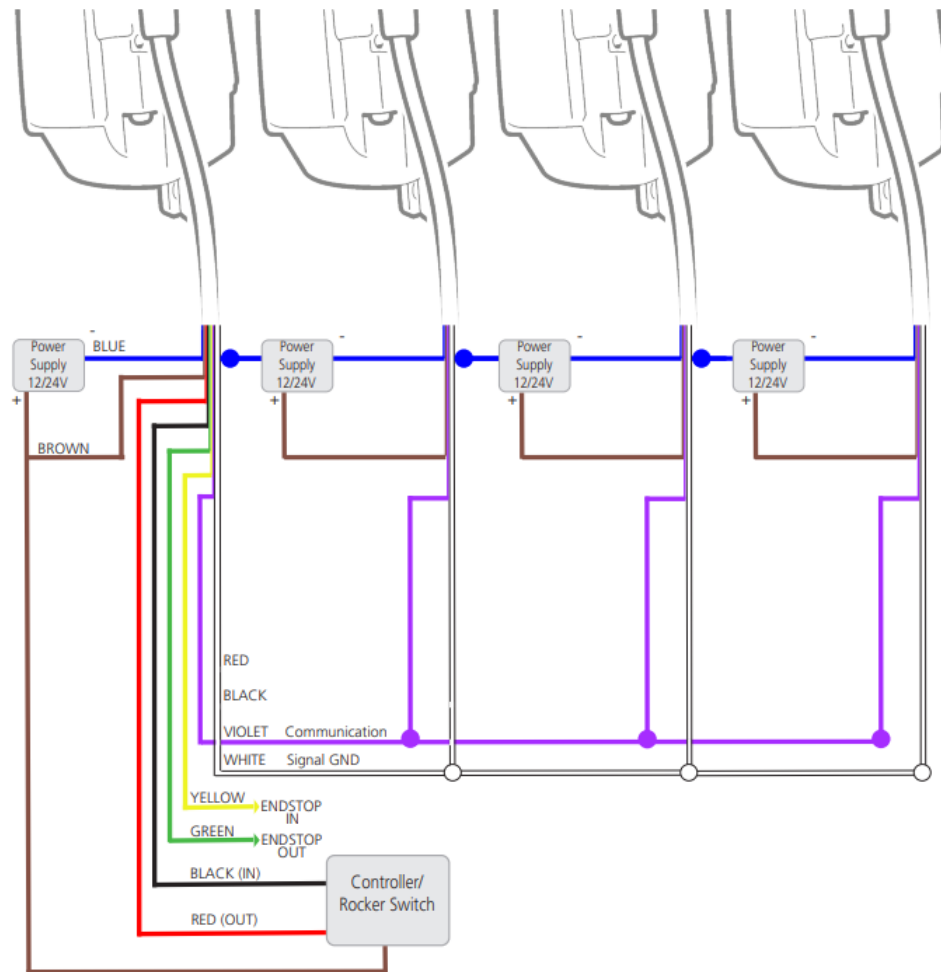
For håndtering av sikkerhetsportene og annet periferiutstyr er I/O-maskinen 750-8212 [44] til WAGO brukt ombord. Dette er en modulbasert maskin som kan settes sammen av flere forskjellige moduler etter behov. Denne maskinen har installerte moduler i henhold til tabell 3.6.3 En oversikt over hva som er tilkoblet modulene finnes i vedlegg **J - I/O liste - WAGO I/O-maskin**.

Modul	Modell	Tilkoblinger
Seriell driver	750-652	1
Digitale innganger	750-1405	16
Digitale utganger	750-1504	16
Analoge innganger	750-455	4
Analoge innganger	750-471	4
Analoge utganger	750-564	4
Endemodul	750-600	-

Tabell 3.6.3: WAGO-moduler montert på I/O-maskinen.

3.6.2.4 Lineær aktuator

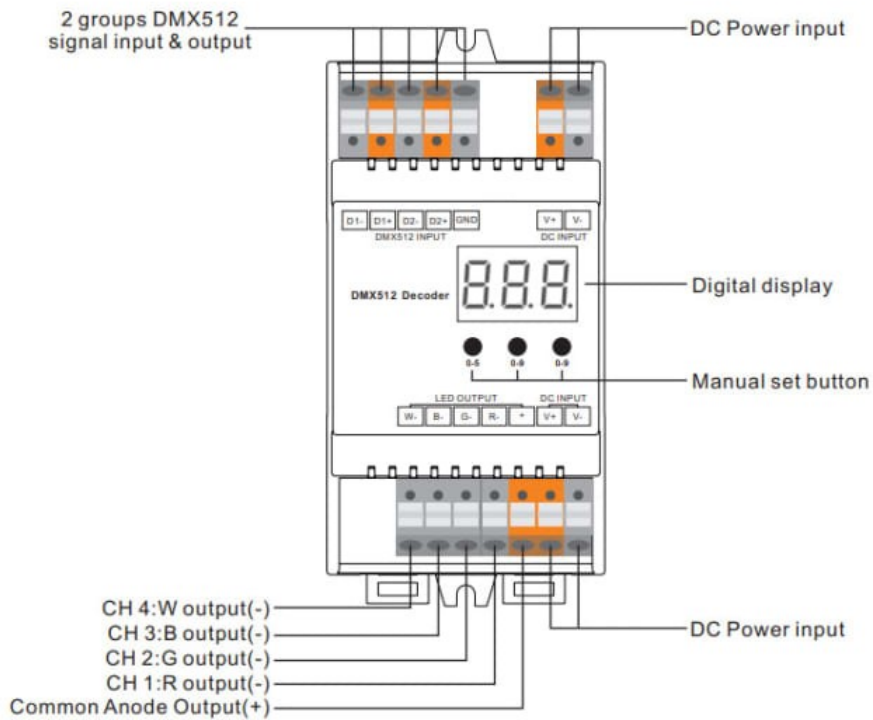
Portene er styrt av to lineær aktuatorer av typen LA25 [45] laget av LINAK. De er begge utstyrt med en funksjon som er kalt parallell. Denne funksjonen gjør at aktuatorene samkjøres selv ved ujevn belastning. Hvis en av de ikke er i lik posisjon, korrigeres denne forskjellen ved neste inn eller ut kjøring. Ved oppkobling av aktuatorene gjøres dette i henhold til figur 3.1.7. For inn og ut kjøring mates en spenning inn på korresponderende inngang helt til den ønskede posisjon er nådd.



Figur 3.6.9: Oppkobling til aktuator. Bilde er tatt fra databladet til LINAK [58].

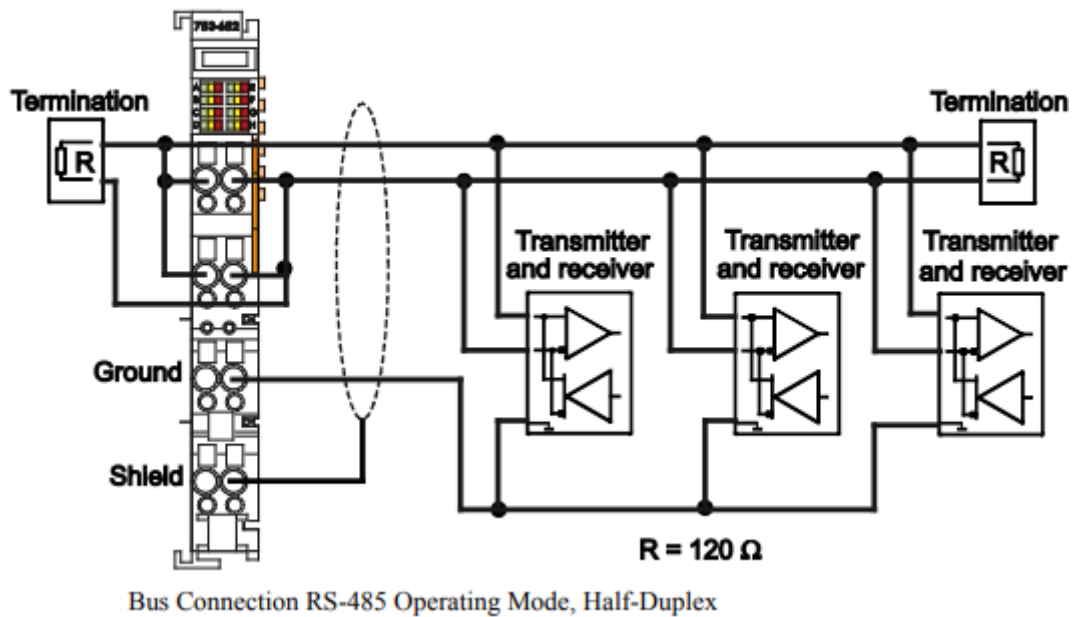
3.6.2.5 DMX-driver

På sikkerhetsportene monteres det LED-striper for visuell signalisering. Disse er drevet av en 4x-kanal (RGBW) DMX512 driver per portpar. Driveren er en SLC S26005 [46] og har 4x5A (1A per kanal), 480 W totalt og har en driftsspenning på 24 VDC. En oversikt over tilkoblingsmulighetene er illustrert i figur 3.6.10.



Figur 3.6.10: Oversiktsbilde av DMX512-driveren [59].

RS-485 Operating Mode



Figur 3.6.11: Oppkobling av DMX-til-seriell-modul tilhørende I/O-maskinen [60].

Driveren kobles til via den serielle modulen på I/O-maskinen, se figur 3.6.11 og DMX-adressene må konfigureres via det fysiske grensesnittet illustrert i figur 3.6.10. Hvordan man kobler den opp er beskrevet i instruksjonen [59]. DMX-adressene er fordelt i henhold til tabellen 3.6.4.

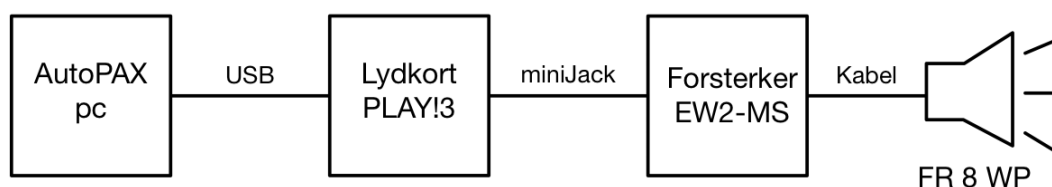
Kanal		DMX-adresse
Fore port	R	1
	G	2
	B	3
	W	4
Aft port	R	5
	G	6
	B	7
	W	8

Tabell 3.6.4: DMX-adresser for tilhørende kanaler.

3.6.2.6 Nødstopp og reflektor

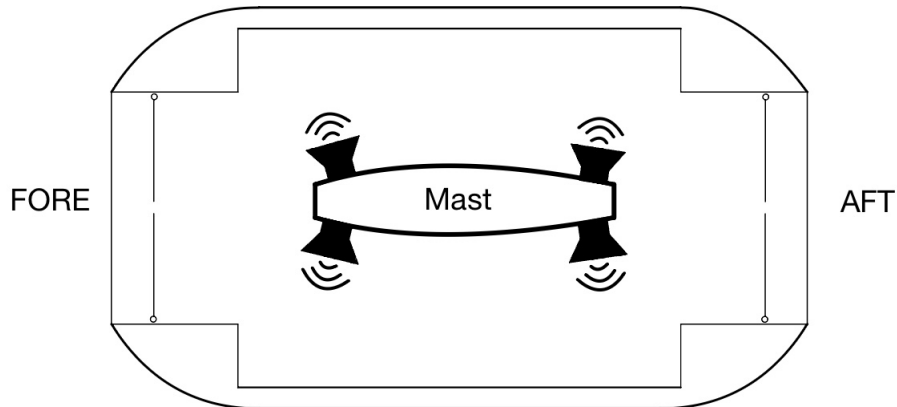
Fergen er allerede utstyrt med en nødstopp til begge lukene. Disse er koblet til et relé som forrigler styresignalet til aktuatorene og tilbakemelding til I/O-maskinen. Det er også prosjektert å utruste sikkerhetsportene med hver sin infrarøde reflektor for deteksjon av passasjerer som passerer gjennom eller står for nærme risikoområdet.

3.6.2.7 Lydkort – høyttalere



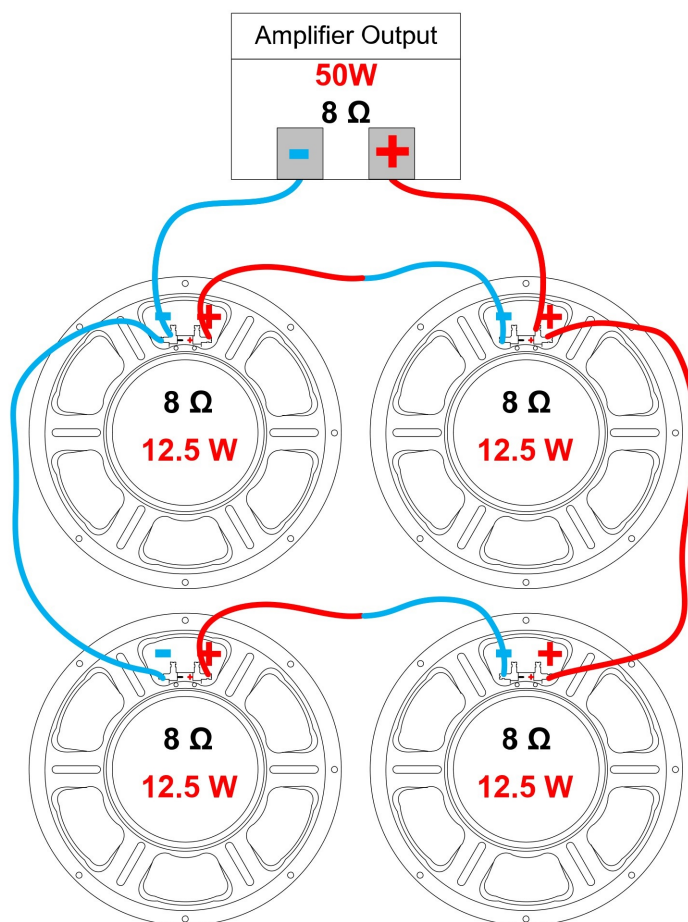
Figur 3.6.12: Oversikt av høyttalersystem.

I masten skal det monteres fire små høyttalere av typen Visaton FR 8 WP [47]. Høyttalerene skal forsterke den intuitive forståelsen av fergens beslutninger gjennom å opplyse om hendelser enten gjennom muntlige beskjeder eller tydelige signaler. Høyttaleren har en diameter på elementene på 4", en motstand på 8 ohm og er effekt-merket med 15 W.



Figur 3.6.13: Konseptskisse for plassering av høyttalerene på fergen, sett ovenfra.

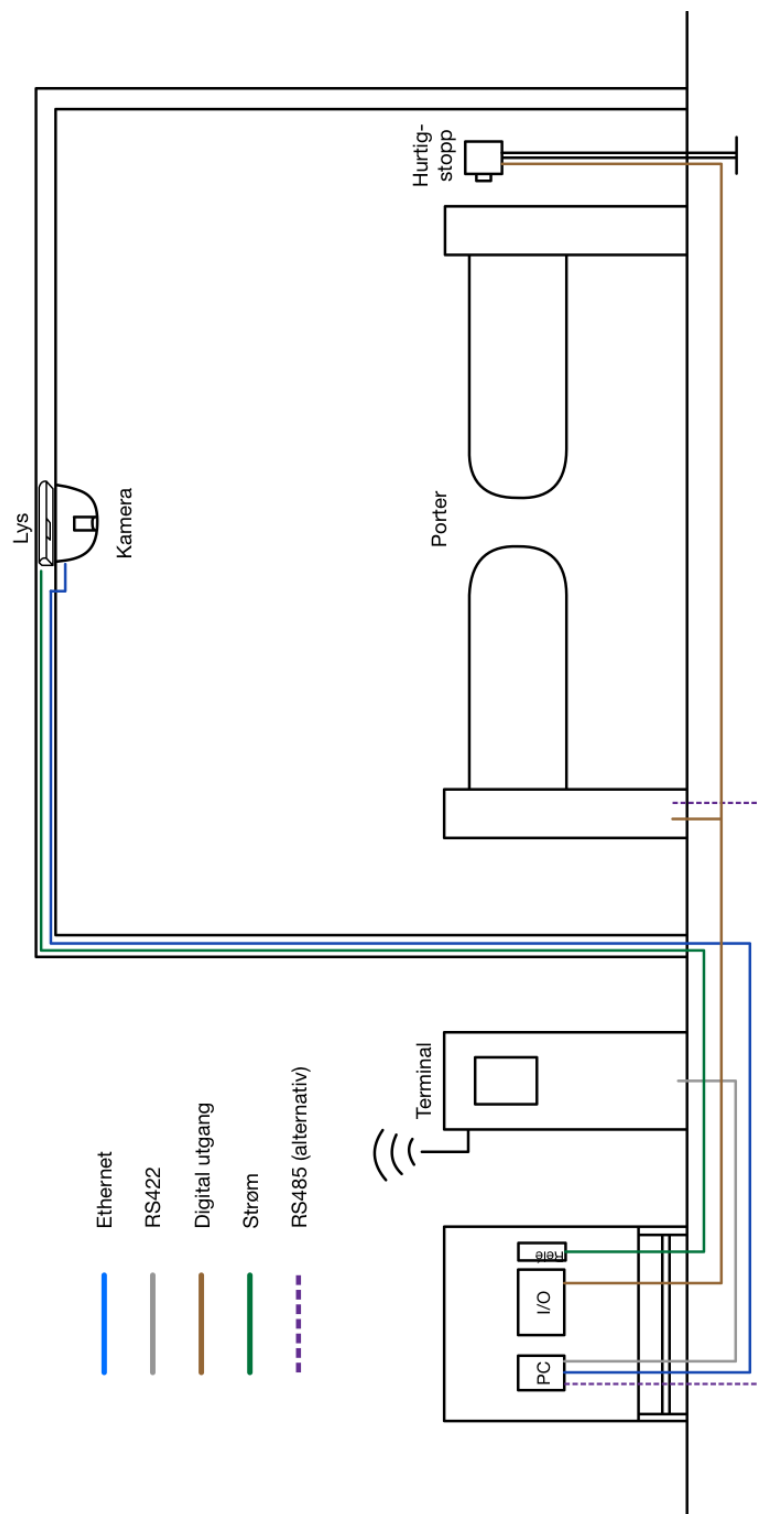
For å sette opp høyttaleren og få nok volum må de kobles til en forsterker. Modellen som er anskaffet er en Marco EW2-MS [48] som er merket med en effekt på 90 W, som ifølge [61] er omtrent 1.6 ganger større enn den anbefalte på kanalen for enkelt vokal bruk. Høyttalerkretsen kommer på 60 W. For å oppnå en krets som tilfredsstille kravene til forsterkeren med mindre enn 90 W og 5 – 8 ohm må den kobles i henhold til figur 3.6.14. Hvis høyttaleren spiller på fullt er forsterkeren for svak, $4 \cdot 25W = 100W$. I forsterkeren er det også en signalgenerator som lager situasjonstilpassede signaler. Den har også en linje-inngang for avspilling av eksterne lydspor.



Figur 3.6.14: Eksempel på serie- og parallellkobling av høyttalerpar [62].

AutoPAX-PC-en har ikke et innebygd lydkort og trenger derfor et eksternt lydkort. Det er tatt i bruk et USB lydkort kalt PLAY!3 [49]. Denne har en linje inn og en linje ut for muligheten til å spille av lydfiler samt opptak fra en mikrofon. For oversikt se figur 3.6.12.

3.6.3 Terminal



Figur 3.6.15: Systemarkitektur for terminaler.

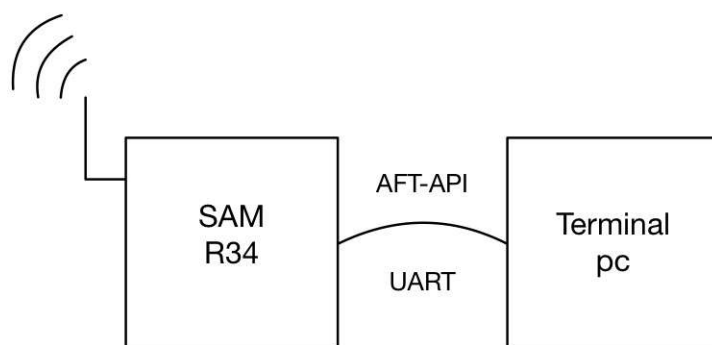
På terminalen vil det befinne seg en del utstyr som skal ivareta sikkerheten til passasjerene som venter eller skal stige på- eller av fergen. Det er også lagt til en funksjonalitet for deteksjon av objekter mellom sikkerhetsportene med mulighet for utvidelse mot telling av personer ved av og på stigning.

3.6.3.1 Landport

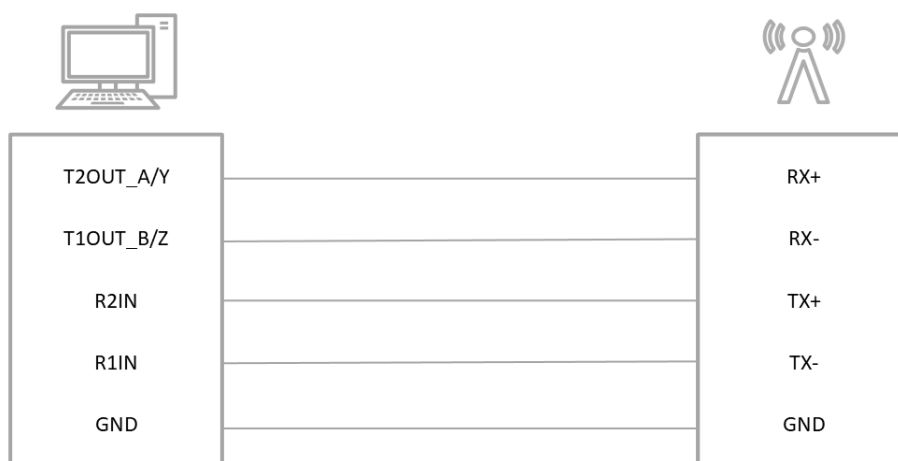
Det er et sett med sikkerhetsporter på land vil hindre at noen kan spaserer inn i dokking sonen før fergen har lagt til og fortøyd. Portene er et komplett sett av typen MT311 og styres av et 24 VDC inngangssignal, se vedlegg **F - Landport koblingskjema** for koblingskjema. Alternativt har fabrikanten av portene henvist til dokumentet som finnes i vedlegg **G - Landport seriellprotokoll** som gir en oversikt over mulighetene for å kontrollere portene over RS-485. Her økes mulighetene dramatisk ved muligheter som å endre hastigheten på portene, sensitivitet ved obstruksjon og mange tilbakemeldinger for ulike tilstander eller feilmeldinger.

3.6.3.2 Terminal-PC

Hjernen bak operasjonene på terminalen er en PC av typen Jetson NX2, som brukes for å samhandle med AFT-API-et [51]. Dette skjer på samme måte som beskrevet i tidligere underseksjoner 3.6.1.1 og 3.6.1.2. For illustrasjon se figur 3.6.16 og eksempel av oppkobling se figur 3.6.17.



Figur 3.6.16: Kommunikasjonen mellom terminalen og AFT-API-et.



Figur 3.6.17: Oppkobling av RS-422 mellom terminal-PC og AFT-API.

Terminal-PC-en skal også operere et I/O-kort for digitale signaler og er tilkoblet et IP-kamera for bildeprosessering med objektgjenkjenning programvare. Denne PC-en er på grunn av objektgjenkjenningen utstyrt med bedre maskinvare, den har 8 GB minne, 250 GB lagring og er utstyrt med de samme grensesnittene som AutoPAX-PC-en se figur 3.6.7.

3.6.3.3 I/O-kort

For å kunne håndtere de forskjellige periferienhetene på terminalen har det blitt anskaffet et I/O-kort av typen Advantech USB-5830 [52] til terminal-PC-en. I/O-kortet benytter seriell kommunikasjon over en USB-kabel og har som hensikt å gjøre industrielle inngang- og utgangsmoduler, modulære og tilgjengelig for PC-styring. Programmeringen kan gjøres ved mange forskjellige programmeringsspråk, som for eksempel C++, C#, Python, QT for å nevne noen. Oversikt over innganger og verdier finnes i tabell 3.6.5 og vises i figur 3.6.18.

Kommunikasjon i kortet er inspirert av guiden til Advantech for kommunikasjon og programmering av I/O-kortet [41]. For å lese og dekode inngangssignalet og skrive til utgang kan en enkelt bruke kodesnutten 1. Bibliotekene som brukes i denne oppgaven finnes ved å laste ned Navigator-programvaren [41]. En ønsker å overføre filen med navn "Automation" over til lokal Python-programvare. Disse filene finnes som regel i mappestrukturen slik som vist her.

Mappestruktur for Automation-filen

```
C:/
├─ Advantech/
│   └─ DAQNavi/
│       └─ Automation/
```

Mappestruktur hvor Automation-filen skal legges

```
C:/
├─ Users/
│   └─ Username/
│       └─ anaconda3/
│           └─ Lib/
│               └─ site-packages
```

```

import sys
sys.path.append('.')
import time
from Automation.BDaq import *
from Automation.BDaq.InstantDiCtrl import InstantDiCtrl
from Automation.BDaq.InstantDoCtrl import InstantDoCtrl
from Automation.BDaq.BDaqApi import AdxEnumToString, BioFailed
from Automation.BDaq.CommonUtils import kbhit

deviceDescription = "USB-5830,BID#0"
profilePath = u"../../profile/DemoDevice.xml"

startPort = 0
portCount = 2
portCountOutput = 1

inputList = ['0x1', '0x2', '0x4', '0x8', '0x10', '0x20', '0x40', '0x80']
outputList = ['0x1', '0x2', '0x4', '0x8', '0x10', '0x20', '0x40', '0x80']

outputListDec = [[0], [1], [2], [4], [8], [16], [32], [64], [128]]

def writingToOutput(device, profilePath, outList, startPort, portCount,
    ↪ outPin):
    ret = ErrorCode.Success

    instantDoCtrl = InstantDoCtrl(device)
    instantDoCtrl.loadProfile = profilePath

    ret = instantDoCtrl.writeAny(startPort, portCount, outList[outPin])
    instantDoCtrl.dispose()
    return 0

def readingFromInput(device, profilePath, inputList, startPort, portCount):
    ret = ErrorCode.Success

    instantDiCtrl = InstantDiCtrl(device)
    instantDiCtrl.loadProfile = profilePath

    while not kbhit():
        ret, data = instantDiCtrl.readAny(startPort, portCount)
        if BioFailed(ret):
            break

        if data[0] != 0:
            indexPort = inputList.index(hex(data[0]))
            #print(indexPort)
            return indexPort

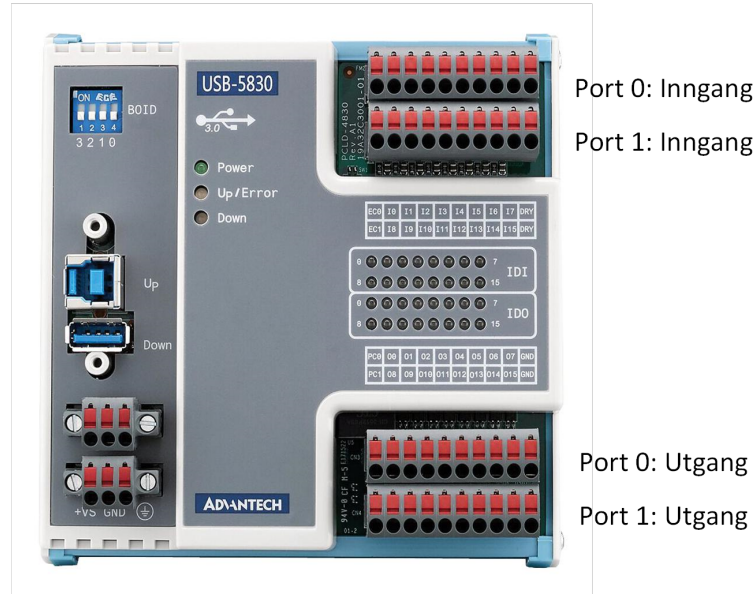
        if data[1] != 0:
            indexPort = inputList.index(hex(data[1])) + 8
            #print(indexPort)
            return indexPort

    instantDiCtrl.dispose()

    if data[0] or data[1] == 0:
        return 0

```

Listing 1: Pythonkode for lesing og skrivning til digitalt I/O-kort.



Figur 3.6.18: Oversiktsbilde av USB-5830, I/O-modul [63].

Inngang	Adresse	Port	Utgang	Adresse	Port
I0	0x01	0	O0	0x01	0
I1	0x02	0	O1	0x02	0
I2	0x04	0	O2	0x04	0
I3	0x08	0	O3	0x08	0
I4	0x10	0	O4	0x10	0
I5	0x20	0	O5	0x20	0
I6	0x40	0	O6	0x40	0
I7	0x80	0	O7	0x80	0
I8	0x01	1	O8	0x01	1
I9	0x02	1	O9	0x02	1
I10	0x04	1	O10	0x04	1
I11	0x08	1	O11	0x08	1
I12	0x10	1	O12	0x10	1
I13	0x20	1	O13	0x20	1
I14	0x40	1	O14	0x40	1
I15	0x80	1	O15	0x80	1

Tabell 3.6.5: Utgangsadresser og pinner til USB-5830-kortet [41].

3.6.3.4 Manuelle knapper

I terminalskapet er det montert to sett med to trykkknapper for enkelt å kunne kjøre fergeluken og begge settene med sikkerhetsporter manuelt. Det er et sett med pil opp og ned til luka og ett for portene. Disse er kun begrenset av forrigling mellom luke og porter samt nødstoppen.

3.6.3.5 Hurtigstopp

Hurtigstoppen er plassert mellom fergeporten og landporten på fergekaien. Denne er ment som en ekstra sikkerhet for passasjerene om det skulle oppstå farlige eller ubehagelige situasjoner. Denne er ikke utstyrt med en låsemekanisme så den vil returnere til normal tilstand etter trykker. Tanken bak dette er at det skal være enklere å gjenoppta fergedriften om noen skulle ta den i bruk og heller forflytte tilbakestillingsfunksjonaliteten til en ekstern operatør.

3.6.3.6 Gangbelysning

Terminalen utstyres med belysningen i sikkerhetsbøylen for å kunne sikre trygg drift på kveldstid. Passasjerene får et mer opplyst område å bevege seg på og vil fungere som et risikoreduserende tiltak. Får å unngå manuell styring av lyset er tilførselen utstyrt med et skumringsrele av typen Acti9 IC2000 [54] tilpasset utendørs bruk. Lyset er en type lyslist som klippes opp slik at det blir omkring 1.5 meter på hver side av kameraet. Lyslisten er av type StripLine 391415 [64] og tilfører sikkerhetsbøylen 1368 lumen per meter. Til sammen vil det være 3 m med lyslist og tilføre totalt 4104 lumen, og noen av styrken vil forsvinne da det havner bak innkapsling. For sammenligning er et normalt gatelys for fortau omkring 4000 lumen [65].

3.6.3.7 Kamera

For å kunne detektere passasjerer på gangbroa henger det et kamera i sikkerhetsbøylen på terminalen. Dette kameraet er et IP-kamera og er tilkoblet med en ethernetkabel, den bruker UDP/IP for å sende videostrømmen til terminal-PC-en. Dette er også et kamera som støtter PoE som strømtilførsel og det er derfor montert opp en PoE-svitsj i terminalskapet [55]. For mer om kamera og design, se underseksjon 4.4.3.

3.6.3.8 Terminalenhet

IES-gruppen som utviklet AFT-API-et laget også en bestillingsterminal for henting av fergen til korresponderende side av fergeruten. Denne terminalen inneholder antennen og mikrokontrolleren til AFT-API-et og kobles sammen med terminalskapet med en RS-422-kabel.

Kapittel 4

Prototype passasjerhåndterings-system

Dette kapitlet presenterer og diskuterer prototypene av passasjerhåndterings-systemet. Det som er presentert er resultatene gruppen fikk under utviklingen av en prototype, hvordan systemet fungerer og hvilke funksjoner systemet har. Disse resultatene kommer fra testing og evaluering av prototypen som da vil gi oss som utviklere informasjon om brukeropplevelsen og eventuelle forbedringer som burde gjøres.

Under hver presentasjon av prototypene er det en diskusjonsdel. Denne delen tar for seg diskusjon hvor prototypene blir diskutert og analysert slik at leseren kan få en forståelse på hvorfor ulike valg er tatt. I diskusjonsdelen vil også konsekvensene til prototypene presentert slik at det er mulig å utføre et videre arbeid med prosjektet.

Målet med dette kapitlet er å gi lesere en grundig og pålitelig presentasjon av funksjonaliteten til systemet, i tillegg til å demonstrere om de oppfyller målene som er presentert i innledningen.

4.1 Fergeport



Figur 4.1.1: Den fysiske fergeporten.

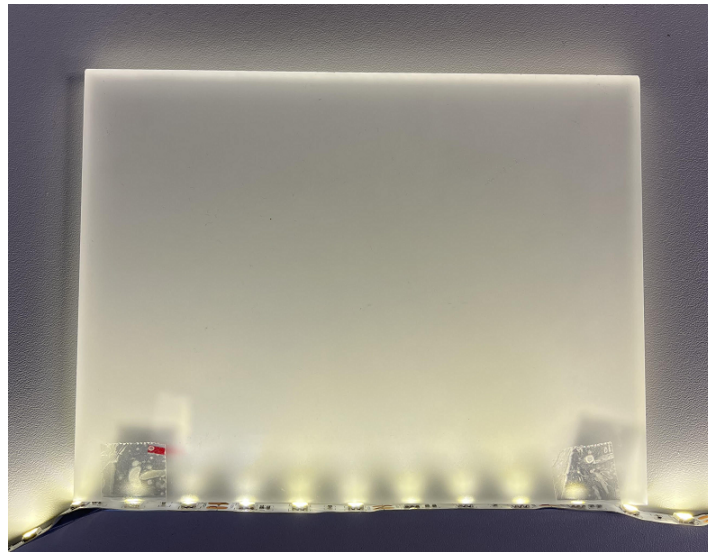
Om bord i fergen er det planlagt å montere den automatisk svingedørporten som en kan se i figur 4.1.1. Det er laget to identiske konstruksjoner av porten som skal utgjøre fergeporten. Det er benyttet rør med en diameter på $\text{Ø}42,4$ mm. I røret er det en U-profil hvor det skal ligge en PMMA-plate og LED-lyslister, se figur 3.1.3 for å se hvordan de skal ligge i U-profilen.

U-profilen har som hensikt å holde på plass platen inne i røret. Etersom LED-lyslisten genererer en del varme er det plassert en varmeavledende aluminiums list under den for å sikre levetiden til LED-stripen.



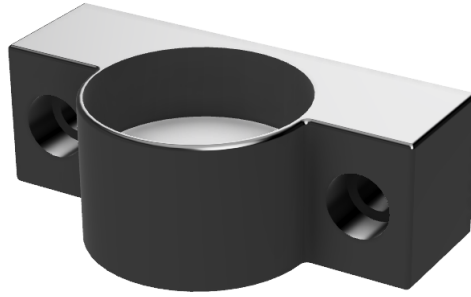
Figur 4.1.2: Nedfelt U-profil i fergeport. Tilpasset PMMA-plate.

Inne i stålrammen er det som nevnt planlagt å feste en satinert opal-plate av typen PMMA. På over og undersiden platen er det montert LED-striper som har til hensikt til å lyse opp platen med ulike farger. Disse fargene skal representere statusen til fergen.



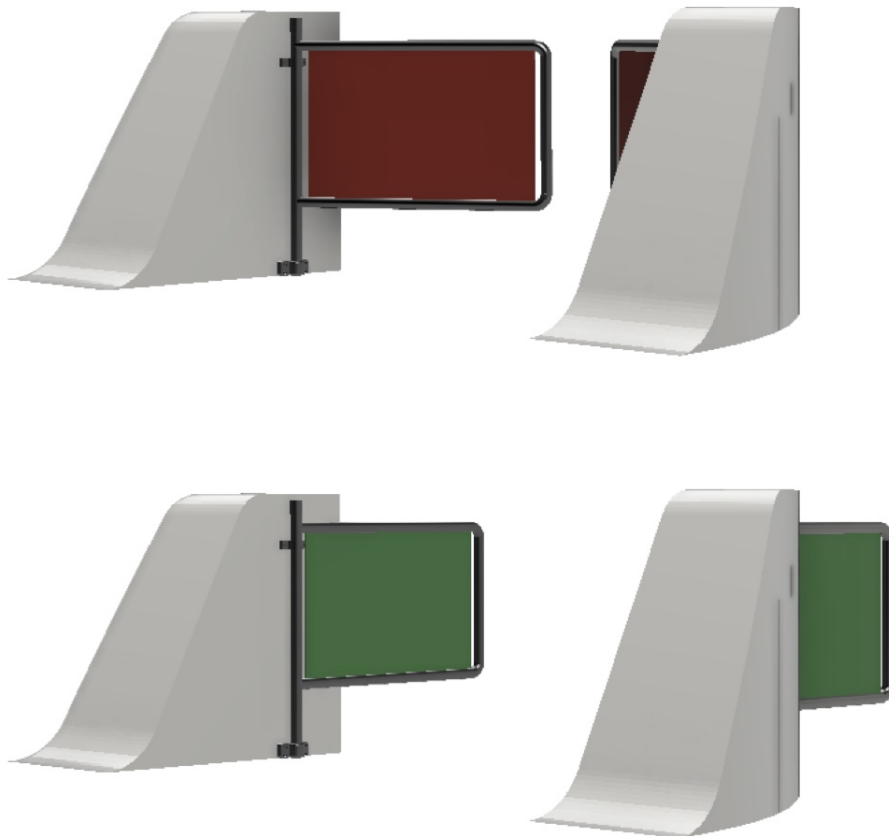
Figur 4.1.3: Test av LED-stripe i PMMA-plate.

Valgte en plate med en lystransmisjon på 50%. LED-lyset skal blinke rødt når den er i bevegelse og lyse grønt når porten er åpen. Mottok en test-PMMA-plate fra leverandør for å teste om den gjengir lyset slik gruppen ønsket.



Figur 4.1.4: Fusion 360-modellert festebrakett til fergeport.

Fergeporten skal festes til fergen med en 3D-printet festebrakett, se figur 4.1.4. Braketten er konstruert så fergeporten skal rotere i den. Hullene som den skal skrues fast til fergen i er tilpasset M8 unbrako-skruer.



Figur 4.1.5: Åpne porter (grønt lys) og porter som lukker (rødt, blinkende lys). Portene er sett fra siden. Festebrakett fra figur 4.1.4 er i bunnen og toppen av porten.

4.1.1 Diskusjon

Av flere årsaker rakk dessverre gruppen ikke å montere fergeporten på fergen. Gruppen erfarte at det tok lengre tid enn antatt å bestille deler, produsere og programmere fergeporter. Gruppen mottok en test-PMMA-plat og var svært fornøyde med denne, bestilte derfor inn mer så det skulle dekke hele fergeporten. PMMA-platen hadde forventet levering i slutten av april, men ankom ikke før bachelorleveringen 22.05.2023.

Det er tatt flere designvalg for å gjøre porten imøtekommende og trygg for passasjerer. Portdesignet er også presentert fram til Einar Johan Hareide som ga gruppen gode tilbakemeldinger angående designvalg [38].

For å gjøre de syrefaste portene mindre brutale er det valgt å bruke runde kantet på hjørnene til fergeportene. Runde hjørner reduserer også faren for kuttskader. Festebrakketten har også avrundede kanter for å unngå at passasjerene får kuttskader.

Målet med bruk av LED-lys i fergeportene er å gi passasjerene visuell informasjon. Det er benyttet farger som de fleste vet hva betyr. Når porten er i bevegelse seg skal LED-lyset blinke rødt for å indikere at det ikke skal foregå passasje gjennom porten. Når porten er åpen lyser den grønt for å indikere at det er klart for av- og påstigning. For å fremme disse lyssignalene i platen er det valgt en så høy lystransmisjon.

4.2 Landport



Figur 4.2.1: Landporten. Bildet sendt fra leverandør i Kina [8].

Landporten er bestilt komplett fra leverandør, men kommer ikke frem i tide [8]. Landporten er bygd opp av rustfritt AISI 316-stål for å tåle omgivelsene på fergekaien.

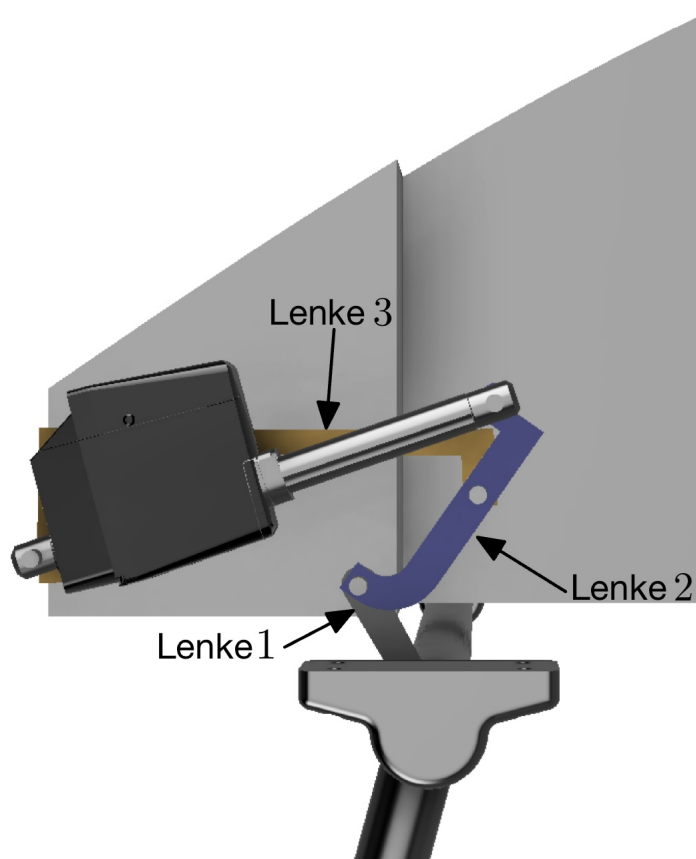
Gjennom kommunikasjonsprotokollen RS-485 bestemmer land-PC-en i el-skapet når porten skal åpne. Landporten har LED-striper på toppen som har en informativ funksjon i form av at det lyser rødt eller grønt basert på om det er klart for av- eller påstigning av fergen. Hastigheten på landporten er satt gjennom RS-485 så den har samme hastighet som fergeporten.

4.2.1 Diskusjon

Å bestille en ferdiglaget landport sparte gruppen en god del tid som da heller er brukt på andre arbeidsoppgaver. Selv om landporten ikke har ankommet enda skal det være relativt enkelt å sette den opp. Den trenger bare å få tilført strøm samt en enkel programkode gjennom RS485 fra land-PC-en for å driftes.

4.3 Testing av systemet

4.3.1 Resultat ved bruk av ledd og aktuator



Figur 4.3.1: Systemet med ledd og aktuator som er 13° lukket. Systemet er låst.

Når aktatoren økte slaglengden sin roterte lenke 2. Denne rotasjonen overførte seg til lenke 1, som førte til at porten begynte å rotere.

Etter å ha lukket porten 45° , befant både lenke 1 og 2 seg i en rett vinkel på 90° i forhold til koblingspunktet på porten. Når aktuatoren trakk slaglengden tilbake etter å ha lukket porten 45° , rykket lenkene raskt tilbake til sin opprinnelige posisjon, startposisjon, og porten stod 90° åpen.

Om aktuatoren fortsatte å øke slaglengden etter å ha lukket porten 45° , førte dette til en videre rotasjon av lenke 2 som igjen lukket porten ytterligere. Imidlertid oppstod en situasjon der systemet låste seg nå porten er 15° lukket. Når aktuatoren prøvde å trekke seg inn igjen, rykket lenkene igjen raskt tilbake til sin opprinnelige posisjon, startposisjon, og porten stod 90° åpen igjen. Se figur 4.3.1 for å se hvordan lenkene ligger når porten er 13° lukket.

4.3.1.1 Diskusjon

I figur 4.3.1 har aktuatoren enda 38.8 mm å trekke seg ut på, men systemet har låst seg. Teoretisk sett skulle aktuator økt slaglengden sin mer og da fått lenke 2 til å rotere med klokken. Porten skulle da klart og rotert de siste gradene for å gå til 0° . Dette skjedde ikke i praksis og systemet låste seg.

Systemet begynner å slite når porten er 45° lukket, som er når lenke 1 og 2 står i en rett vinkel på 90° i forhold til koblingspunktet på porten. Aktuatoren klarer å lukke systemet mer etter de 45° , men klarer ikke å åpne det behersket og rolig når porten er 45° lukket.

Mistenker også at det hurtige rykket som oppstår idet aktuatoren trekker inn slagstangen kommer fra at lenkene ligger i spenn hele tiden. Observerte lik oppførsel på systemet ved testing med 3D-bitene. Tror også at systemet er så mye i spenn at kraften mellom lenkene fører til en mekanisk lås når porten er 13° lukket.

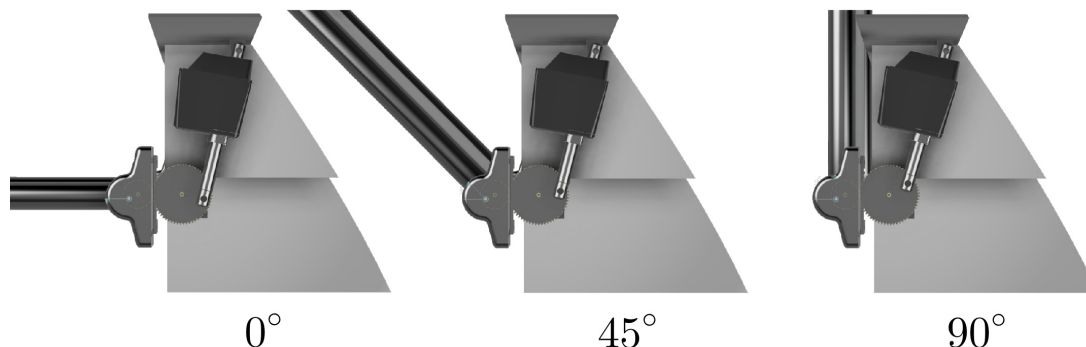
Det å bruke leddene til å rotere porten med kraft fra aktuatoren skal i teorien fungere, men som vist i resultatet fungerte det ikke i praksis. Etter å ha studert systemet i virkeligheten og via simulering kan det virke som at systemet låser seg mekanisk. Har prøvde å løse problemet med å endre lenke 2 og 3 sine fasonger og lengder, det hjalp ikke. I senere tid ser en at det hadde vært lurt å involvere mekanikk studenter for å kunne undersøkt denne problemstillingen.

Grunnet tidspress er det valgt å ikke bruke mer tid på å finne ut hva man må legge til, andre vinklinger eller løsninger, for å få leddene til å rotere om.

Tannhjul er derimot en løsning som enkelt fungerte, men er mer krevende å montere inn i porten. Fant ut en god måte å sveise inn tannhjulet i porten etter å ha idémyldret med ITK-verkstedet. Det hadde vært lettere å montert ledd enn tannhjul, men valgte å gå for tannhjul fordi det var en løsning som faktisk fikk porten til å rotere 90° .

Ved senere påbygging av systemet er det mulighet til å undersøke ledd-alternativet i en større grad og da eventuelt bytte ut tannhjulene med ledd. Ledd-systemet som er laget i Fusion 360 ligger i vedlegg **A - GitHub, kode og Fusion 360-filer**.

4.3.2 Resultat ved bruk av aktuator og tannhjul



Figur 4.3.2: Aktuator og tannhjul sine posisjoner i 0°, 45° og 90°.

Når aktuatoren har rom til å rotere om egen akse, klarer den å overføre kraften sin til tannhjulene som da roterer porten fra 0° til 90°. Må ha rett vinkling på tannhjulene ellers sliter systemet med å vite hvilken retning det skal gå og systemet låses.

Det lille tannhjulet roterer 90° og det store roterer 42° som vil si at det lille tannhjulet roterer mer enn dobbelt så fort som det store. Dette er fordi forholdet mellom tannhjulene er på mer enn 2, 2.143 : 1.

Hastigheten til det minste tannhjulet, når porten har maksimal hastighet, er på 53.625 mm/s som vil si at det tar porten mindre enn 0.5 sekunder å åpne seg.

Aktuatoren har en slaglengde på 22.484 mm når porten er 0°, 10.26 mm når porten er 45° og 2.044 mm når porten er 90°.

Figur 4.3.2 viser hvordan aktuator vil rotere tannhjulet og da porten.

4.3.2.1 Diskusjon

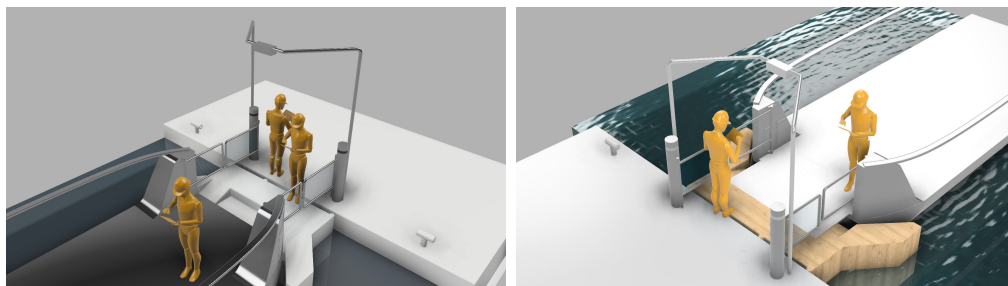
En må endre på endepunktet til aktuatoren for at dette er en løsning som kan brukes som en del av systemet. Dette gjøres ved bruk av BusLink-kabelen til aktuatoren.

En kan også endre hastigheten til aktuatoren ved bruk av samme BusLink-kabel. LINAK informerer at ved en hastighet under 60% av maks hastighet kan aktuatoren miste den oppgitte lastkapasiteten som er på 600 N. Gruppen mistenker at det ikke blir et stort problem ettersom aktuatoren skal rotere porten og eventuell luftmotstand. Dette er noe som må testes og prøves litt ut når hele systemet er oppkoblet.

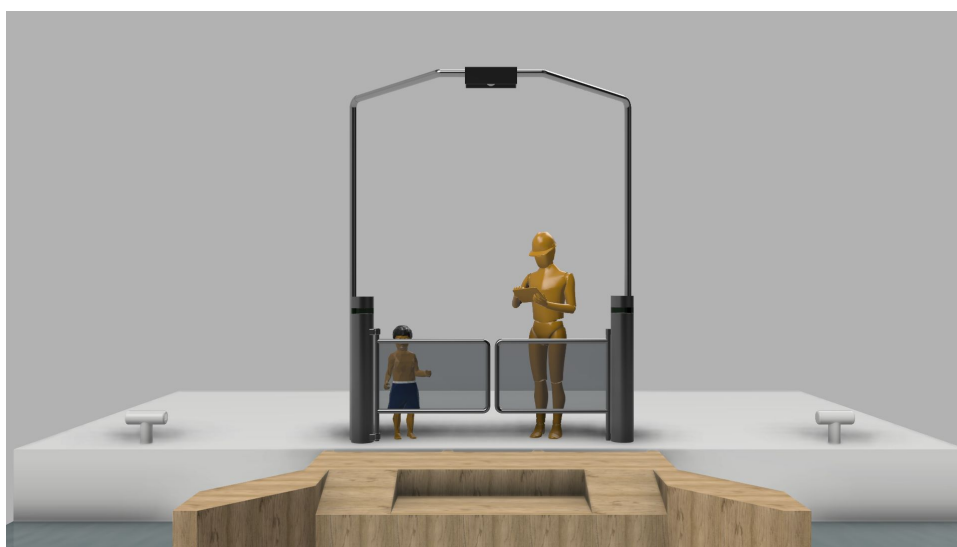
For å se hvordan en kobler opp, endre hastigheten og endestopp til aktuatoren se databladet til LINAK om BusLink [66].

Systemet er også testet ut i virkeligheten med å 3D-printe ut tannhjulene, og observerte at portene roterte etter kraftpåvirkning fra tannhjul som er påført fra aktuator.

4.4 Sikkerhetszone



Figur 4.4.1: Kameraløsning, designforslag.



Figur 4.4.2: Sidesyn, oppsett fra kameraside.

4.4.1 Programvare

Ved å gjøre et hurtig søk på en tilfeldig søkemotor, med for eksempel “People detection in zone algorithm” eller “python detection zone”, kommer det opp uendelig mange treff. Et treff som ofte kommer om igjen, er YOLO-modellen. YOLO-modellen er enkel å bruke, og er laget for å detektere objekter i rekordfart. Siste versjon av YOLO, versjon 8, er den tregeste av de utgitte modellene, men den mest nøyaktige. Modellen veier opp hastighet mot nøyaktighet, jo høyere nøyaktighet, jo kraftigere prosessering.

Funksjonen for deteksjon og klassifisering er utviklet i Python, og kan finnes i vedlegg **A - GitHub, kode og Fusion 360-filer**. Algoritmen er utviklet på PC og testet ut på CPU før gruppen gikk til innkjøp av en DSBOX-NX2 [39]. Denne boksen er utviklet av FORECR med innmat fra Nvidia og består av et Nvidia Xavier NX-kort [67] med tilkoblinger for blant annet seriellkommunikasjon, Ethernet og HDMI.

Nvidia Xavier Nx er utviklet av Nvidia for å gjennomføre bildegjenkjenning på GPU og være ekstremt rask. Nvidia har utviklet en rekke software som er tilpasset maskinen deres, og disse brukes i oppgaven. Oppgaven har brukt software

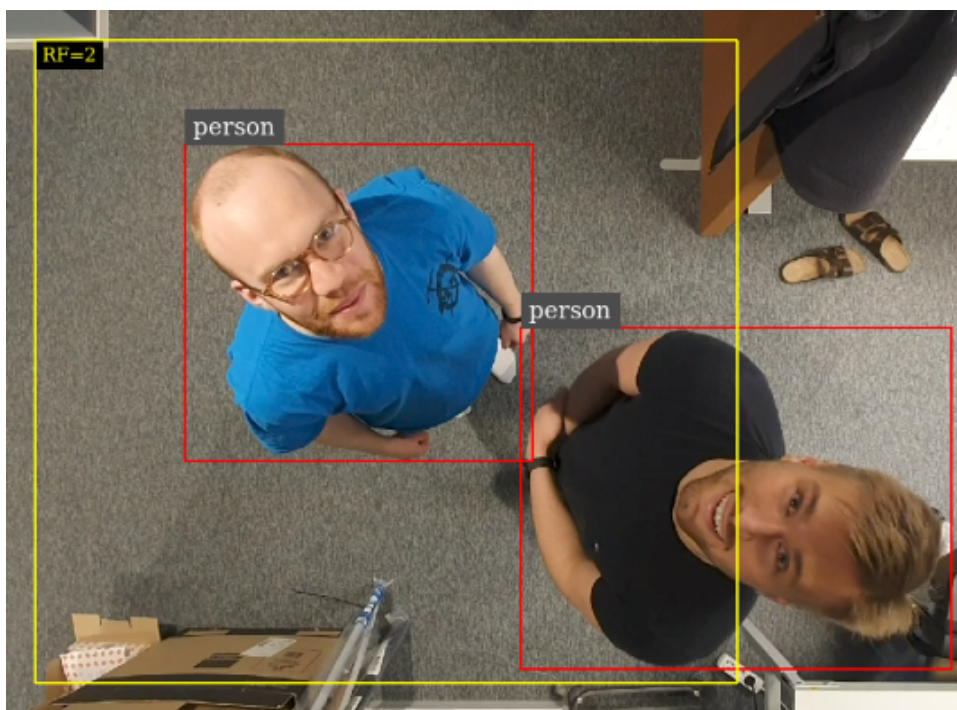
som Deepstream [68], TensorRT [69] og GStream [40]. Deepstream og TensorRT er software som installeres direkte på Xavier Nx-modulen, og gjør det mulig å kjøre programmer på grafikkprosessoren (GPU) istedenfor CPU. Dessverre er det ikke mulig å konvertere direkte fra CPU over til GPU, så dette måtte endres i konfigurasjonsfilene i brukergrensesnittet.

I denne oppgaven er YOLO-modellen implementert i Deepstream og TensorRT, med å konvertere den opprinnelige YOLO-modellen fra “yolov8l.pt” og over til “yolov8l.onnx”, deretter “yolov8l.engine”. Konverteringen fra .pt -> .onnx -> .engine er gjort ved hjelp av manualen til SeedStudio [70].

Ved å implementere YOLO direkte på Xavier NX senker inferens-tiden seg fra omkring 3500 – 4000 ms og ned til omkring 200 ms. Inferens er tiden det tar fra et bilde kommer inn i modellen, til modellen har klassifisert og bestemt seg for hva som er i bildet. Ved FP32 er inferenstiden på omkring 3500 ms, og ved bruk av INT8 kan det senkes helt ned til omkring 50 ms.

Først kjørte alt direkte på CPU som gjorde det svært enkelt å hente ut all informasjonen, som hvor mange eller hva som stod innenfor sikkerhetssonen. Ettersom scriptet for YOLO kjøres direkte på GPU, må det brukes Python Bindings [40] for å hente ut informasjon. Analogt kan dette sammenlignes med informasjon som ligger på en database, og må hentes ut. Bindingene mot den såkalte databasen, er ikke rett frem, og det må en del oppsett til å få til dette.

4.4.2 Filkonfigurering og kjøring av program



Figur 4.4.3: Deteksjon av to personer ved kjøring av program.

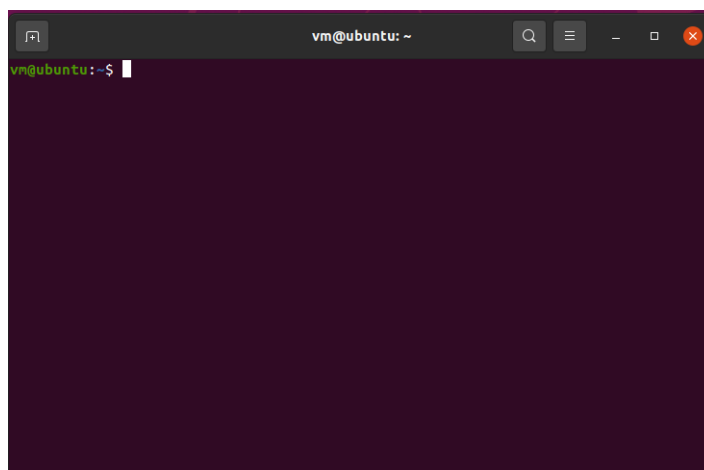
Ved kjøring av script i Deepstream er det tre filer som er sentrale og som det kan gjøres endringer i. Filen "Deepstream_ap_config.txt" inneholder alle parametere for oppsett av videostrøm, format på video, og de rent praktiske parametere. Filen "config_nvdsanalytics.txt" inneholder alt som har med deteksjon og oppsett av sone å gjøre. Denne filen er opprettet med Gstream [40], som gjør det mulig å spesifisere sikkerhetssonen og eventuelt andre spesifikasjoner om ønskelig.

Fokuset i denne oppgaven har vært på deteksjonen innenfor gitt område, og det er programmert inn mulighet for telling av passasjerer også. Telling av passasjerer krever tracking, altså en logg på hvor et objekt har gått og i hvilken retning. Tracking er ikke satt opp, men kan implementeres etter behov i filen "config_nvdsanalytics.txt".

```
HOME/
├── DeepStream-Yolo/
│   ├── docs
│   ├── nvdsinfer_custom_impl_Yolo
│   ├── utils
│   ├── [...]
│   ├── config_nvdsanalytics.txt
│   ├── deepstream_app_config.txt
│   └── config_infer_primary_yoloV8.txt
```

Filen "config_nvdsanalytics.txt" inneholder en del informasjon etter at den har blitt konfigurert. Innholdet er satt opp på følgende måte for å få tilgang tilføre informasjon til skjermbildet, og også for å hente ut informasjon ved hjelp av GStream senere. Informasjon om hvilke parametere som er tilgjengelig og hvordan de kan brukes finnes på hjemmesiden til Nvidia [40].

For å kjøre selve koden som aktiverer YOLO-modellen må terminalen brukes. Terminalen i Linux sitt operativsystem kan enkelt søkes opp ved å trykke på **Windows**(Windows) eller **Command**(Mac) og skrive Terminal etterfulgt av **Enter**. Da vil terminalvinduet komme opp du vil får et vindu slik som figur 4.4.4.



Figur 4.4.4: Eksempel på Linux-terminal.

```

[property]           #Screen information
enable=1             #Bool
config-width=1280    #Int
config-height=720    #Int
osd-mode=2           #Int
                     #0: disable of any lines, ROIS and text
                     #1: Display only lines and static info
                     #2: Display all info, information and dynic info
display-font-size=12 #String

[roi-filtering-stream-0] #Enables detection within ROI(Region of interest)
enable=1
roi-RF=200;50;800;50;800;600;200;600 #Area where ROI is placed
inverse-roi=0
class-id=-1 #Allows every class to be detected

[line-crossing-stream-0] #Let the used count passengers on and off ferry,
→ NOT IN USE
enable=0
line-crossing-Entrance=789;672;1084;900;851;773;1203;732
# two first position is direction arrow, two next parameters are the line
→ position.
line-crossing-Exit=789;672;1084;900;851;773;1203;732
extended=0
mode=loose #loose, strict or balanced. Loose counts everything, strict
→ counts only whats spesified and balanced counts something in between.

```

For å aktivere scriptet må en skrive kommandoene som er vist nedenfor. Kommandoene tar deg inn i mappen til "DeepStream-Yolo", og kjører scriptet "deepstream-app" med spesifikasjonene oppgitt i "deepstream_app_config.txt". Det vil da se ut som i figur 4.4.3.

```

$ cd DeepStream-Yolo
$ deepstream-app -c deepstream_app_config.txt

```

4.4.3 Kamera og design



Figur 4.4.5: Eksempelbilde av et ABUS TVIP44510 [71].

For å kunne benytte seg av modellen som er programmert, er det kjøpt inn et kamera av typen ABUS TVIP44510. Kameraet er av typen som vist i figur 4.4.5.

Valget stod mellom en rekke sensorer og kamera. Kamera er en billig og robust løsning, med mulighet for å koble seg på fra andre steder som for eksempel "shore

control lab” ved en eventuell tilkobling til Ethernet og 5G-kommunikasjon. Kameraet har en utsiktsvinkel på 98° som gjør det mulig å detektere objekter som måtte befinne seg i sikkerhetssonen, slik som vist i figur 4.4.5. Det er laget et forslag til videre utbygging av kameradeteksjon med inspirasjon fra masteroppgaven om “Design av brukerreise og brygger til autonom passasjerferge” [6].

Tanken er en bøyle eller stolpe som skal virke intuitivt som en inngang eller portal. Det er tenkt å konstruere bøylen/stolpen i rustfritt stål og plassere den “midt” over portene. For senere utvikling kan denne bøylen utvikles med skjerm for status og indikasjon.

4.4.4 Diskusjon om kommunikasjon mot ferge

Det er ingen resultater å vise til ettersom systemet ikke har blitt testet ut. Det drøftes da heller rundt måten kommunikasjonen er mot fergen.

Det er ikke noen praktisk måte å få varslet fergen fra sikkerhetssonen om brudd i sikkerheten uten bruk av trådløse signaler. Land-PC-en er koblet til AFT-API-et for nettopp dette.

Trådløse signaler er ikke like stabile og er mer utsatt for forstyrrelser enn kablede løsninger. Gruppen slite seg selv derfor spørsmål om en slik løsning er trygg nok for applikasjonen den er ment for. Kommunikasjonen skal ta for seg all signalbehandling til rettet mot sikkerhetssignaler og portstyring på landside. Land-PC skal også sende meldinger om bruk av hurtigstoppknappen som er tenkt å monteres mellom portene (ferge og landside).

For å prøve å få besvart krav rund sikkerheten ved autonome ferge drift var en samtale med Nils Hector Bua [72] fra Sjøfartsdirektoratet startet. Flere positive tilbakemeldinger på løsningen kom og responsen på de innovative delen i oppgaven var at de burde prøves ut. Med dette, bestemtes det at kommunikasjonen mellom fergen og terminale måtte ansees å kunne gjøres stabil og rask nok til bruk i denne applikasjonen. På videreutvikling med telling av passasjerer sier Nils Hector at noe av det viktigste er telling av passasjerer ombord, for å raskt kunne sette i gang søk og redningsoppdrag ved behov, og dette er en prioritert oppgave ved fergedrift.

4.4.5 Sikkerhetsbøyle

Sikkerhetsbøylen opererer som en inngang til fergen. I toppen og litt vertikalt ut fra kaien, skal det monteres et kamera. Kameraet ser rett ned på sikkerhetssonen.

4.4.6 Diskusjon

Sikkerhetsbøylen vil oppleves som en inngang/utgang til fergen, og inntrykket passasjerer får fra dette tårnet vil ha noe å si for brukeropplevelsen. Det å ha et stort og høyt tårn vil oppleves som fornem og imponerende noe som vil gjøre inngangen til fergen mer innbydende hvilket vil tiltrekke seg flere passasjerer. Et høyt og stort tårn vil også skape en trygghetsfølelse om at det er lett å komme seg både inn og ut fra fergen.



Figur 4.4.6: Sikkerhetsbøyle på kai med ferge lagt til.

Det ble først utviklet et program som skulle detektere objekter på lokal PC. Dette programmet kjørte på CPU og gikk ekstremt sakte, og ville ikke være sikkert nok. Da DSBOX-NX2 ble introdusert kjørte denne saktere på CPU enn PC-en brukte tidligere. Det var derfor nødvendig å konvertere programmet til å kjøre på GPU. Det viste seg at Nvidia har predefinert og laget et oppsett for dette allerede, som gjorde programmet overflødig. Når programmet ble kjørt på GPU gikk det ekstremt mye raskere og kan da faktisk brukes til prosessering. Tiden det tar for programmet å detektere i dette tilfellet et menneske, ble målt til 200 ms. Ettersom denne prosessen skal varsle fergen og landport kan vi regne med at det i løpet av omkring 300ms er gitt signal til prosessen om å stoppe portene.

4.5 Telling av passasjerer

Som en del av sikkerhetsfaktoren for trygg sjøfart er en avhengig av at telling av passasjerer er gjort riktig. Foreløpig er ikke telling av passasjerer en del av oppgaven, men det er allikevel en såpass viktig faktor at det må være her. Slik båten er designet, er det i dag lov med opptil tolv passasjerer om bord på fergen til enhver tid. I dialog med Sjøfartsdirektoratet har det blitt stilt spørsmål til hvorvidt en barnevogn eller annet transportmiddel skal registreres som en passasjer eller ikke. Dette er et vanskelig spørsmål, særlig med tanke på en eventuell evakuering eller redningsoppdrag. Som pekepinn har det blitt enighet om at en barnevogn skal antas å inneholde en passasjer. Det er bedre at en tar utgangspunkt i den verste situasjonen som kan oppstå, og deretter avkrefte eventuell informasjon. Etter dialog med Sjøfartsdirektoratet, kom de med følgende uttalelse:

“This is in fact a quite challenging question. The main objective of such registration is to be able to properly muster, evacuate and perform SAR actions if necessary. In order to do so, the correct number of passengers onboard is needed and therefore both the approach of counting a baby stroller as containing an infant or not may lead to different problems, due to the uncertainty of that assumption. Nevertheless, I think that it is much more likely for a baby stroller to contain a baby than not, so, if such an assumption is necessary, it could be considered as a reasonable one.”
- Sifis Papageorgiou, Sjøfartsdirektoratet [73].

4.5.1 VR-briller



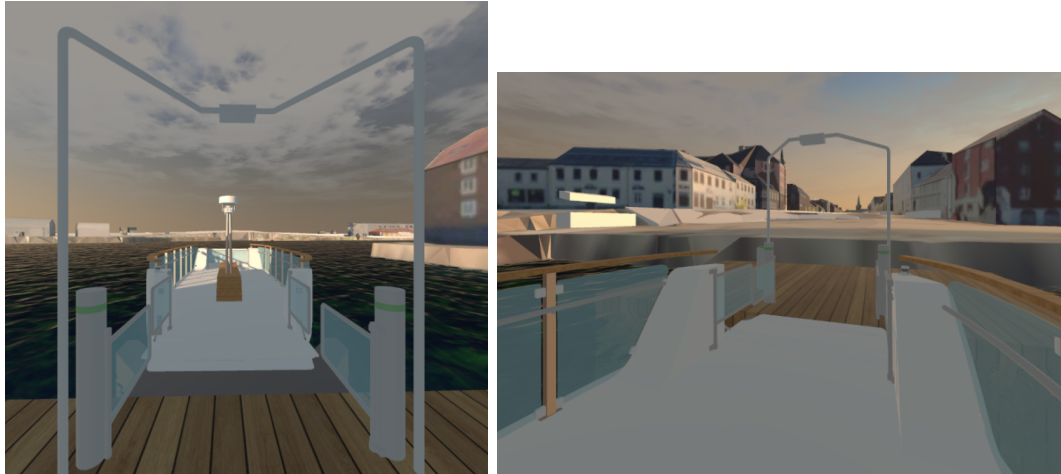
Figur 4.5.1: Test av VR-briller i MR-labben.

Robin Stokke satt opp ruter, PC og VR-briller i MR-labben. Når VR-verdenen er koblet opp til brillene konfigurerte han systemet til 1 : 1 trevirke-versjonen av milliAmpere2 i MR-labben.

Gruppen noterte ned opplevelser og sendte inn endringer til Robin som la disse til i VR-verdenen via Unity.

Komponent	Opplevelse	Hva funker fint	Burde endres
Port på ferge	Opplevs innbydende å gå av og på. Noe en kan holde fast i ved på/av stigning	Beveger seg i takt med porter på land	Ingen endring
Port på land	Opplevs innbydende å gå av og på. Komplimenterer sikkerhetsbøyle	Beveger seg i takt med porter på ferge	Ingen endring
Sikkerhetsbøyle	Opplevs som en inngang/utgang Føles litt kompakt.	En ser fra avstand hvor fergen skal	Endre høyde

Tabell 4.5.1: Tanker vedrørende testing av passasjerhåndteringssystemet med VR-briller.



Figur 4.5.2: VR-verdenen sett med VR-briller. Av- (bildet til venstre) og om bord-stigning (bildet til høyre).

4.5.1.1 Diskusjon om MR-lab

Hensikten med å teste ut systemet i MR-labben var at utvikler, gruppen, skulle få et innblikk i hvordan det er å være bruker/passasjer som skal ta fergen for første gang. Målet var å observere hva som fungerte, hva som ikke fungerte og om systemet var individuelt eller ikke. Hver og en komponent er vurdert individuelt samt hvordan de opererte sammen.

Når portene lukker seg får passasjerene visuell indikasjon, gjennom lys og at portene lukker seg. Så selv om hovedhensikten til portene er å holde passasjerer trygge, vil de samtidig gi en trygghetsfølelse på grunn av det visuelle.

Ved ombord og avstigning av sjøfartøy kan en oppleve en frykt for å falle i sjøen. Land- og fergeportene sikrer det med å fungere som et gjerde som da skiller passasjerer fra sjøen samtidig som det kan brukes som et gelender med ubalanse.

Etter første testing er det observert at sikkerhetsbøylen som originalt var på 2 meter burde økes til 3 meter fordi det følte for kompakt med original størrelse. En større port vil også oppleves mer innbydende. På figurene 4.5.2 er bøylen 3 meter. Det er dessverre ikke noen bilder av hvordan den så ut med 2 meter.

Grunnet tidspress er denne testingen dessverre bare gjort en gang, men det er et verktøy som er nyttig for videreutbygging på passasjerhåndteringssystemet.

4.6 Systemarkitekturen

Etter mye planlegging startet jobben med å sette sammen de forskjellige elementene av systemarkitekturen. For å kunne mota en kommando fra autonommaskinen, behandle denne og utføre de riktige handlingene samt gi en tilbakemelding på hva som skjer til autonommaskinen trengs en del kode. I denne seksjonen vil kodene som er skrevet bli drøftet kort og henvises til. Det har også vært behov for å lage et styreskap på terminalen til landporter og sikkerhetsbøylen, senere kalt el-skap.

4.6.1 Ferge

4.6.1.1 I/O-maskin programkode

For å programmere I/O-maskinen må e!COCKPIT brukes. Zeabuz har en GitHub repository med programfilen som allerede eksisterer, og det er blitt opprettet en egen gren i GitHub av original programmet for å kunne jobbe uavhengig frem til en utprøvd versjon av den nye programfilen virker.

I samarbeid med Bjørn-Olav Holtung Eriksen, seniorautomatiseringsingeniør og medgründer hos Zeabuz, ble det diskutert hvordan implementeringen av AutoPAX-systemet skulle integreres i den allerede eksisterende infrastrukturen til milli-Ampere2. Det ble i et møte bestemt å plassere all sekvensiell logikk i I/O-enheten og bruke det allerede implementerte grensesnittet for meldings utveksling mellom autonomisystemet og luke-styringen også for portene. I programfilen eksisterte det allerede en styringssekvens for lukestyringen, for å gjøre leseligheten av programmet mest mulig gjenkjennelig ble samme form også brukt til port-styringen. Se vedlegg **A - GitHub, kode og Fusion 360-filer** under "io_machine/io-machine_new.ecp", for siste versjon av programfilen med portstyring.

Første del av programmeringen var å sette opp en funksjonsblokk for DMX-styring av LED-lysene til portene. WAGO har et eget bibliotek for DMX-signaler kalt WagoAppDMX. For å sette dette opp er videoen [74] brukt som inspirasjon.

```
$WAGOGATE,<ID>,<Kommando>*Sjekksum
```

Figur 4.6.1: NMEA melding fra autonomimaskinen.

```
$WAGOGATEFB,<FORE_GATE_ID>,<Tilstand>,<Handling>,<Feil>,  
<AFT_GATE_ID>,<Tilstand>,<Handling>,<Feil>*Sjekksum
```

Figur 4.6.2: NMEA melding til autonomimaskinen.

Tilbakemelding	Beskrivelse
Fore gate 1. Tilstander 2. Feil 3. Handleringer	Tilstander: 1. 0: Undefined 2. 1: Open 3. 2: Close 4. 3: Blocked 5. 4: Paused Feil: 1. 0: Normal 2. 1: EM stop 3. 2: Gate ferry did not open in time. 4. 3: Gate ferry did not close in time. 5. 4: Gate terminal did not open in time. 6. 5: Gate terminal did not close in time. Handling: 1. 0: Idle 2. 1: Opening 3. 2: Closing
Aft gate 1. Tilstander 2. Feil 3. Handleringer	Tilstander: 1. 0: Undefined 2. 1: Open 3. 2: Close 4. 3: Blocked 5. 4: Paused Feil: 1. 0: Normal 2. 1: EM stop 3. 2: Gate ferry did not open in time. 4. 3: Gate ferry did not close in time. 5. 4: Gate terminal did not open in time. 6. 5: Gate terminal did not close in time. Handling: 1. 0: Idle 2. 1: Opening 3. 2: Closing
ID	
Fore port	1
Aft port	2

Tabell 4.6.1: Tilbakemeldinger fra portstyring.

ID	
Fore port	1
Aft port	2

Kommando	Beskrivelse
0: stop	Pauser alt, konstant rødt lys. Hvis pauset, stopp. Resettes tilbake til initieringssteget
1: open	Starter åpning med lydsignal og blinkende rødt lys
2: close	Starter lukking med lydsignal og blinkende rødt lys
3: reset	Går tilbake til initieringssteget, hvis feiltilstand er 0
4: dock_ready	(Ikke fulstendig implementert)

Tabell 4.6.2: Kommandoer og ID for portstyring.

Som nevnt tidligere er kommunikasjonen for lukestyring allerede eksisterende. Kommunikasjonsmetoden som brukes er NMEA 0183 meldinger. For å informere at meldingene skal til portene er det lagt ved en kode som ser etter kodeordet \$WAGOGATE og hvilken port det refereres til.

Det er blitt laget en oversikt over de forskjellige operasjonene som kan utføres i tabell 4.6.2 samt tilbakemeldinger i tabell 4.6.1. Dette er kun for kommunikasjon mot autonommaskinen. Meldingene blir sendt på formene illustrert i figur 4.6.1 og 4.6.2.

Sekvens-programmeringen følger stegene beskrevet i underseksjonen 4.7 som er lengre ned i rapporten.

En av de viktigste funksjonene i programmet er forriglingen av porter og luken. Denne kodesnutten skal hindre at lukene lukker mens korresponderende porter er åpne eller at portene åpner når korresponderende luker er lukket. Dette vil forhindre mekaniske skader forårsaket av brukerfeil.

```
// Interlocking with hatch
IF NOT(di_hatch_open) THEN
do_open :=FALSE;
do_close:=FALSE;
END_IF
```

Figur 4.6.3: Eksempel fra kildekode på forrigling av porter mot luke.

4.6.1.2 AutoPAX-PC programkode

I vedlegg **A - GitHub, kode og Fusion 360-filer** ligger siste versjonen Python skriptet "main.py". Dette er en testet kode for ruting av data mellom I/O-maskinen og AFT-APIet. Den håndterer seriell kommunikasjon og UDP/IP pakker i hver sine separate program-tråd og pakker om meldingene til videresending, enten til I/O-maskinen over NMEA0183 eller AFT-APIet over UART.

Meldingene som sendes til I/O-maskinen settes opp på formen:

```
$AUTOPAXFB,<REID>,<SEID>,<STATUS>*Sjekksum<CR><LF>
```

Mens meldingene som mottas fra I/O-maskinen kommer på formen:

```
$AUTOPAX,<REID>,<SEID>,<KOMMANDO>*Sjekksum<CR><LF>
```

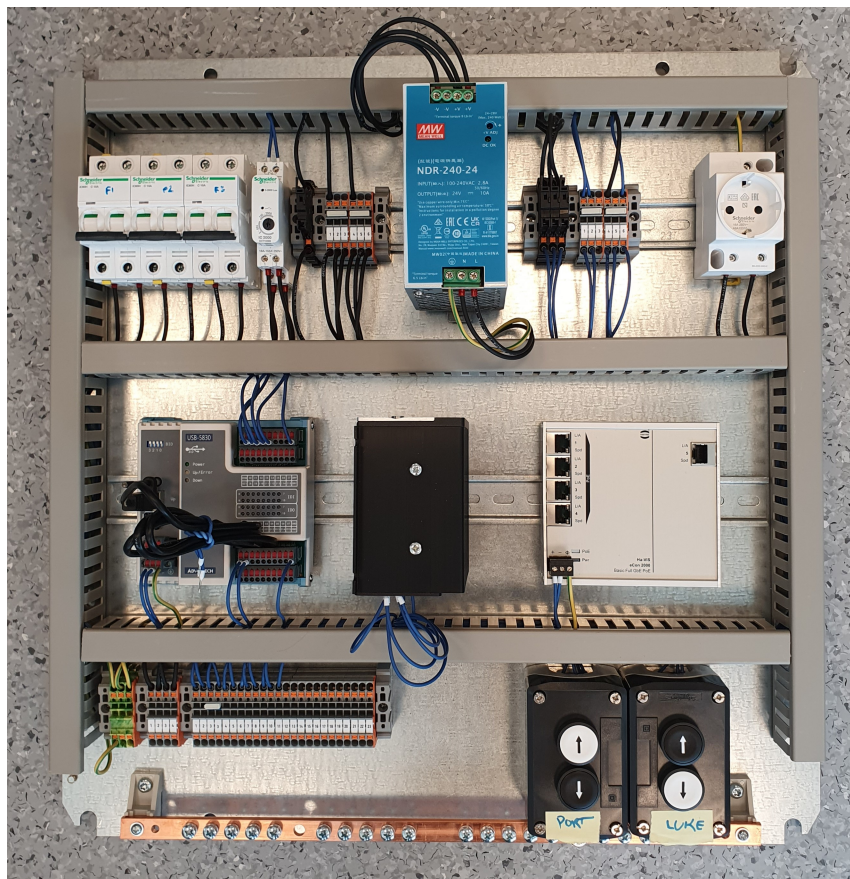
Forskjellen på disse meldingene er identifikatoren, som forteller hvem som er forventet mottaker av meldingen. Data som skal videresendes til terminalene er på heksadesimalform og er beskrevet i vedlegg **D - AFT-API grensesnitt og kommandoer**. Data til AutoPax-PC-en er i tabell 4.6.3.

Kommandoer	Beskrivelse
0x10: Start audio	Starter varsel signal
0x11: Stop audio	Stopper varsel signal
0x12: Play message	Spiller av en lyd beskjed ved blokkade av sensorer
REID / SEID	Mottaker ID / Sender ID
0x00: All	Alle enheter
0x01: Term A	Terminal A
0x02: Term B	Terminal B
0x03: Ferry	Fergen

Tabell 4.6.3: Kommandoer til AutoPax-PC.

For detaljer om kommunikasjonen mellom AutoPax-PC-en og AFT-APIet, se underseksjon 3.6.1 er i kapittel 2, Systemkomponenter.

4.6.2 El-skap tilhørende terminal



Figur 4.6.4: El-skapet sett ovenfra.

Det er blitt bygget et skap til land-terminalen for å få alle elektriske komponenter kapslet inn. Skapet kan sees i figur 4.6.4 som viser hvordan el-skapet er utformet.

Skapet er utformet etter gruppens fagkompetanse fra bygging av elektriske tavler. Valg av utstyr og farge på ledninger er også basert på dette. Elektriske skjemaer for oppkobling og rekkeklemmetabell finnes i vedlegg **H – Koblingskjema og rekkeklemmetabell, for landterminal**. Komponenter og utstyr som er brukt i skapet finnes i vedlegg **I – Innkjøp, merket med hvor varen er bestilt fra**.

4.7 Sekvensiell styring

4.7.1 Forklaring

Flytskjema for sekvensiell styring er essensielt for å forstå hvordan interaksjonen mellom ferge og land, samt manuell styring fungerer. De sekvensielle skjemaene er delt inn i to hovedkategorier, manuell styring og automatisk styring. Manuell styring blir aktivert hvis en på grunn av uforutsett hendelse må manuelt overstyre luker og porter. Automatisk styring er ved vanlig drift, og en skal ikke behøve manuelle inngripen.

De sekvensielle skjemaene er tegnet med utgangspunkt i fungerende deteksjon av kamera, hurtig og pålitelig kommunikasjon, og et operativt system. Det er ikke tatt hensyn til eventuelle strømbrydd og oppkjøring av systemet igjen, dette er løsninger som burde/må utvides etter hvert som systemet er satt i drift.

4.7.2 Manuell drift

Manuell drift er tenkt å kunne kjøres fra «shore control lab» eller ved hjelp av knappene som er koblet opp i land-PC-skapet. Sekvensen for manuell styring skal ikke være mulig å kjøre hvis en av hurtigstoppene er inne, hurtigstoppene er stoppknappen mellom portene eller hurtigstoppen på fergen. Det er lagt inn forrigling for å forhindre at portene kan åpnes hvis luken er oppe, og luken ikke kan lukkes hvis portene står i åpen posisjon. Flytskjema for manuellstyring finnes i vedlegg **C - Sekvensielle skjemaer, manuell styring.**

4.7.3 Automatisk drift

Det er designet et forslag for sekvensiell åpning og lukking i automatisk modus. Denne sekvensen er programmert på WAGO-PLS og vises i kildekoden i vedlegg **E - NMEA0183 kommando- og statusmeldinger lukestyring.** Det er en liten forskjell fra åpne og lukke sekvensen, men skiller i henhold varsling og tider. For eksempel ønsker en ved lukking av porter at passasjerene har tid til å komme seg ut av sikkerhetssonen.

Når sekvensen starter trykkes start-knappen på båten som fører til at lukkesekvensen starter. Deretter starter lydvarsling i form av piping og rødt lys med blinking i portene. Dette er for å indikere porter i bevegelse og varsle om fare. Dersom en av hurtigstoppene blir trykket i denne perioden pauser prosessen og det må en manuell inngripen til for å resette feilen. Ved en slik hendelse skal lyden slås av og portene gå over til rødt konstant lys.

Hvis alle kriterier er opprettholdt skal fergen og landporter starte lukkingen samtidig. Dersom noen beveger seg i sikkerhetssonen, eller trykker på en av hurtigstoppene, skal prosessen settes på pause. Logikken kjører en selvtest etter dette slik at prosessen kan startes igjen om objektet har kommet seg ut av sonen. Dersom hurtigstoppen trykkes, skal dette også løses ved manuell inngripen.

Når portene har lukket seg skal de spørre om status på landport eller fergeport, og bekrefte at begge porter er lukket. Dersom de ikke er lukket begge to, skal portene gå ut i en såkalt time-out, og kontrollrommet skal varsles.

Når portene er lukket og det er klart til å heise luken, skal lyden slås av og portene lyse hvitt for å fremstå som sikkert for passasjerene.

Flytskjema for de automatiske prosessene kan sees i vedlegg **B - Sekvensielle skjemaer, portstyring, ankomst og avgang.**

4.8 Risikoanalyse

RISIKO	Høy 3	3	6	9
	Middels 2	2	4	6
	Lav 1	1	2	3
		1 Lav	2 Middels	3 Høy
SANNSYNLIGHET				

Figur 4.8.1: Risikomatrise.

I alle prosjekter må det utføres en risikoanalyse. Analysen skal foretas for å fastsette risiko og sannsynlighet i alle hendelser som potensielt kan oppstå. Risikovurderingen tar utgangspunkt i oppgavebeskrivelsen, og omfatter ikke en helhetlig risikovurdering av milliAmpere2, kun styringen og sikkerheten rundt lukestyring og portstyring. For å sette opp en risikovurdering, har det blitt utviklet en risikomatrise, knyttet til dette prosjektet. Risikomatrisen med sannsynlighet og risiko kan sees i figur 4.8.1.

Sannsynligheten er oppgitt i tidsintervall, som antall passasjer per tidsintervall, og kan sees i tabell 4.8.1. Grunnen til at frekvensen er satt til passasjer, handler om fergen og dens hyppighet i drift. Det er ikke sikkert fergen vil operere til enhver tid, og ved å oppgi frekvens i forhold til antall reisende vil en få et mer perspektiv på hyppigheten.

Risikoen er satt til menneskelig og økonomisk og kan sees i tabell 4.8.2. Dette er gjort med bakgrunn i at alle skader ikke nødvendigvis treffer personer eller objekter, og kan isteden ha en økonomisk konsekvens. Risiko knyttet til mennesker skal alltid prioriteres og jobbes etter for å ha et så lavt utgangspunkt som mulig. Nivåene er tildelt etter gruppens preferanser, og er basert på et helhetsinntrykk av hvilke scenarioer som er oppnåelig på fergen.

Den økonomiske risikoen er knyttet til slitasjedeler eller deler som kunne blitt ødelagt av andre faktorer som for eksempel mekanisk påkjenning av personer. Summene som er beregnet er tatt i utgangspunkt i kostnader knyttet til oppgaven og innkjøp.

Sannsynlighet	
Nivå	Begrunnelse
1	<1 ganger pr. passasjer/ pr. år
2	>1 gang pr. passasjer/ pr. mnd
3	>1 gang pr. passasjer/ pr. dag

Tabell 4.8.1: Sannsynlighet for hendelse.

Risiko	
Nivå	Menneskelig begrunnelse
1	Skade som ikke krever inngripen. Skrubbsår, lettere skrapemerker.
2	Skade som kan kreve lettere førstehjelp. Kuttskader, enkel klemskade, dypere kutt.
3	Alvor skade som krever oppsyn hos lege/legevakt. Klemskader med varige men, operasjon eller død.
Nivå	Økonomisk begrunnelse
1	Ingen/ ubetydelig økonomisk belastning
2	Middels økonomisk belastning >5.000NOK og <20.000NOK
3	Stor økonomisk belastning >20.000NOK

Tabell 4.8.2: Risikobegrunnelser.

Selve risikoanalysen kan leses på neste side. Vurderingen tar utgangspunkt i tenkte oppståtte og sannsynlige situasjoner, og det er forsøkt å fange opp fleste mulige scenarioer. Tabellen tar inn en type hendelse som kan skje, hvilke faremomenter som kan oppstå og et tall fra 1 til 3 for sannsynlighet og risiko. Det tallet som blir skapt når en multipliserer sammen sannsynlighet og risiko, og det sier noe om hvorvidt tiltak må gjøres eller ikke. Dersom karakteren blir markert som rødt, betyr det at risiko \cdot sannsynlighet ≥ 6 , og det må gjøres tiltak. Disse tiltakene må være risiko og eller sannsynlighetsreducerende. Det må også gjøres en ny vurdering etter at tiltak er gjennomført. Denne måten å lage risikovurdering på er hentet med inspirasjon i standarden ISO12100 [75].

ID	Uønskede hendelse	Vurdering av årsak, sannsynlighet og konsekvens	Sannsynlighet	Risiko	Karakter	Risikoreducerende tiltak/ kommentar
001	Klemskade mellom porter	Fare: Objekter eller mennesker kan komme i klem mellom porter, og skade seg.	1	2	2	Risikoreducerende tiltak er å plassere portene "langt nok" fra hverandre, slik at at hender ikke kan komme i klem. Et kamera som skal se over sonen og at ingen objekter eller mennesker ligger i veien skal også varsle systemet.
002	Kamera deteksjon feiler	Fare: Objekter eller mennesker kan komme i klem og eller skade seg i ombordstigningsprosessen.	1	3	3	Ved hjelp av "shore control", skal slike situasjoner kunne ivaretas fra kontrollrom. Da vil fergen styres manuelt.
003	Person i vann mellom ferge og kai/dock	Fare: Person blir klemt og i verstefall drukne.	1	3	3	Kamera skal fange opp eventuelle personer som befinner seg mellom ferge og docking. Sannsynligheten vil være liten ettersom en aktivt må hoppe ut i sjøen, og det antas at det er redningstiger ved brygge.
004	Tannhjul, port i bevegelse	Fare: Klemfare, klær og gjenstander kan sitte fast/bli dratt inn i mekanikken.	1	2	2	Deksel blir montert for å hindre tilgang til tannhjul.
005	Strømbrudd på båt	Fare: Porter og luke kan ikke åpnes. Personer får ikke gått av fergen uten manuelt inngrep.	1	2	2	Personer med funksjonshemming vil være spesielt utsatt i en slik situasjon.
006	Strømbrudd	Fare: Porter blir ute av drift. Sikkerhetsdeteksjon ute av drift. Generell kommunikasjon blir borte. Porten kan låses i låst eller åpen posisjon som gjør at personer ikke kan gå av fergen, eller at fergen ikke går.	1	2	2	Ved strømbrudd blir deteksjonen i sikkerhetssone og styring/kontroll av porter på land satt til manuell drift.
007	Luke, opp og ned	Fare: Klemfare mellom port og ferg. Befinner seg objekter på luke som ikke er detektert. Klemfare mellom brygge og luke.	1	3	3	Luke ned: Ingen mekanisk beskyttelse, kun kameradeteksjon. Luke opp: Kun kameradeteksjon og sensor.
008	Knust pleksiglass i port	Fare: Kuttefare	1	2	2	10mm tykt pleksiplate, oppgitt for å takle klimaet og miljøet. Vanskelige å ødelegge!!!!
009	Snublekanter	Fare: Fulle mennesker. Falle og slå seg på grunn av kanter. Eldre mennesker. Funksjonshemmede.	1	3	3	Reduserende tiltak: Merking og tilstrekkelig håndløper. Lys- og lyd varsling.
010	Lys- og lydsignal	Fare: Opplevs som støyende. Lite tydelig. Informasjonen blir ikke oppfattet.	3	1	3	Tiltak: Norsk og engelsk informasjon. Kort og konsist. Indikasjon på melding, før meldingen kommer.
011	Tannhjul tagger over	Fare: Mekanisk belastning på port som fører til overtagging. Port kommer ut av stilling.	1	2	2	Tiltak: Kamera og sensor hindrer at personer står i sonen og evt. Dytter på porter.
012	Smuss på kameralinse	Fare: Vær og vind fører til at kamera ikke detekterer objekter.	2	2	4	Tiltak: Månedlig vask av beskyttelse, del av mnd. Vedlikehold sammen med fergen.
013	Kommunikasjon land - båt	Fare: Kommunikasjon mellom landterminal og båt blir borte. Båten får ikke lagt til eller hurtigstoppen ikke fungerer. Får ikke forlatt båten.	1	2	2	Tiltak: Dersom pingetiden blir for lang, skal kontrollrom varsles og eventuelt overstyrt manuelt.
014	Misbruk av sikkerhet	Fare: Trykker hurtigstopp mellom porter uten at det er reell fare. Trykker nødstopp på fergen.	1	1	1	Upraktisk med tanke på nødstopp som må manuelt tilbakestilles. Burde gjøres en manuell sjekk for å starte prosessen igjen.
015	Luke stenger med åpne porter.	Fare: Portene blir ødelagt. Personer sitter eventuelt fast på fergen.	1	2	2	Styringssystemet skal være programmert med forrigling.
016	Porter åpner med luke oppe.	Fare: Portene blir ødelagt. Personer sitter eventuelt fast på fergen.	1	2	2	Styringssystemet skal være programmert med forrigling.
017	Fuktskader	Fare: Hurtig slitasje av skader. Korrosjon fører til brudd og ødelagte deler. Elektronikk kan bli ødelagt.	1	2	2	Tiltak: Utstyr og design er tilpasset eventuell fukt og følge opp en viss IP-grad for kapsling.
018	Vær og vind, ødelegger utstyr	Fare: Saltvann i luft, snø, kulde, tærer på utstyr.	2	2	4	Tiltak: Utstyr og design er tilpasset eventuell fukt og følge opp en viss IP-grad for kapsling. Jevnlige renhold og vedlikehold senker sannsynligheten.
019	Elektrisk skap på fergekai	Fare: Uautorisert tilgang. Elektrisk spenning, liv kan gå tapt.	1	3	3	Merket for uautorisert personell, skal være vanskelig å komme inn i skapet uten tilgjengelig nøkkel.

Kapittel 5

Konklusjon og videre arbeid

5.1 Konklusjon

Oppgaven har fokusert på utviklingen av et passasjerhåndteringssystem for den urbane autonome fergen, milliAmpere2. Gjennomføringen av oppgaver har tatt lengere tid enn forventet grunnet bestilling av deler og et fag som gikk parallelt med bacheloroppgaven frem til slutten av mars. Dette har skapt utfordringer med å få deler i tide og montert de på fergen.

Til tross for tidsrammen er rammeverket for fergeporten og el-skapet blitt satt sammen, og klargjort for testing og senere montering på fergen. Videre er det utarbeidet et forslag til systemarkitekturen, men dette er ikke testet. De bestilte delene er enten mottatt eller på vei, og det er derfor mulig å montere systemet og teste passasjerhåndteringssystemet når resterende deler kommer.

Passasjerhåndteringssystemet har blitt utformet for å dekke behovene som en sikkerhetsvert ellers ville ha måtte håndtert. For å unngå fare for at passasjerer faller i sjøen under på- og avstigning har det blitt laget to porter til fergen, og bestilt to porter til land. Det er brukt LED-lys i rammen til fergeporten som da skal visuelt informere passasjerene om prosedyren når fergen legger til eller fra kai. Lysene som er montert i portene skal operere som et lyskryss, rødt for stopp og vent, og grønt for å gå.

Sikkerhet er en sentral del av systemet, og ved bruk av deler som beveger seg kan det oppstå farlige og uønskede situasjoner. Det skal derfor monteres et kamera i en bøyse som har oversikt over området der lemmen og portene beveger seg. Dette kameraet stopper systemet om det står passasjerer i dette området når lem eller porter utfører bevegelser. Det er lagt opp til at dette kameraet også kan telle passasjerer ved en utvidelse.

For at komponentene i passasjerhåndteringssystemet skal kunne samarbeide med det eksisterende systemet som er om bord på fergen, er det nødvendig med kommunikasjon mellom de to systemene. Det er utarbeidet et forslag til kommunikasjonsmetode, men dette har ikke blitt testet ut. Passasjerhåndteringssystemet skal kunne kommunisere med en landbasert terminal så eventuelle passasjerer kan bestille fergen over til ønsket side. Det er en annen bachelorgruppe som har laget landterminalen [4]. Gruppene har jobbet parallelt for å sikre god kommunikasjon mellom ferge og land.

Avslutningsvis har planleggingen og forslaget til passasjerhåndteringssystemet en stor betydning for å sikre passasjerenes trygghet under på- og avstigning. Planleggingsprosessen har vært sentral for å finne sikkerhetsbehovene komponentene må ha for at passasjerene skal være trygge og ikke føle ubehag. Komponentenes funksjon og utseende fungerer også som informasjon for passasjerene, og gir oversikt over hendelsesforløpet ved ankomst og avgang.

Dette systemet er en betydningsfull del av videreutviklingen til milliAmpere2 som en trygg, autonom ferge som opererer i urbane områder.

5.2 Videre arbeid

5.2.1 Fergeport

Etterarbeidet som gjelder fergeportene vil kunne finne sted så snart de nødvendige bestilte delene har ankommet. Under veiledermøtet den 11.05.2023 besluttet gruppen sammen med veiledere den videre gangen i hvordan portene kan monteres.



Figur 5.2.1: Tannhjulet som skal sveises inn i rammen på fergeporten.

Monteringen skal ta utgangspunkt i designtegningene som gruppen har tegnet. Designtegningene som er lagt ved i vedlegg **A - GitHub, kode og Fusion 360-filer**. Når fergeportene er montert, må de kobles til WAGO PLS-en om bord i fergen. Programmet skal være klart, men er ikke testet ut.

Det er bekreftet at gruppen ikke rekker å montere portene på fergen i løpet av bachelortiden, men det kan være aktuelt for gruppemedlemmer å fortsette arbeidet utover sommeren.

5.2.1.1 Videre arbeid med aktuator

Slaglengden til aktuatoren må endres fra 100.00 mm til 22.54 mm. En kan endre enten start eller endepunktet til aktuatoren ved bruk av en BusLink-kabel. Se datablad til BusLink av LINAK for hvordan en kobler BusLink-kabelen til aktuator og PC [66]. I databladet får en også en gjennomgang på hva som må gjøres for å endre endestoppet.

Det er også ved bruk av BusLink-kabelen en kan endre hastigheten til aktuatoren. Grunnet forholdet mellom de to tannhjulene vil det minste tannhjulet, altså porten, rotere dobbelt så fort som hastigheten til aktuatoren. Dette vil si at om aktuatoren bruker ett sekund på å gå ut 22.54 mm så roterer porten 90° på under 0.50 sekunder.

Derfor må hastigheten til aktuatoren endres. Dette gjøres ved å redusere maks hastigheten prosentvis. For å gjøre denne endringen kan en følge anvisningen [66]. LINAK informerer om at ved en hastighet på under 60% av maks hastighet kan den oppgitte lastkapasiteten til aktuatoren bli redusert.

5.2.1.2 Montering av tannhjul i baugkasse

Tannhjulet som er plassert inne i baugkassen burde monteres etter skisseringen i Fusion 360 for å oppnå samme resultat.

Tannhjulet må ha en plate som det roterer på, som da må smøres etter behov. Denne platen kan sveises fast i baugkassen. For at tannhjulet som er koblet til aktuatoren ikke skal rotere fritt om i baugkassen må det sveises en stang fast som tannhjulet roterer rundt. Stangen må ha en diameter på 14.9 mm slik at tannhjulet får rom til å rotere.

Det må sveises en stang på flaten av tannhjulet. Stangen burde ha en diameter på 10.10 ± 0.1 for å kunne passe skaftet til aktuatoren som er på 10.20 ± 0.1 . Stangen burde være 25.00 mm lang. Se figur 3.1.13 for hvor på tannhjulet stangen bør monteres.

Platen som aktuatoren skal monteres fast til må monteres i veggen til baugkassen. I denne platen må det være et hull med en diameter på 10.20 ± 0.1 mm. Monter fast en skrue som gjør det mulig for aktuatoren å rotere om. Bruk mutter på hver side. La avstanden mellom kassen inne i baugkassen og platen være på 63.60 mm.

5.2.2 Landport

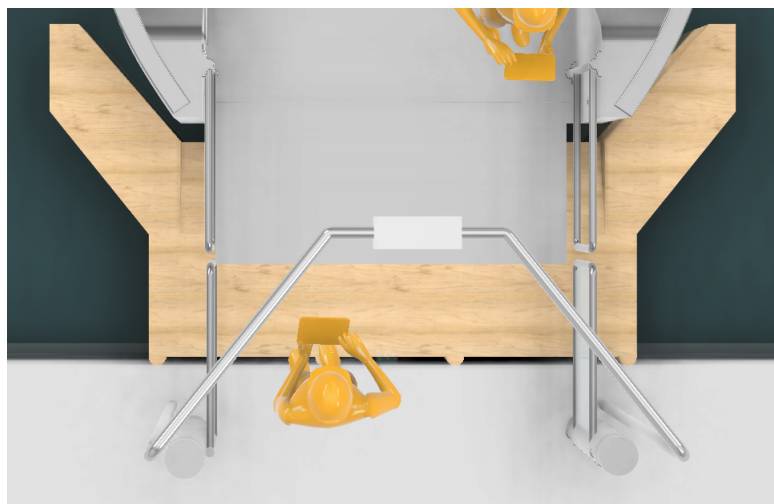
Det som gjenstår når det gjelder landporten er den fysiske monteringen på fergekaien, samt programmeringen av den. Det er bestilt nødvendige deler og komponenter for at monteringen skal kunne skje. Porten er forventet å ankomme i slutten av mai, og de resterende deler allerede på plass.

Porten trenger å kobles opp mot el-skapet som skal ligge på fergekaien. Den trenger 230 VAC strømtilførsel. Til å styre porten trenger en kommunikasjon gjennom RS-485. Da kan en velge hvilken hastighet porten skal ha. Denne hastigheten bør bestemmes lik fergeportens, men er også noe som burde testet ut for å finne ut hva som fungerer best.

5.2.3 Montering av sikkerhetsbøyle over sikkerhetssone

For å ferdigstille sikkerhetsbøylene og sikkerhetssonen må programvare fra land-terminal-PC-en monteres og testes. Det er tenkt at sikkerhetsbøylene skal være laget av 100 mm rustfritt rør som skal sveiset slik at kamera og lys blir stående midt mellom portene. Kassen, hvor kameraet skal være, skal også lages i rustfritt stål og lyslistene skal være kapslet inn bak frosted plast. Forslag til montering kan sees i figur 5.2.2.

Informasjonen som skal hentes ut fra sikkerhetssonen må gjøres via en binding-programvare. Bibliotek og hvordan en går frem for å løse dette kan finnes på https://github.com/NVIDIA-AI-IOT/deepestream_python_apps.



Figur 5.2.2: Forslag til montering av sikkerhetsbøyle.

5.2.4 El-skap

El-skapet som har blitt bygget må monteres ved landterminalen etter gjeldende regelverk som installasjon av elektrisk utstyr og dekkes til med tak. Taket skal hindre unødvendig eksponering for klimaet i Trondheim.

Det gjenstår å lage kabelgjennomføring for kabler som skal kobles til i el-skapet. Dette gjelder hovedsakelig tilførsel, lys i sikkerhetsbøyle, kommunikasjon med bestillingsterminalen, Ethernet-kabel til kamera, hurtigstopp og landport. Når konnektorene til PC-en kommer, må disse også kobles opp. Dette skal gjøres etter koblingskjema i vedlegg **H - Koblingskjema og rekkeklemmetabell for landterminal**.

5.2.5 Speiling av systemet

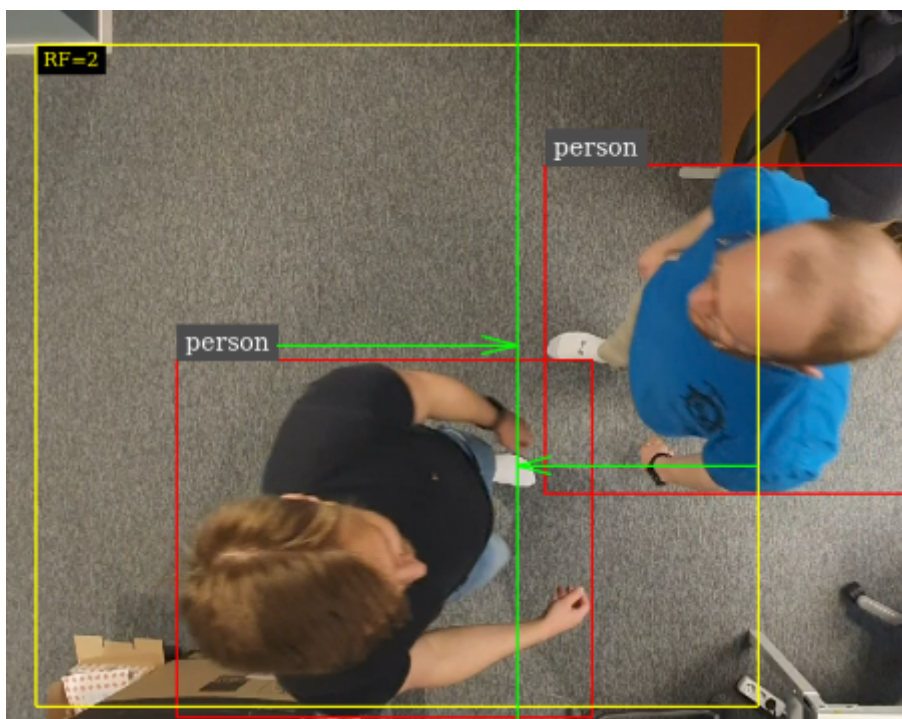
Det er i denne omgang bare lagt opp til montasje av passasjerhåndteringssystemet på en side av kanalen og fergen. Dette vil si at det bare er bestilt inn deler til en side. For å kunne lage et likt system på andre siden må bestilles inn igjen, ved unntak av delene som er tilhørende AutoPAX-bestillingene. Se vedlegg **I - Innkjøp, merket med hvor varen er bestilt** for å se hva som må bestilles inn for å lage et likt system på andre siden.

5.2.6 Programmering

Det er ønskelig å utvikle et system som detekterer antall passasjerer som har gått om bord eller av fergen. Det er lagt til rette for dette på PC-en på land, men det er ikke programmert opp. For å få dette til må det også programmeres inn følgende av objekter.

Logikken for trigging av landport og alt det sekvensielle rundt styring og sending og mottakelse av signaler må programmeres på land-PC-en.

PC-en som står om bord, referert til som AutoPAX-PC, må settes og kobles opp. Denne er ikke blitt satt opp grunnet mangel på koblingspunkter og tid.



Figur 5.2.3: Forslag til videreutvikling for telling av passasjerer.

5.2.7 Testing

Grunnet begrensning i tid og levering av utstyr har det ikke blitt utført kommunikasjonstest med bachelorgruppen for elektronisk systemdesign. Testing av kommunikasjon må skje før en setter i gang drift og bevegelse av systemkomponenter.

Referanser

- [1] Zeam. «*Making urban life flow*,» Zeam.se. (2023), adresse: <https://zeam.se/> (sjekket 16.05.2023).
- [2] I. Haugan. «*Selvkjørende ferge – først i verden med prøvedrift*,» Gemini.no. (21. sep. 2022), adresse: <https://gemini.no/2022/09/selvkjorende-ntnu-ferge-starter-verdens-forste-provedrift/> (sjekket 05.04.2023).
- [3] Fremtidens Industri. «*NTNU MilliAmpere 2 Kick-Off - World's first autonomous passenger ferry in operation*,» Youtube.com. (3. okt. 2022), adresse: <https://www.youtube.com/watch?v=j3v47HiJmos> (sjekket 13.05.2023).
- [4] I. Evenstad, T. J. Engum, M. D. Oseid og P. Wandsvik, «*Trådløs terminal for «on-demand» bestilling av autonom ferge*,» mai 2023.
- [5] L. Digerud og A. N. Fagerhaug, «*Design av et automatisk dokkingsystem for en liten autonompassasjerferge*,» årg. 1, mai 2019. adresse: <https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/2613407> (sjekket 04.2023).
- [6] H. M. Ellingsen og P. K. Glesaaen, «*Design av brukerreise og brygger til autonom passasjerferge*,» årg. 1, jun. 2020. adresse: <https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/2776805> (sjekket 03.2023).
- [7] Statens vegvesen. «*AutoPASS for ferje*,» Autopass.no. (31. jan. 2023), adresse: <https://www.autopass.no/no/kunde/ferjer/> (sjekket 16.05.2023).
- [8] Mairs Turnstile. «*Electric Gates*,» Mairsturnstile.com. (2023), adresse: <https://www.mairsturnstile.com/> (sjekket 16.05.2023).
- [9] L. Mæhlum. «*3D-printing*,» SNL.NO. (13. okt. 2021), adresse: <https://snl.no/3D-printing> (sjekket 03.05.2023).
- [10] V. Brøtan. «*Additiv produksjon kan gi mer bærekraftige produkter med mindre avfall*,» SINTEF.NO. (8. mar. 2021), adresse: <https://blogg.sintef.no/vareproduksjon-nb/additiv-produksjon-barekraft/> (sjekket 03.05.2023).
- [11] 3DNet. «*PETG 1.75*,» 3DNET.NO. (2023), adresse: <https://3dnet.no/products/petg-1-75-1-0-kg> (sjekket 29.04.2023).
- [12] Emily. «*PETG vs ABS: Which Filament Should You Use?*» Clevercreations.org. (13. mar. 2023), adresse: <https://clevercreations.org/petg-vs-abs-filament-comparison/> (sjekket 29.04.2023).
- [13] GaoXiang. «*The working principle of electric linear actuator*,» Electric-linear-actuators.com. (24. nov. 2021), adresse: <https://www.electric-linear-actuators.com/en-ca/blogs/news-1/the-working-principle-of-electric-linear-actuator> (sjekket 16.05.2023).
- [14] Heason Technology. «*How does a Linear Actuator work?*» Heason.com. (24. jan. 2020), adresse: <https://www.heason.com/news-media/technical-blog-archive/how-does-a-linear-actuator-work-> (sjekket 26.04.2023).
- [15] M. Spong, S. Hutchinson og M. Vidyasagar, *Robot Modeling and Control* (Second Edition). Wiley, 2020, ISBN: 978-1119523994.
- [16] D. Yadav. «*Slider Crank Mechanism: Theory and Applications*,» Byjusexamprep.com. (21. des. 2022), adresse: <https://byjusexamprep.com/slider-crank-mechanism-i> (sjekket 17.04.2023).

- [17] Engineers Edge. «*Piston Slider Crank Mechanism Design Equations and Calculator,*» Engineersedge.com. (2016), adresse: https://www.engineersedge.com/mechanics_machines/piston_slider_crank_mechanism_14925.htm (sjekket 17.04.2023).
- [18] Engineers Edge. «*Design Equations and Formula Circular Pitches and Equivalent Diametral Pitches Table,*» Engineersedge.com. (2016), adresse: https://www.engineersedge.com/gear_formula.htm (sjekket 24.04.2023).
- [19] Engineering Tutorials. «*Terminology for Spur Gears | Kinematics of Machines Tutorials,*» Myindialist.com. (20. mar. 2013), adresse: <https://engineering.myindialist.com/2013/terminology-for-spur-gears-kinematics-of-machines-tutorials/#.ZEj9hnZBw2z> (sjekket 24.04.2023).
- [20] Industrial Quick Search. «*Spur Gears,*» Iqsdirectory.com. (2023), adresse: <https://www.iqsdirectory.com/articles/gear/spur-gears.html> (sjekket 24.04.2023).
- [21] Autodesk. «*Fusion 360: det er mer enn CAD, det er fremtiden for design og produksjon,*» Autodesk.no. (2023), adresse: <https://www.autodesk.no/products/fusion-360/overview?term=1-YEAR&tab=subscription> (sjekket 26.04.2023).
- [22] L. Fuentes. «*What Is Fusion 360? - Simply Explained,*» All3dp.com. (21. nov. 2022), adresse: <https://all3dp.com/2/what-is-fusion-360-simply-explained/> (sjekket 26.04.2023).
- [23] L. Schardon. «*What is Unity? - A Guide for One of the Top Game Engines,*» Gamedevacademy.org. (2023), adresse: <https://gamedevacademy.org/what-is-unity/> (sjekket 26.04.2023).
- [24] The MathWorks Inc. «*Introduction to Deep Learning: What are Convolutional Neural Networks?*» Mathworks.com. (24. mar. 2017), adresse: <https://www.mathworks.com/videos/introduction-to-deep-learning-what-are-convolutional-neural-networks--1489512765771.html> (sjekket 18.04.2023).
- [25] G. Jocher, A. Chaurasia og Laughing. «*Ultralytics YOLOv8,*» Ultralytics.com. (9. mai 2023), adresse: <https://docs.ultralytics.com/> (sjekket 22.04.2023).
- [26] Z. Keita. «*YOLO Object Detection Explained,*» Datacamp.com. (2023), adresse: <https://www.datacamp.com/blog/yolo-object-detection-explained> (sjekket 18.05.2023).
- [27] TheCloudStrap. «*What is RS422 protocol? | Explained,*» Thecloudstrap.com. (11. feb. 2021), adresse: <https://thecloudstrap.com/what-is-rs422-protocol-explained/> (sjekket 29.04.2023).
- [28] S. Campbell. «*Basics of UART Communication,*» Circuitbasics.com. (14. nov. 2021), adresse: <https://www.circuitbasics.com/basics-uart-communication/> (sjekket 25.04.2023).
- [29] TheCloudStrap. «*What is the RS485 protocol | Explained,*» Thecloudstrap.com. (14. mar. 2021), adresse: <https://thecloudstrap.com/what-is-the-rs485-protocol-explained/> (sjekket 25.04.2023).
- [30] Cloudflare. «*What is UDP?*» Cloudflare.com. (2022), adresse: <https://www.cloudflare.com/learning/ddos/glossary/user-datagram-protocol-udp/> (sjekket 09.05.2023).

- [31] Versa technology. «*What is Power Over Ethernet (PoE), and What is it Used for?*» Versatek.com. (6. feb. 2023), adresse: <https://versatek.com/what-is-power-over-ethernet/> (sjekket 09.05.2023).
- [32] Wikipedia. «*DMX512*,» Wikipedia.org. (10. mar. 2023), adresse: <https://en.wikipedia.org/wiki/DMX512> (sjekket 25.04.2023).
- [33] Wikipedia. «*NMEA 0183*,» Wikipedia.org. (1. apr. 2023), adresse: https://en.wikipedia.org/wiki/NMEA_0183 (sjekket 25.04.2023).
- [34] B. C. Holtan, *Personlig kommunikasjon, samtale med ansatt fra Astrup*, feb. 2023.
- [35] American Iron and Steel Institute. «*The Voice of the American Steel Industry*,» Steel.org. (2023), adresse: <https://steel.org> (sjekket 16.05.2023).
- [36] J. K. Solberg. «*Styrefast stål*,» SNL.NO. (16. okt. 2022), adresse: https://snl.no/syrefast_st%C3%A5l (sjekket 15.05.2023).
- [37] Jens S. Transmisjoner AS. «*Sylindriske tannhjul moduler*,» Jens-s.no. (2023), adresse: <https://www.jens-s.no/media/159816/sylindriske-tannhjul.pdf> (sjekket 16.04.2023).
- [38] E. J. Hareide, *Personlig kommunikasjon, Teamsmøte med Einar Johan Hareide*, mar. 2023.
- [39] Forecr. «*JetPack-5.1 Installation for DSBOX-NX2*,» Forecr.io. (1. apr. 2023), adresse: <https://www.forecr.io/blogs/installation/jetpack-5-1-installation-for-dsbox-nx2> (sjekket 22.04.2023).
- [40] NVIDIA Corporation. «*Gst-nvdsanalytics*,» Nvidia.com. (2023), adresse: https://docs.nvidia.com/metropolis/deepstream/dev-guide/text/DS_plugin_gst-nvdsanalytics.html (sjekket 18.05.2023).
- [41] L. Watson, «*How to install Python examples and run it on both Windows and Linux*,» årg. 1, adresse: https://downloadt.advantech.com/download/downloads.aspx?File_Id=1-23XZB6D (sjekket 22.04.2023).
- [42] C. Liechti. «*pySerial, Python serial port access library*,» GitHub.com. (21. sep. 2021), adresse: <https://github.com/pyserial/pyserial> (sjekket 18.04.2023).
- [43] Forecr. «*DSBOX-N2 Nano industrial box PC*,» Forecr.io. (2023), adresse: <https://hs.forecr.io/hubfs/DATASHEETS/dsbox-n2-datasheet-v1.0.pdf> (sjekket 16.05.2023).
- [44] WAGO. «*WAGO I/O System 750*,» Wago.com. (2023), adresse: <https://www.wago.com/medias/m07508212-xxxxxxx-0en.pdf?context=bWFzdGVyfGRvd25sb2Fkc3w0NzY3Mjc1fGFwcGxpY2F0aW9uL3%20BkZnxoM2UvaGYyLzEzNzQ0TMzMzEzMTU4L20wNzUwODIxM194eHh4eHh%204eF8wZW4ucGRmfDI4MzI2MDNjMzZlZGE5NjY4MTU4ODJmODI0NGQxNzB%201Zjc0N2I2ZTU0YzExMzIxZGM0NGVlNGJiM2U4NGFlZDU&attachment%20=true> (sjekket 16.05.2023).
- [45] LINAK. «*Actuator LA25, Data sheet*,» Linak.com. (des. 2020), adresse: <https://cdn.linak.com/-/media/files/data-sheet-source/en/linear-actuator-la25-data-sheet-eng.pdf> (sjekket 11.05.2023).
- [46] The Light Group AS. «*SLC DMX controller CV 4x5A 12-36V RGBW DIN*,» TLG.NO. (mai 2023), adresse: https://www.tlg.no/product-fdv/create?product=10384&comment=&locale=en_GB (sjekket 16.05.2023).
- [47] Mouser Electronics. «*Visaton FR 8 WP*,» Mouser.com. (2023), adresse: https://no.mouser.com/datasheet/2/700/FR_8_WP_1-2322433.pdf (sjekket 16.05.2023).

- [48] Marco Marine tech. «*Electronic whistle with two fog signals, microphone, siren,*» Marco.it. (25. aug. 2021), adresse: https://items.marco.it/media//attach/booklet/EW2-MS_00_ENG.pdf (sjekket 16.05.2023).
- [49] Creative. «*Sound Blaster PLAY!3,*» Floralimited.com. (2016), adresse: <https://floralimited.com/Files/Products/CREATIVE/BrochureUrl/Sound%20Blaster%20Play%203.pdf> (sjekket 16.05.2023).
- [50] Mairs Turnstile. «*Supermarket swing gate turnstile MT311,*» Mairsturnstile.com. (2023), adresse: <https://mairsturnstile.com/product/supermarket-swing-gate> (sjekket 16.05.2023).
- [51] Forecr. «*DSBOX-NX2 Xavier NX industrial box PC,*» Forecr.io. (2023), adresse: <https://hs.forecr.io/hubfs/DATASHEETS/dsbox-nx2-data-sheet-v1.0.pdf> (sjekket 16.05.2023).
- [52] Advantech. «*16-ch Isolated Digital Input and 16-ch Isolated Digital Output USB 3.0 I/O module,*» Advantech.com. (27. jun. 2018), adresse: <https://advdownload.advantech.com/productfile/PIS/USB-5830/Product%20-%20Datasheet/USB-583020180910101909.pdf> (sjekket 16.05.2023).
- [53] H. Technologie. «*Ha-VIS eCon 2000 Full Gigabit Ethernet Basic PoE / PoE+,*» RS-online.com. (apr. 2014), adresse: <https://docs.rs-online.com/b841/0900766b8131342d.pdf> (sjekket 16.05.2023).
- [54] Schneider Electric SE. «*Acti 9 skumringsrele - 1NO - 2..2000 lux med utendørs lyssensor,*» SE.COM. (20. mai 2023), adresse: <https://www.se.com/no/no/product/download-pdf/CCT15369> (sjekket 21.05.2023).
- [55] ABUS. «*ABUS IP video surveillance 4MPx mini dome camera,*» Elfadistrelec.no. (jul. 2022), adresse: https://www.elfadistrelec.no/Web/Downloads/_t/ds/TVIP44510_eng_tds.pdf (sjekket 18.05.2023).
- [56] US converters LLC. «*Serial Data Communication,*» Usconverters.com. (2023), adresse: https://www.usconverters.com/?main%5C%5C_page=page%5C%5C&id=35%5C%5C&chapter=0 (sjekket 18.05.2023).
- [57] Forcer. «*DSBOX-N2 Serial Communication Interfaces Tutorial,*» Forecr.io. (2023), adresse: <https://www.forecr.io/blogs/connectivity/dsbox-n2-serial-communication-interfaces-tutorial> (sjekket 18.05.2023).
- [58] LINAK. «*Actuator LA25, IC Parallel: Connection diagram,*» Linak.com. (des. 2021), adresse: <https://cdn.linak.com/-/media/files/connection-diagrams/la25/actuator-with-parallel.pdf> (sjekket 11.05.2023).
- [59] The Light Group AS. «*Universal series DMX512 decoder,*» cloudinary.com. (2023), adresse: https://res.cloudinary.com/thelightgroup/image/upload/v1641280254/p_S26005_documentation_1 (sjekket 18.05.2023).
- [60] WAGO. «*Wago Serial Interface RS-232 / RS-485,*» Wago.com. (2017), adresse: <https://www.wago.com/medias/m07500652-xxxxxxx-0en.pdf?context=bWFzdGVyfGRvd25sb2Fkc3wzODA1NjE3fGFwcGxpY2F0aW9uL3%20BkZnxoY2EvaDFmLzEzNjI1NzAyNjQ1NzkwL20wNzUwMDY1Ml94eHh4eHh4eF8wZW4ucGRmfGJlNGYwOGFlZjNmYjBiYTBiOTk2NjA1MDAzYWI4ODMyNDVkJZWEzNzgxZjM3MGYyNjg2MTUyYTRhMGUwMjZkNGY&attachment=true> (sjekket 18.05.2023).
- [61] Crown Audio. «*How Much Amplifier Power Do I Need,*» Crownaudio.com. (2023), adresse: <https://www.crownaudio.com/how-much-amplifier-power> (sjekket 18.05.2023).

- [62] Jentsentone. «*Speaker Impedance, Power Handling, and Wiring*,» Jentsentone.com. (2023), adresse: <https://www.jentsentone.com/speaker-impedance-power-handling-and-wiring> (sjekket 18.05.2023).
- [63] M. Electronics. «*Advantech USB-5830-AE*,» Mouser.com. (2023), adresse: <https://no.mouser.com/ProductDetail/Advantech/USB-5830-AE?qs=MLItCLRbWsyXYOzPxbz05A%3D%3D/> (sjekket 21.05.2023).
- [64] Solar Norge, *STRIPLINE 5M 15W/827 2X5M KABEL*, Solarnorge.no. adresse: <https://www.solarnorge.no/no-webshop/lighting/interior-luminaires/light-ribbon-hose-strip/1200486485/> (sjekket 18.05.2023).
- [65] Ladelys AS, *LED-Gatelys*, Ladelys.no. adresse: <https://www.ladelys.no/news/led-gatelys/> (sjekket 15.05.2023).
- [66] LINAK. «*BusLink: User Guide*,» Linak.com. (jan. 2019), adresse: <https://cdn.linak.com/-/media/files/brochure-source/en/techline-buslink-quick-guide-brochure-eng.pdf> (sjekket 11.05.2023).
- [67] NVIDIA Corporation. «*Jetson Xavier NX Series Modules*,» Nvidia.com. (2023), adresse: <https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-xavier-nx/> (sjekket 18.05.2023).
- [68] NVIDIA Corporation. «*NVIDIA DeepStream SDK*,» Nvidia.com. (2023), adresse: <https://developer.nvidia.com/deepstream-sdk> (sjekket 18.05.2023).
- [69] NVIDIA Corporation. «*NVIDIA TensorRT SDK*,» Nvidia.com. (2023), adresse: <https://developer.nvidia.com/tensorrt> (sjekket 18.05.2023).
- [70] L. Dissayanake. «*Deploy YOLOv8 on NVIDIA Jetson using TensorRT and DeepStream SDK*,» Seeedstudio.com. (21. apr. 2023), adresse: <https://wiki.seeedstudio.com/YOLOv8-DeepStream-TRT-Jetson/> (sjekket 23.04.2023).
- [71] Proshop AS, *ABUS TVIP44510*, Proshop.no. adresse: <https://www.proshop.no/Overvaakningskamera/ABUS-TVIP44510-network-surveillance-camera-dome/3075671> (sjekket 25.04.2023).
- [72] N. H. Bua, *Personlig kommunikasjon, Teamsmøte med Sjøfartsdirektoratet*, mar. 2023.
- [73] S. Papageorgiou, *Personlig kommunikasjon, e-post med Sjøfartsdirektoratet*, apr. 2023.
- [74] NTPHX. «*WAGO 750-8202 DMX302 TRIAC Dimmer controller*,» Youtube.com. (4. apr. 2020), adresse: <https://www.youtube.com/watch?v=HaevbBvAc10> (sjekket 19.05.2023).
- [75] «*Maskinsikkerhet - Hovedprinsipper for konstruksjon - Risikovurdering og risikoreduksjon*,» Standard Norge, Oslo, NO, Standard, jan. 2011. adresse: <https://www.standard.no/nettbutikk/sokeresultater/?search=iso12100&subscr=1> (sjekket 18.05.2023).

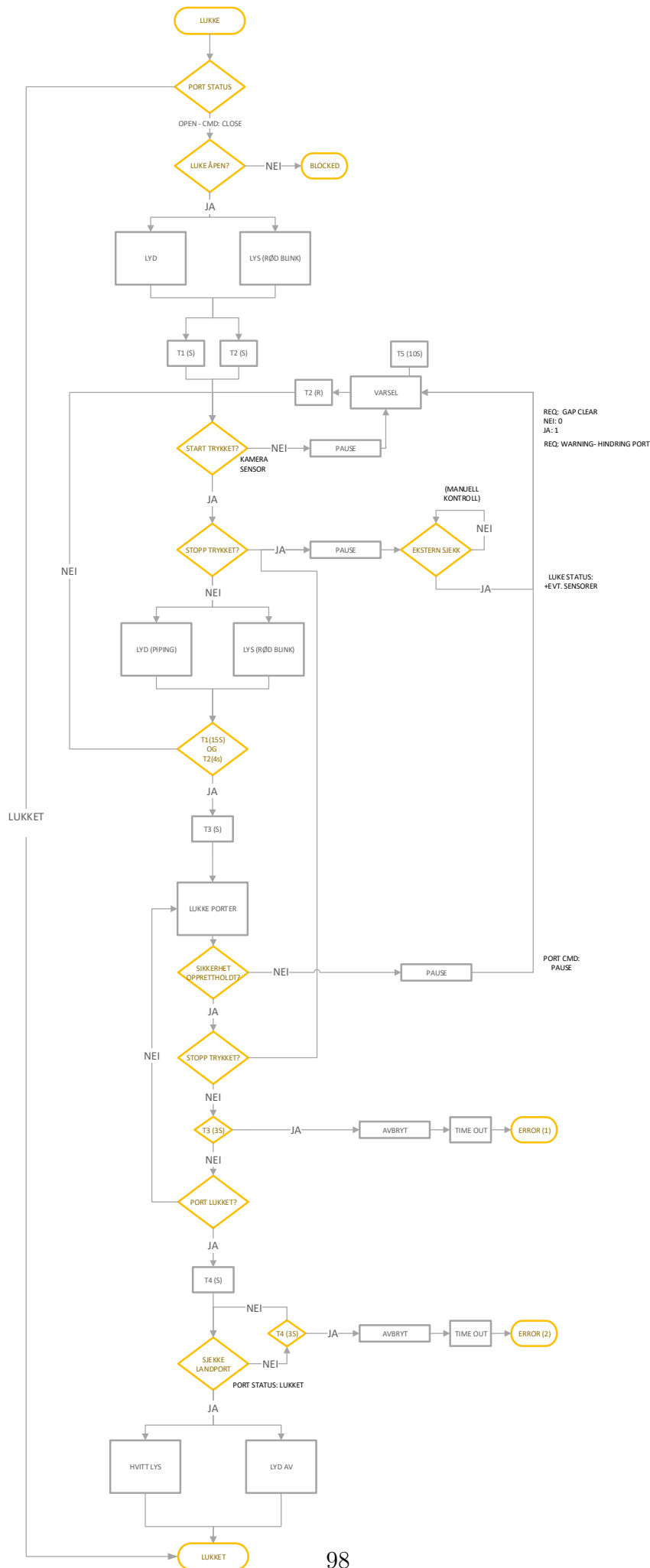
Vedlegg

A - GitHub, kode og Fusion 360-filer

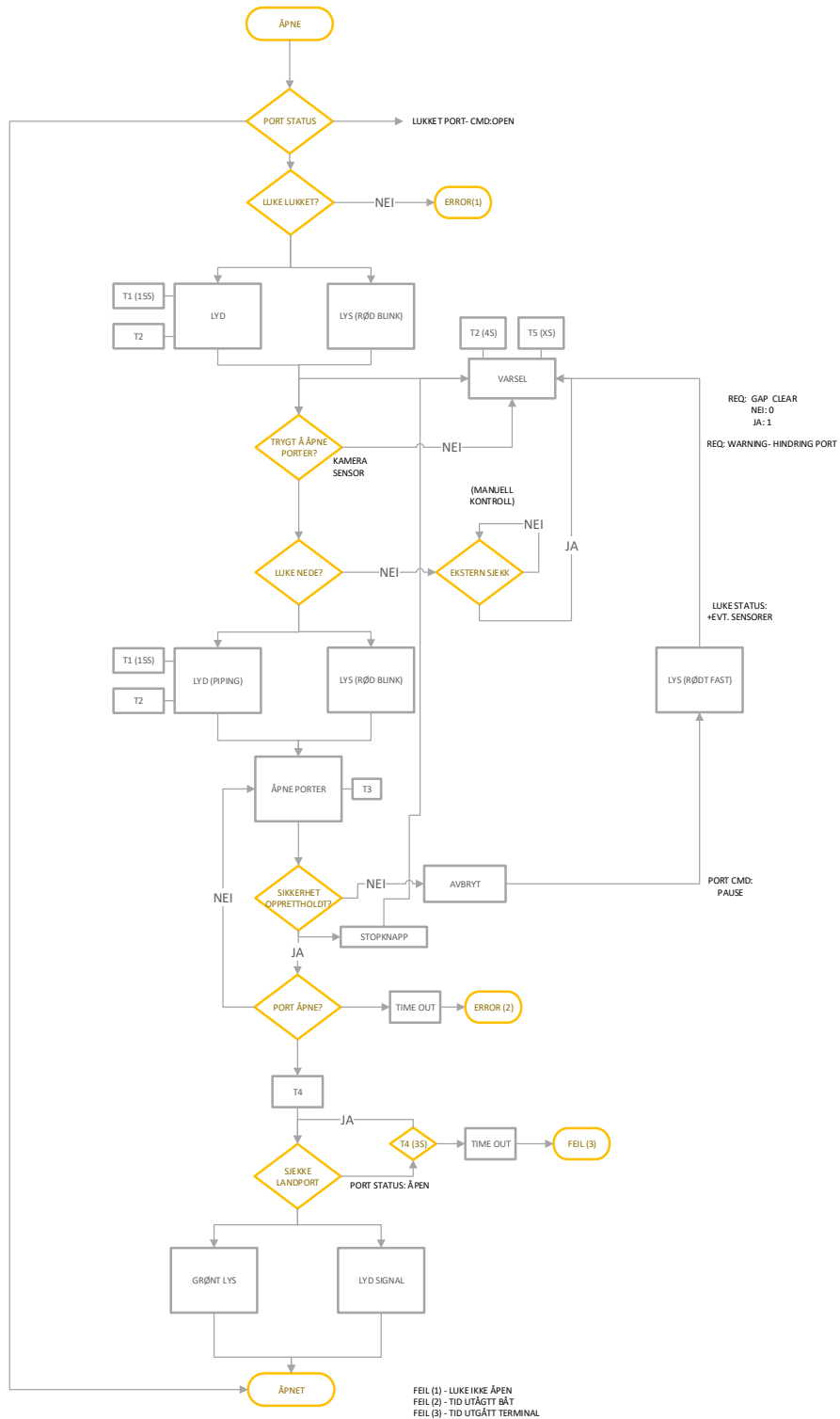
GitHub - AutoPAX Repository
<https://github.com/anvi16/AutoPax>

B - Sekvensielle skjemaer, portstyring, ankomst og avgang

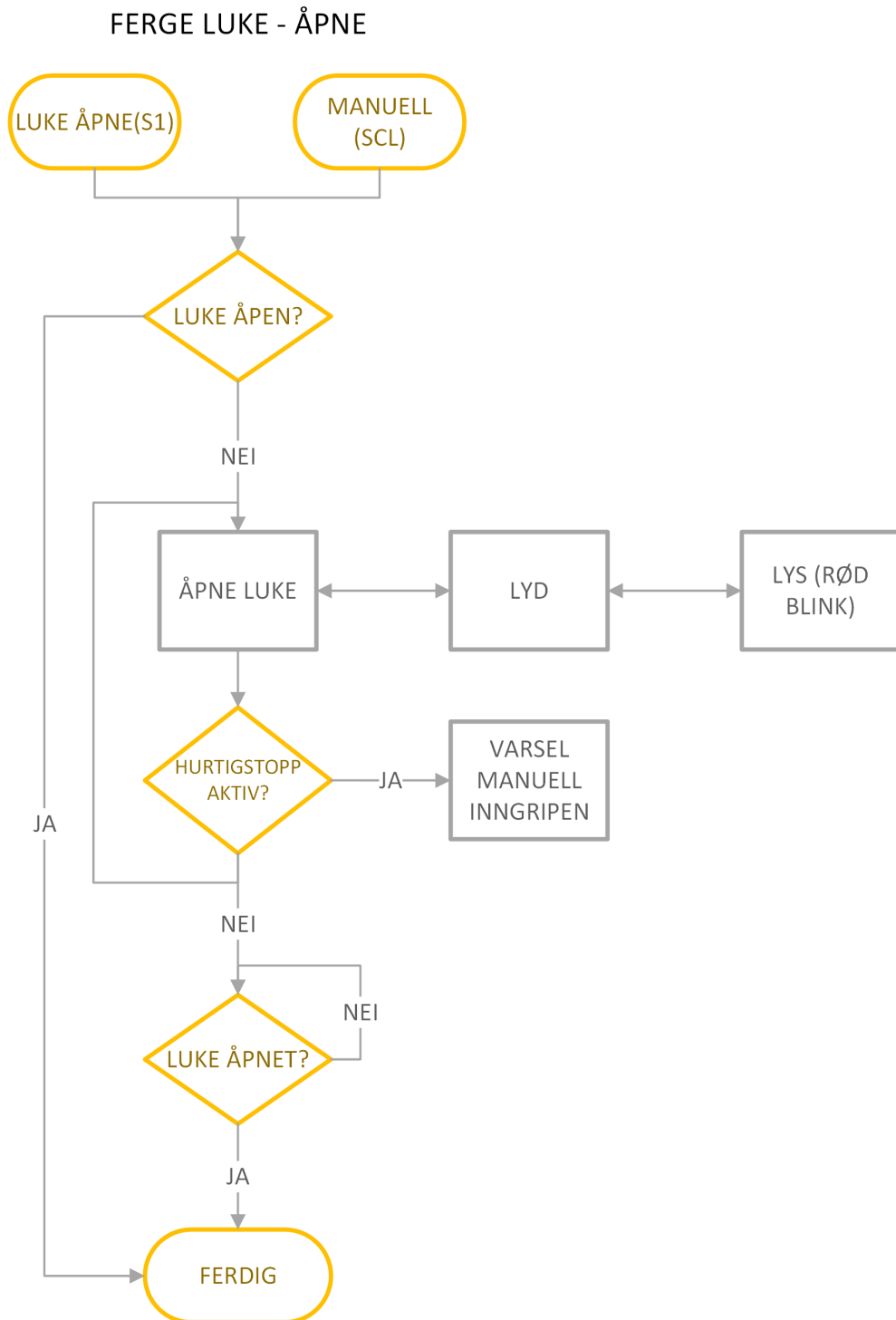
PORT - LEGGE FRA



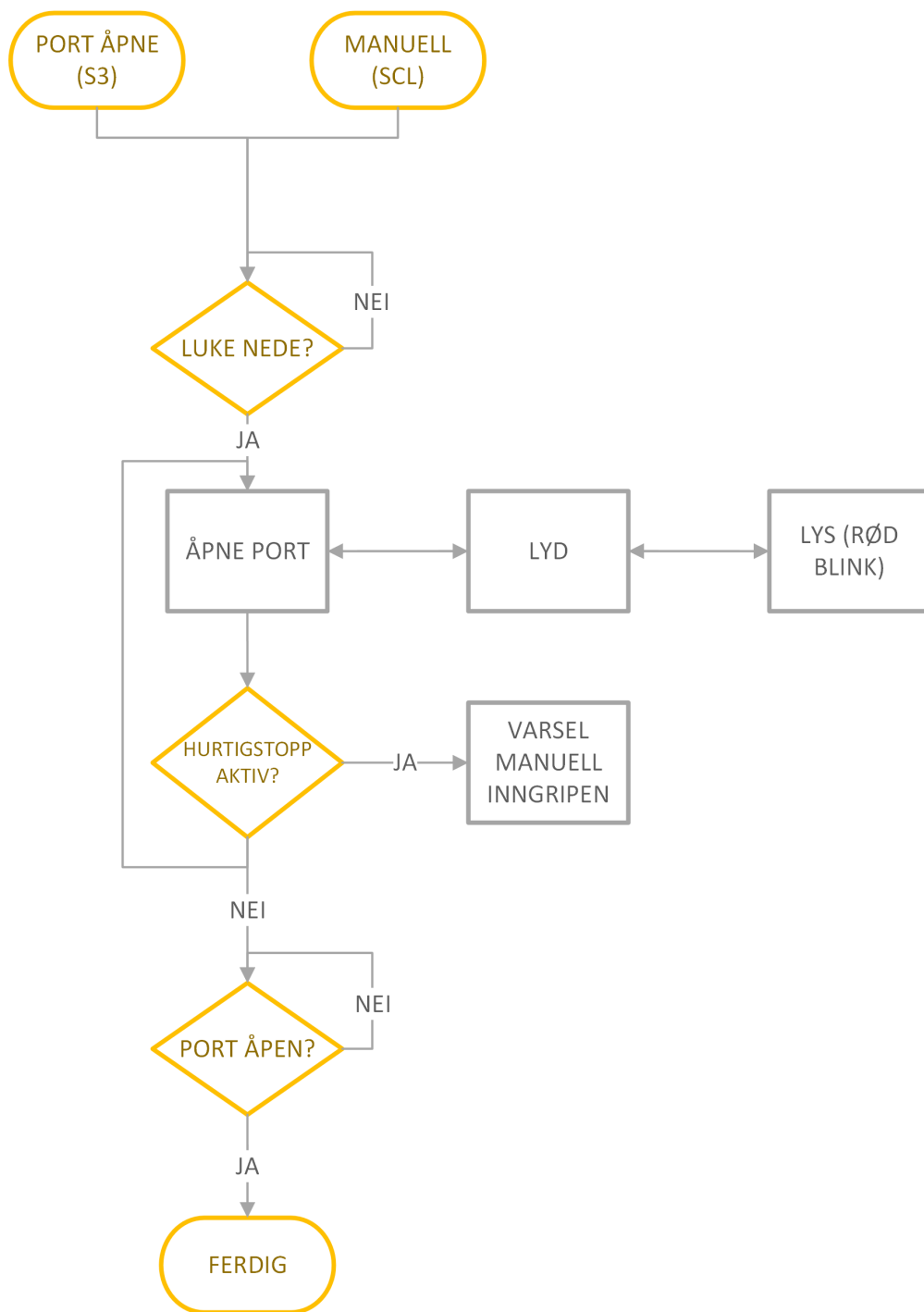
PORT - LEGGE TIL



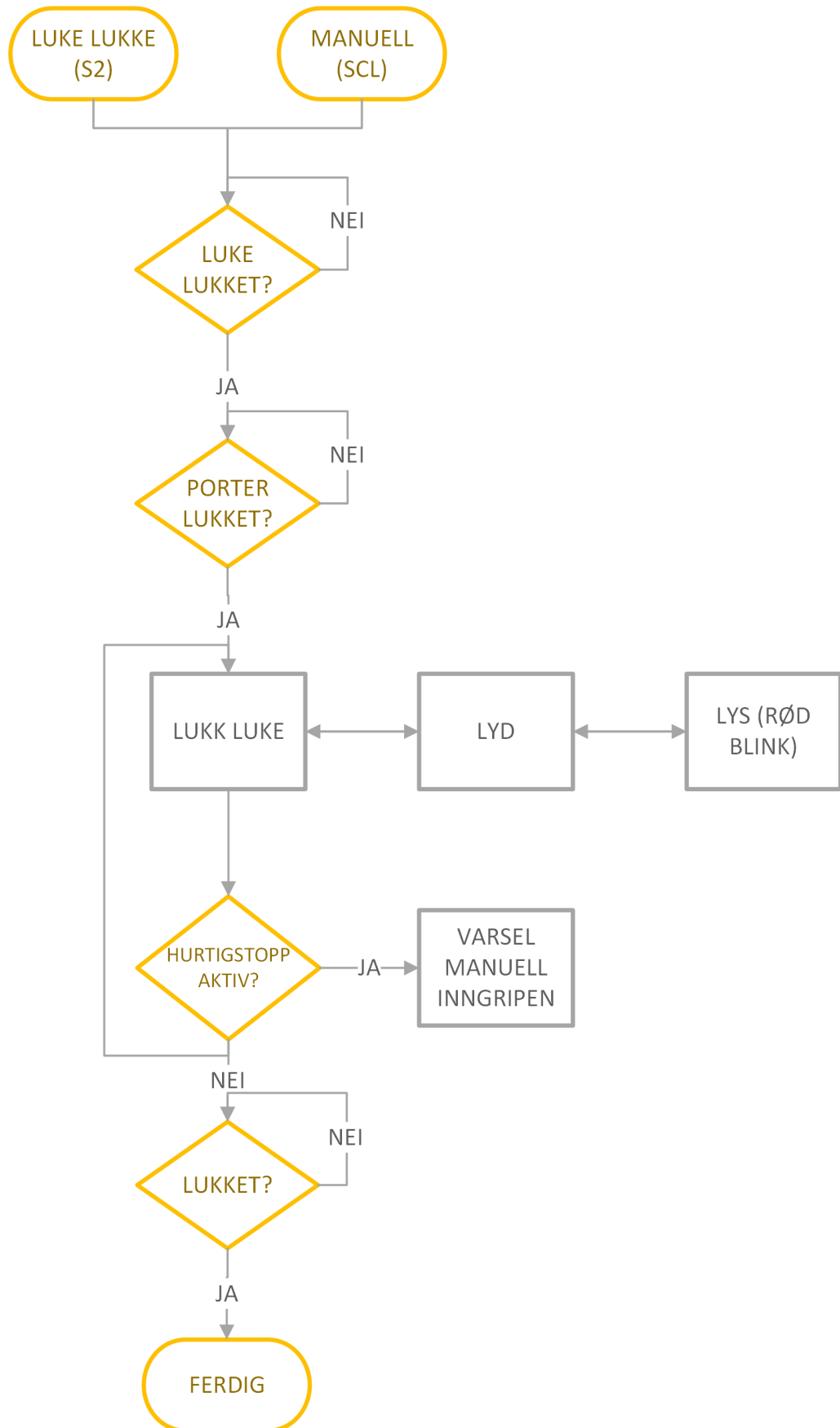
C - Sekvensielle skjemaer, manuell styring



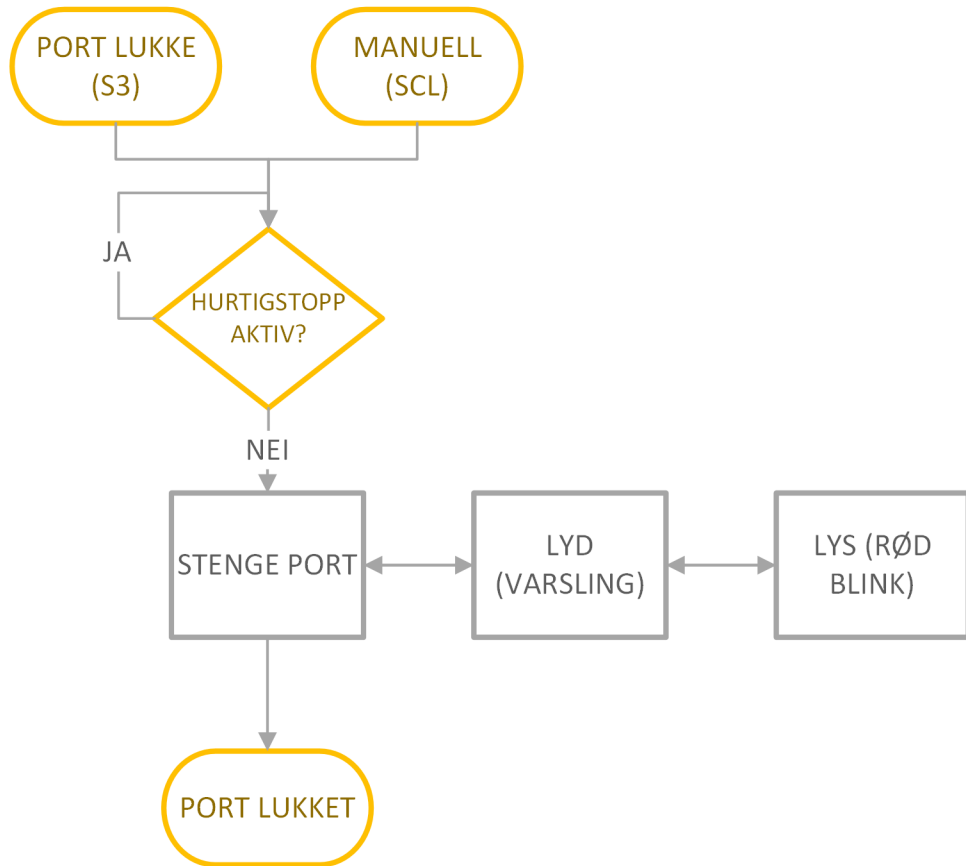
PORT ÅPNE



FERGE LUKE - LUKKE



PORT LUKKE



D - AFT-API grensesnitt og kommandoer

Milliampere2 bestillingsterminal - fergemodem API

Dette dokumentet er en omfattende oversikt over API-et som brukes i kommunikasjonen mellom betalingsterminalene, personhåndteringssystemet (PHS) og fergemodem. Det gir en detaljert beskrivelse av alle mulige kommandoer som brukes i systemet, samt de ulike kombinasjonene av bits som utgjør et gyldig signal.

Et gyldig signal består av tre hovedkomponenter: en mottaker-ID (REID - Receiver ID), en avsender-ID (SEID - Sender ID) og en kommando/handling. Sammen danner disse komponentene et 8-bit signal. Mottaker-ID er gitt av bit nr. 7:6, mens avsender-ID er gitt av bit nr. 5:4. De resterende 4 bitene, gitt som bit nr. 3:0, utgjør selve kommandoen.

Konfigurering av gyldig signal:

For å sette sammen et gyldig signal som skal brukes, velges først en mottaker av meldingen. Dersom fergen er mottaker av signalet må de 2 mest signifikante bitene (MSB) defineres med 0x3, 0b11 → REID = 0b11. Videre må avsender defineres. Dette gjøres ved å definere SEID til 0x1 eller 0x2 → SEID = 0b10. Til slutt velges en gyldig kommando gitt av bit 3:0 i en av kommando-identifikatorene. Eks: 0x3 i kommandoen STATE[3:0] i [GSTATE](#), som betyr *Gate close*. Sammensetningen av bits blir da 0b1110011 → 0x73.

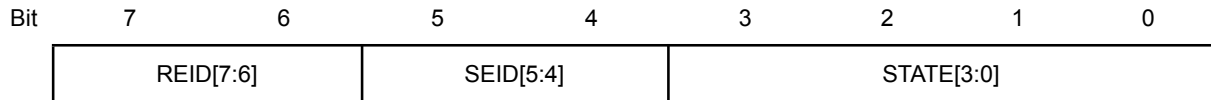
Sammendrag av kommando-identifikatorer

Navn	Bit Pos.	7	6	5	4	3	2	1	0
GSTATE	7:0	REID[7:6]		SEID[5:4]		STATE[3:0]			
DRDY	7:0	REID[7:6]		SEID[5:4]		STATE[3:0]			
GPCLEAR	7:0	REID[7:6]		SEID[5:4]		STATE[3:0]			
MSTP	7:0	REID[7:6]		SEID[5:4]		MANUALSTP[3:0]			
MCMD	7:0	REID[7:6]		SEID[5:4]		CMD[3:0]			
REQE	7:0	REID[7:6]		SEID[5:4]		REQ[3:0]			
REQFGS	7:0	REID[7:6]		SEID[5:4]		REQGS[3:0]			
GCMD	7:0	REID[7:6]		SEID[5:4]		CMD[3:0]			
CQUD	7:0	REID[7:6]		SEID[5:4]		CONFIRM[3:0]			
FREQS	7:0	REID[7:6]		SEID[5:4]		STATE[3:0]			
FPROC	7:0	REID[7:6]		SEID[5:4]		PROC[3:0]			
FGS	7:0	REID[7:6]		SEID[5:4]		STATES[3:0]			
PPLCOUNT	7:0	REID[7:6]		PPLCNTID[5:4]		COUNT[3:0]			

Gate States

Navn: GSTATE

Egenskap: Indikerer til mottaker ID tilstand på port hos avsender.



Bit 7:6 - REID[7:6] Receiver Identification

Dette bit-feltet definerer mulige mottakere av meldingen.

Verdi	Navn	Beskrivelse
0x3	FEID	Ferry Identification

Bit 5:4 - SEID[5:4] Sender Identification

Dette bit-feltet definerer mulige avsendere av meldingen.

Verdi	Navn	Beskrivelse
0x1	TEAID	Terminal A identification
0x2	TEBID	Terminal B identification

Bit 3:0 - STATE[3:0] Tilstand på port

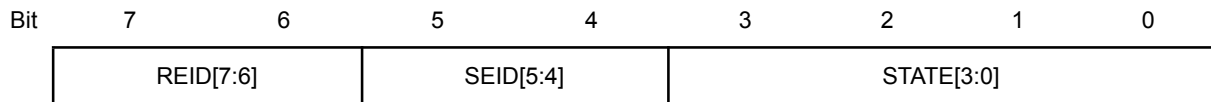
Dette bit-feltet definerer tilstanden på port ved avsender-ID.

Verdi	Navn	Beskrivelse
0x1	TIMEOUT	Gate timeout
0x2	OPEN	Gate open
0x3	CLOSE	Gate closed
0x4	PAUSE	Gate pause

Dock Ready

Navn: DRDY

Egenskap: Indikerer til mottaker-ID at brygge ved avsender-ID er fri for personer og objekter.



Bit 7:6 - REID[7:4] Receiver Identification

Dette bit-feltet definerer mulige mottakere av meldingen.

Verdi	Navn	Beskrivelse
0x3	FEID	Ferry Identification

Bit 5:4 - SEID[7:4] Sender Identification.

Dette bit-feltet definerer mulige avsendere av meldingen.

Verdi	Navn	Beskrivelse
0x1	TEAID	Terminal A identification
0x2	TEBID	Terminal B identification

Bit 3:0 - STATE[3:0] State of dock.

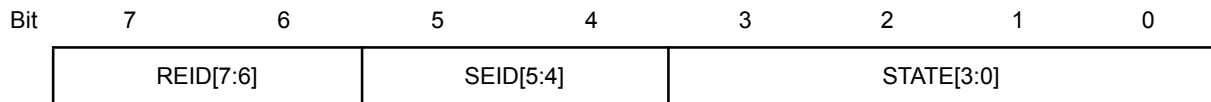
Dette bit-feltet indikerer at bryggen er fri for objekter.

Verdi	Navn	Beskrivelse
0x5	READY	Dock Ready
0x6	NREADY	Dock Not Ready

Gap clear

Navn: GPCLEAR

Egenskap: Indikerer til mottaker ID at gangbru mellom brygge og ferge er klar



Bit 7:6 - REID[7:4] Receiver Identification

Dette bit-feltet definerer mulige mottakere av meldingen.

Verdi	Navn	Beskrivelse
0x3	FEID	Ferry Identification

Bit 5:4 - SEID[7:4] Sender Identification

Dette bit-feltet definerer mulige avsendere av meldingen.

Verdi	Navn	Beskrivelse
0x1	TEAID	Terminal A identification
0x2	TEBID	Terminal B identification

Bit 3:0 - STATE[3:0] State of gap.

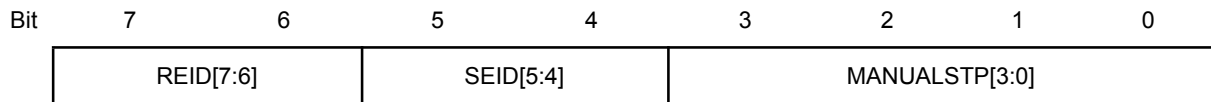
Dette bit-feltet indikerer at gangbru tilstand på gangbru

Verdi	Navn	Beskrivelse
0x7	READY	Gap Clear
0x8	NREADY	Gap Not Clear

Manual Stop

Navn: MSTP

Egenskap: Indikerer til mottaker ID at manuell stoppknapp er trykket.



Bit 7:6 - REID[7:4] Receiver Identification

Dette bit-feltet definerer mulige mottakere av meldingen.

Verdi	Navn	Beskrivelse
0x3	FEID	Ferry Identification

Bit 5:4 - SEID[7:4] Sender Identification

Dette bit-feltet definerer mulige avsendere av meldingen.

Verdi	Navn	Beskrivelse
0x1	TEAID	Terminal A identification
0x2	TEBID	Terminal B identification

Bit 3:0 - MANUALSTP[3:0] Manual Stop

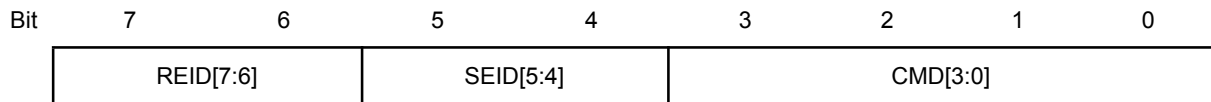
Dette bit-feltet indikerer om den manuelle stopp-knappen er trykket

Verdi	Navn	Beskrivelse
0x9	PUSHED	Button pushed

Manual Commands

Navn: MCMD

Egenskap: Manuelle kommandoer for styring av port og luke.



Bit 7:6 - REID[7:4] Receiver Identification

Dette bit-feltet definerer mulige mottakere av meldingen.

Verdi	Navn	Beskrivelse
0x3	FEID	Ferry Identification

Bit 5:4 - SEID[7:4] Sender Identification

Dette bit-feltet definerer mulige avsendere av meldingen.

Verdi	Navn	Beskrivelse
0x1	TEAID	Terminal A identification
0x2	TEBID	Terminal B identification

Bit 3:0 - CMD[3:0] Kommando

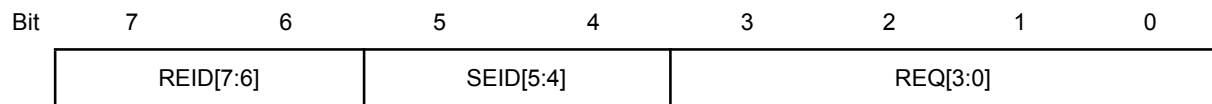
Dette bit-feltet definerer de ulike manuelle kommandoene for overstyring av port og luke

Verdi	Navn	Beskrivelse
0xA	MOG	Manual Open Gate
0xB	MCG	Manual Close Gate
0xC	MOH	Manual Open Hatch
0xD	MCH	Manual Close Hatch

Request Ferry

Navn: REQF

Egenskap: Indikerer at avsender ønsker å bestille ferge.



Bit 7:6 - REID[7:4] Receiver Identification

Dette bit-feltet definerer mulige mottakere av meldingen.

Verdi	Navn	Beskrivelse
0x3	FEID	Ferry Identification

Bit 5:4 - SEID[7:4] Sender Identification

Dette bit-feltet definerer mulige avsendere av meldingen.

Verdi	Navn	Beskrivelse
0x1	TEAID	Terminal A identification
0x2	TEBID	Terminal B identification

Bit 3:0 - REQ[3:0] Request ferry

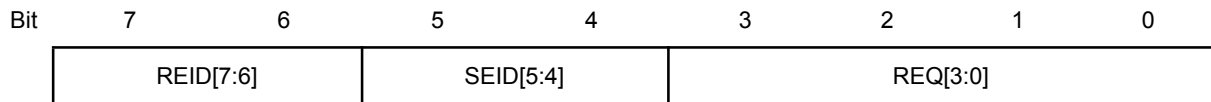
Dette bit-feltet definerer en bestilling av fergen.

Verdi	Navn	Beskrivelse
0xE	REQUEST	Request ferry

Request Ferry Gate States

Navn: REQF

Egenskap: Etterspør tilstand på portene plassert på fergen.



Bit 7:6 - REID[7:4] Receiver Identification

Dette bit-feltet definerer mulige mottakere av meldingen.

Verdi	Navn	Beskrivelse
0x3	FEID	Ferry Identification

Bit 5:4 - SEID[7:4] Sender Identification

Dette bit-feltet definerer mulige avsendere av meldingen.

Verdi	Navn	Beskrivelse
0x1	TEAID	Terminal A identification
0x2	TEBID	Terminal B identification

Bit 3:0 - REQ[3:0] Request ferry

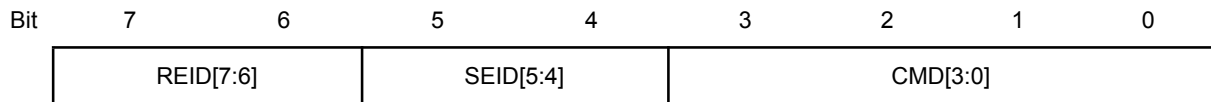
Dette bit-feltet definerer en bestilling av fergen.

Verdi	Navn	Beskrivelse
0xF	REQGS	Request Gate States

Gate Command

Navn: GCMD

Egenskap: Kontroll-kommandoer for overstyring av port fra ferge.



Bit 7:6 - REID[7:4] Receiver Identification

Dette bit-feltet definerer mulige mottakere av meldingen.

Verdi	Navn	Beskrivelse
0x1	TEAID	Terminal A identification
0x2	TEBID	Terminal B identification

Bit 5:4 - SEID[7:4] Sender Identification

Dette bit-feltet definerer mulige avsendere av meldingen.

Verdi	Navn	Beskrivelse
0x3	FEID	Ferry Identification

Bit 3:0 - CMD[3:0] Gate Command

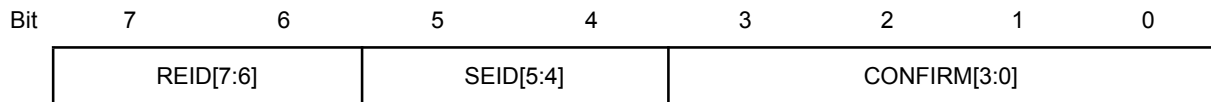
Dette bit-feltet definerer ulike kommandoene for styring av port.

Verdi	Navn	Beskrivelse
0x1	RESET	Gate Reset
0x2	OPEN	Gate Open
0x3	CLOSE	Gate Close
0x4	PAUSE	Gate Pause

Confirm Queue Update

Navn: CQUD

Egenskap: Bekrefter at bestilling er lagt til i bestillingskø.



Bit 7:6 - REID[7:4] Receiver Identification

Dette bit-feltet definerer mulige mottakere av meldingen.

Verdi	Navn	Beskrivelse
0x1	TEAID	Terminal A identification
0x2	TEBID	Terminal B identification

Bit 5:4 - SEID[7:4] Sender Identification

Dette bit-feltet definerer mulige avsendere av meldingen.

Verdi	Navn	Beskrivelse
0x3	FEID	Ferry Identification

Bit 3:0 - CONFIRM[3:0] Confirm

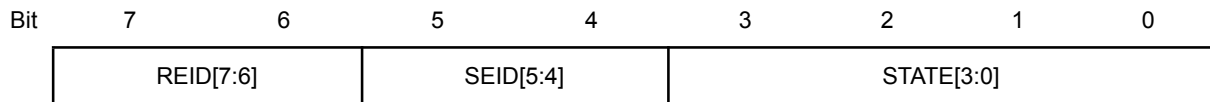
Dette bit-feltet definerer en bekreftelse på mottatt bestilling.

Verdi	Navn	Beskrivelse
0x5	QCONF	Queue confirm

Ferry Request States

Navn: FREQS

Egenskap: Forespørsel til mottaker-ID på status definert av bit 3:0.



Bit 7:6 - REID[7:4] Receiver Identification

Dette bit-feltet definerer mulige mottakere av meldingen.

Verdi	Navn	Beskrivelse
0x1	TEAID	Terminal A identification
0x2	TEBID	Terminal B identification

Bit 5:4 - SEID[7:4] Sender Identification

Dette bit-feltet definerer mulige avsendere av meldingen.

Verdi	Navn	Beskrivelse
0x3	FEID	Ferry Identification

Bit 3:0 - STATE[3:0] States

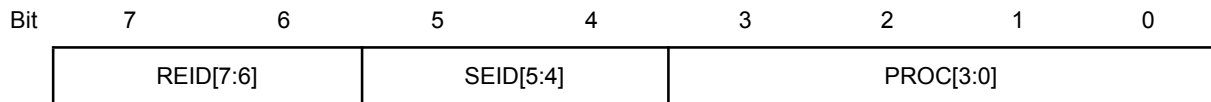
Dette bit-feltet definerer de ulike forespørsler etter tilstand på mottaker-ID.

Verdi	Navn	Beskrivelse
0x6	REQCNT	Request people counter
0x7	REQDRDY	Request Docking Ready
0x8	REQGCLR	Request Gap Clear
0x9	GSREQ	Gate State Request
0xA	MSREQ	Request State Manual Stop Button

Ferry Process

Navn: FPROC

Egenskap: Indikerer til mottaker-ID fullført prosess definert av bit 3:0.



Bit 7:6 - REID[7:4] Receiver Identification

Dette bit-feltet definerer mulige mottakere av meldingen.

Verdi	Navn	Beskrivelse
0x1	TEAID	Terminal A identification
0x2	TEBID	Terminal B identification

Bit 5:4 - SEID[7:4] Sender Identification

Dette bit-feltet definerer mulige avsendere av meldingen.

Verdi	Navn	Beskrivelse
0x3	FEID	Ferry Identification

Bit 3:0 - PROC[3:0] Process

Dette bit-feltet indikerer en gjennomført prosess

Verdi	Navn	Beskrivelse
0xB	DCOMP	Docking Complete
0xC	BCOMP	Boarding Complete

Ferry Gate States

Navn: FGS

Egenskap: Indikerer til mottaker-ID tilstand på portene plassert på fergen.

Bit 7 6 5 4 3 2 1 0

REID[7:6]	SEID[5:4]	STATE[3:0]
-----------	-----------	------------

Bit 7:6 - REID[7:4] Receiver Identification

Dette bit-feltet definerer mulige mottakere av meldingen.

Verdi	Navn	Beskrivelse
0x1	TEAID	Terminal A identification
0x2	TEBID	Terminal B identification

Bit 5:4 - SEID[7:4] Sender Identification

Dette bit-feltet definerer mulige avsendere av meldingen.

Verdi	Navn	Beskrivelse
0x3	FEID	Ferry Identification

Bit 3:0 - STATE[3:0] Gate States

Dette bit-feltet indikerer tilstand på portene på fergen.

Verdi	Navn	Beskrivelse
0xD	TIMEOUT	Gate Timeout
0xE	OPEN	Gate Open
0xF	CLOSED	Gate Closed
0x0	PAUSE	Gate Paused

People Count

Navn: PPLCNT

Egenskap: Indikerer til mottaker ID antallet passasjerer om bord i fergen.

Bit	7	6	5	4	3	2	1	0
	REID[7:6]		PPLCNTID[5:4]		COUNT[3:0]			

Bit 7:6 - REID[7:4] Receiver Identification

Dette bit-feltet definerer mulige mottakere av meldingen.

Verdi	Navn	Beskrivelse
0x0	All	All nodes in PAN

Bit 5:4 - PPLCNTID[7:4] People Count ID

Dette bit-feltet definerer mulige avsendere av meldingen.

Verdi	Navn	Beskrivelse
0x0	None	None

Bit 3:0 - COUNT[3:0] Count

Dette bit-feltet indikerer antallet passasjerer på fergen.

Verdi	Navn	Beskrivelse
0x0	0	Zero passengers
0x1	1	One passenger
0x2	2	Two passengers
0x3	3	Three passengers
0x4	4	Four passengers
0x5	5	Five passengers
0x6	6	Six passengers
0x7	7	Seven passengers
0x8	8	Eight passengers
0x9	9	Nine passengers
0xA	10	Ten passengers
0xB	11	Eleven passengers

0xC	12	Twelve passengers
0xD	13	Thirteen passengers
0xE	14	Fourteen passengers
0xF	15	Fifteen passengers

E - NMEA0183 kommando- og statusmeldinger lukestyring

Wago hatch control

Thursday, May 18, 2023 6:00 PM

Feedback: \$WAGOHATCHFB,<EM_STATE>,<HATCH_ID>,<state>,<action>,<fault>,<HATCH_ID>,<state>,<action>,<fault>*ch

Feedback	Description
Em state	Emergency stop (1=active, 0=inactive)
Fore hatch 1. State 2. Fault 3. Action	<p>States (states are read and sent in seperate program):</p> <ul style="list-style-type: none"> 1. 0: Undefined 2. 1: Open 3. 2: Closed 4. 3: 50 % <p>Faults:</p> <ul style="list-style-type: none"> 1. 0: ok 2. 1: Hatch did not open in time 3. 2: Hatch did not close in time 4. 3: Hatch did not reach 50 % from open state in time 5. 4: Hatch did not reach 50 % from closed state in time <p>Actions</p> <ul style="list-style-type: none"> 1. 0: Idle 2. 1: Opening 3. 2: Closing 4. 3: 50 % closing 5. 4: 50 % opening
Aft hatch 1. State 2. Fault 3. Action	
ID	
Fore hatch	1
Aft hatch	2

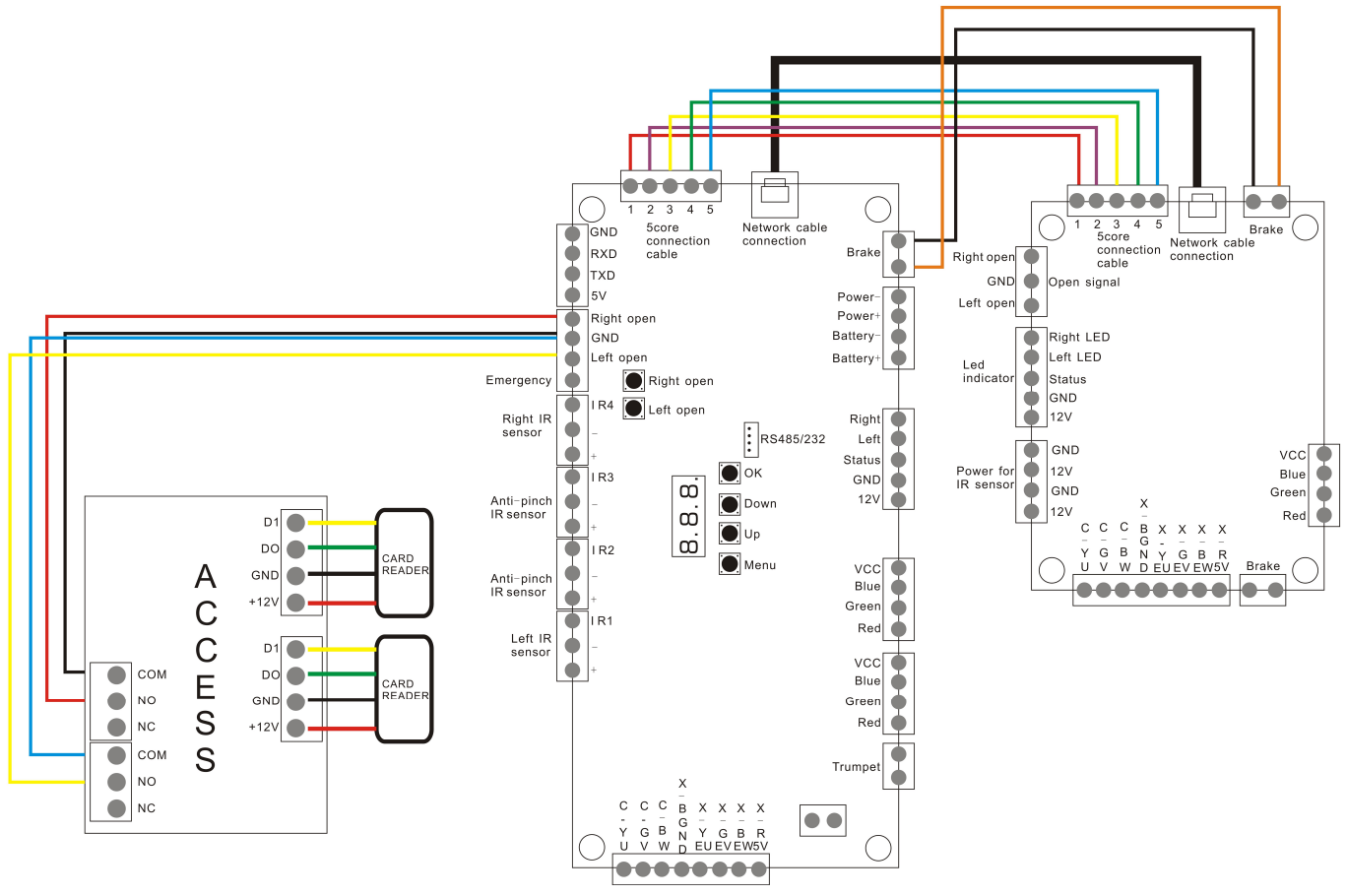
Command: \$WAGOHATCH,<ID>,<command>*ch

Command	Description	In e!COCKPIT
0: stop	Stop everything and go back to initial step	<p>Commands are received in the function block command_receiver and are being stored as bytes on GVL_hatch_control.fore_command and GVL_hatch_control.aft_command.</p> <p>The command is processed in the function block hatch_control_read_command.</p>
1: open		
2: 50%	Go to 50 %	
3: close		
4: reset	Go back to initial step if fault is not 0.	

F - Landport koblingskjema

Wiring Diagram for Brushless Mainboard 1.1

- Note: 1. Pls add extra power supply if need to fix any other device.
 2. Connection cable: 1pc 7*0.3mm power supply cable, and 1pc category 5, 8 core network cable.
 Note: Brake(1pc 1*0.3mm) is for speed gate only.



G - Landport seriellprotokoll

RS232/RS485 Protocol Baud rate: 19200 n 8

1. Routine switching instruction (this instruction feedback is to reply immediately after sending, which means normal communication)

Note: Send instruction CRC bit can not be calculated, send a fixed byte

95 FC

Broadcast readout machine number:

7E 80 00 00 00 00 01 AA 00 01 A3 7A 7E

Feedback: 7E 80 00 01 00 00 01 00 00 01 00 01 00 38 7E

(Red is the designated machine number)

Read the designated machine number (machine 1):

7E 80 00 01 00 00 01 AA 00 01 1B 1B 7E

Feedback: 7E 80 00 01 00 00 01 00 00 01 00 01 00 38 7E(machine 1)

Left gate open(machine 1): 7E 80 00 01 00 00 80 AA 00 01 01 00 DE 62 7E

Feedback: 7E 80 00 01 00 00 80 00 00 01 CA CA 7E

Left gate open (normally open) (machine 1):

7E 80 00 01 00 00 80 AA 00 01 01 01 CE 43 7E

Feedback: 7E 80 00 01 00 00 80 00 00 01 CA CA 7E

Left gate close(machine 1):

7E 80 00 01 00 00 81 AA 00 01 01 8A AB 7E

Feedback: 7E 80 00 01 00 00 81 00 00 01 BC 7E 7E

Right gate open(machine 1):

7E 80 00 01 00 00 80 AA 00 01 02 00 8B 31 7E

Feedback: 7E 80 00 01 00 00 80 00 00 01 CA CA 7E

Right gate close(machine 1):

7E 80 00 01 00 00 81 AA 00 01 02 C8 7E 7E

Feedback: 7E 80 00 01 00 00 81 00 00 01 BC 7E 7E

Right gate open (normally open) (machine 1):

7E 80 00 01 00 00 80 AA 00 01 02 01 9B 10 7E

Feedback: 7E 80 00 01 00 00 80 00 00 01 CA CA 7E

Turn on the right infrared sensor to trigger gate opening (machine 1)

(This instruction is not to save after power failure. If you need to save after power

failure, please use the Set L-2 menu command)
 7E 80 00 01 00 00 80 AA 00 01 01 03 95 fc 7E
 Feedback: 7E 80 00 01 00 00 80 00 00 01 CA CA 7E

Close the right infrared sensor to trigger gate opening (machine 1)
 (This instruction is not to save after power failure. If you need to save after power failure, please use the Set L-2 menu command)
 7E 80 00 01 00 00 80 AA 00 01 01 04 95 fc 7E
 Feedback: 7E 80 00 01 00 00 80 00 00 01 CA CA 7E

Turn on the left infrared sensor to trigger gate opening (machine 1)
 (This instruction is not to save after power failure. If you need to save after power failure, please use the Set L-2 menu command)
 7E 80 00 01 00 00 80 AA 00 01 02 03 95 fc 7E
 Feedback: 7E 80 00 01 00 00 80 00 00 01 CA CA 7E

Close the left infrared sensor to trigger gate opening (machine 1)
 (This instruction is not to save after power failure. If you need to save after power failure, please use the Set L-2 menu command)
 7E 80 00 01 00 00 80 AA 00 01 02 04 95 fc 7E
 Feedback: 7E 80 00 01 00 00 80 00 00 01 CA CA 7E

2. General result feedback: (This instruction is automatically sent back by the motherboard under specific trigger conditions, crc can not do the check) (the following red bottom byte is the key identification bit) (if you meet the reply of 2 7E beginning, please ignore the first 7E)

7E 80 00 01 00 00 61 00 00 01 94 38 7E	Pass left
7E 80 00 01 00 00 62 00 00 01 48 A3 7E	Pass right
7E 80 00 01 00 00 63 00 00 01 FC D5 7E	Left pass timeout
7E 80 00 01 00 00 64 00 00 01 84 D1 7E	Right pass timeout
7E 80 00 01 00 00 73 00 00 01 crc crc 7E	Left pass timeout when people in the passage
7E 80 00 01 00 00 74 00 00 01 crc crc 7E	Right pass timeout when people in the passage
7E 80 00 01 00 00 83 00 00 01 crc crc 7E	Reverse alarm after the left pass is opened
7E 80 00 01 00 00 84 00 00 01 crc crc 7E	Reverse alarm after the right pass is opened
7E 80 00 01 00 00 93 00 00 01 crc crc 7E	Tail alarm after left pass is opened
7E 80 00 01 00 00 94 00 00 01 crc crc 7E	Tail alarm after right pass is opened
7E 80 00 01 00 00 95 00 00 01 crc crc 7E	Upload in standby state when blocking

infrared

3. State query instruction: (can query door state, query infrared state)(menu setting command interface)

Enable motor	01 06 00 04 00 00 C8 0B(send)	01 06 00 04 00 00 AA 55			
Disabled motor	01 06 00 04 00 01 09 CB(send)	01 06 00 04 00 01 AA 55			
The brakes on	01 06 00 05 00 01 58 0B(send)	01 06 00 05 00 01 AA 55			
The brakes off	01 06 00 05 00 00 99 CB(send)	01 06 00 05 00 00 AA 55	When the upper computer receives the returned value, it means that the command is sent successfully		
Query Door Opening Status	01 03 00 12 00 01 24 0F(send)	01 03 02 00 00 AA 55	Indicates the door is closed in place		
		01 03 02 00 01 AA 55	Indicates open door A is in place		
		01 03 02 00 02 AA 55	Indicates open door B is in place		
		01 03 02 00 03 AA 55	Indicates the opening process of A		
		01 03 02 00 04 AA 55	Indicates the opening process of B		
		01 03 02 00 05 AA 55	Indicates the closing process of A		
		01 03 02 00 06 AA 55	Indicates the closing process of B		
		01 03 02 00 07 AA 55	Indicates initializing		
		01 03 02 00 08 AA 55	Means someone is pushing at the door		
		01 03 02 00 09 AA 55	Indicates that the motor is looking for zero		
Query	01 03 00 21 00 01 D4	01 03 02 00 00 -- --	No infrared trigger		
		01 03 02 00 01 -- --	Through the first point		

infrared status	00(send)	01 03 02 00 02 -- --	Through the second point		
		01 03 02 00 03 -- --	Through the third point		
		01 03 02 00 04 -- --	It means someone is retrograde		
		01 03 02 00 05 -- --	Indicates that someone is following		
		01 03 02 00 06 -- --	Indicates infrared timeout closing		
		01 03 02 00 07 -- --	Indicates an intruder.		
		01 03 02 00 08 -- --	Indicates that the person passes through the channel normally and is in standby state		
	Sending value	The return value		Default value	The scope of parameters
Order number	Example: 55 01 01 X1 X2 X3 X4 X5 X1 X2 data bit (X1 data is 8 bits higher, X2 data is 8 bits lower), X3 status bit (write 0 for setting parameter when sending (automatically save after writing), write 1 for reading parameter) (for feedback) check X4 X5 end bit, The value is fixed as AA. 55 Data is automatically saved if the Settings are different				
L-1	55 01 01 00 06 00 AA 55 Set L-1 to 06 seconds 55 01 01 00 00 01 AA 55 Read and open L-1 parameter duration		55 01(menu type) 01(menu number) 00(high data level) 06(Low data level) 00(Status 00 indicates that the operation succeeds, 02 indicates that the operation fails, and 01 indicates that the read command is sent) AA 55	6	0-99
L-2	55 01 02 00 01 00 AA 55 Set left swipe card right infrared 55 01 02 00 01 01 AA 55 Read left swipe card right infrared	Control mode	Gate working mode :0 left swipe card right swipe card 1: left infrared right swipe card 2: right infrared left swipe card 3: left infrared 4: left forbidden right pass 5: right forbidden left pass 6: two-way ban	0	0-6

L-3	55 01 03 00 01 00 AA 55 Sets the delay closing duration is 1s. The unit is seconds. The unit is s 55 01 03 00 00 01 AA 55 Read delay closing duration	Exit delay time	Delayed closing duration (unit second)	0.2	0.1--10
L-4	55 01 04 XX 00 00 AA 55		Left pass voice	1	0-18
L-5	55 01 05 XX 00 00 AA 55		Right pass speech	15	0-18
L-6	55 01 06 XX 00 00 AA 55		Infrared overlay time (in seconds)	3	0.1-30
L-7	55 01 07 XX 00 00 AA 55		Direction of power off	0	0: left open 1: right open
L-8	55 01 08 XX 00 00 AA 55		Alarm voice	0	0 stands for: illegal breakin please swipe card 14 stands for: beep sound
L-9	55 01 09 XX 00 00 AA 55		Configuration of the entrance and exit memory function	0	0 indicates that the memory function is disabled. 1 indicates that the memory function is enabled
L-10	55 01 0A XX 00 00 AA 55		Infrared anti-clip action	0	0: Stop 1: Bounce open the door
L-11	55 01 0B XX 00 00 AA 55		Infrared number	0	0 indicates 4. 1 indicates 6
L-12	55 01 0C XX 00 00 AA 55		Voice volume	7	0 minimum 15 maximum
L-13	55 01 0D XX 00 00 AA 55		Voice test		Press OK to exit
L-14	55 01 0E XX 00 00 AA 55		Aging test		Press OK to enter, press in add and subtract for

					test
L-15	55 01 0F XX 00 00 AA 55		Brand selection of main motor: E-1 is changed to 1 in the main board	0	
L-16	55 01 10 XX 00 00 AA 55		Brand selection of slave motor: E-2 is changed to 1 in the main board	0	
L-17	55 01 11 XX 00 00 AA 55		Reverse gate close selection	0	0:Do not close 1: close
L-18	55 01 12 XX 00 00 AA 55		Two-color light three-color light control options	0	0: two-color indicator. 1: three-color indicator
L-19	55 01 13 XX 00 00 AA 55				
Querying the Power Supply Voltage	55 01 FF 00 00 00 AA 55		Feedback 55 01 FF 5D00 02 AA 55 5D00 Represents voltage Voltage unit MV Query the supply voltage		
Reset instructions	55 01 FE 00 00 00 AA 55				
Obtain infrared state	55 01 FD 00 00 01 AA 55		55 01 FD 00 xx 01 AA 55, the first pair with infrared signal is 01, the sixth pair with signal is 20, the lower 6 bits of xx represents the state of the 6 pairs of infrared port, the corresponding bit with signal is 1, and the corresponding bit without signal is 0		
D-1	55 02 01 XX 00 00 AA 55		If the door is not aligned into the motor loose shaft swing to the desired closing position		
D-2	55 02 02 XX 00 00 AA 55		After entering the left open and then the motor loose shaft, swing to the desired opening position		
D-3	55 02 03 XX 00 00 AA 55		After entering the right open and then the motor loose shaft swing to the desired opening position		
D-4	55 02 04 XX 00 00 AA 55		Opening and closing speed setting	3	1 is the fastest and 10 is the slowest
D-5	55 02 05 XX 00 00 AA		Gate mode	0	0 swing

	55				barrier and 1 flap barrier
D-6	55 02 06 XX 00 00 AA 55		Is there a clutch	0	0:No 1:Yes
D-7	55 02 07 XX 00 00 AA 55		Rebound pattern	0	0: bounce back and continue to drive after hitting the person 1: pause and continue to drive after hitting the person
D-8	55 02 08 XX 00 00 AA 55		Against the strength	7	1-20, The higher the value ,the greater the resistance. The smaller the value, the smaller the resistance
D-9	55 02 09 XX 00 00 AA 55		The teset time	0	0-40
D-10	55 02 0A XX 00 00 AA 55		Main door close buffering	30	The greater the value of 1-100, the greater the force in place
D-11	55 02 0B XX 00 00 AA 55		Main motor rotating speed	90	1-100, The higher the value ,the faster the motor rotate speed, and the smaller the value, the slower the rorate motor speed
D-12	55 02 0C XX 00 00 AA 55		Slave door close buffering	50	1-100,The greater the

					value ,the greater the force in place
D-13	55 02 0D XX 00 00 AA 55		Slave motor rotating speed	90	1-100,The higher the value, the faster the motor rotate speed, and the smaller the value, the slower the rorate motor speed
D-14	55 02 0E XX 00 00 AA 55		Power	6	1--10
D-15	55 02 0F XX 00 00 AA 55		Speed for find zero after power on	5	The higher the value, the faster the find zero, and the lower the value, the slower the find zero
D-16	55 02 10 XX 00 00 AA 55		The direction of zero	0	0/1 If the gate is out of position, adjust to 1
D-17	55 02 11 XX 00 00 AA 55		Physical anti-pinch sensitivity	75	1-99, The larger the value,the less sensitive the physical anti-pinch. The smaller the value, the more sensitive the anti-pinch
D-18	55 02 12 XX 00 00 AA 55		Physical anti-pinch current time	10	1-99,The larger the value, the less sensitive the physical

					anti-pinch. The smaller the value, the more sensitive the anti-pinch
D-19	55 02 13 XX XX 00 AA 55		Acceleration the current	210	The larger the value, the faster the motor starts, and the smaller the value, the slower the motor starts
D-20	55 02 14 XX 00 00 AA 55		Acceleration the current time	10	The larger the value, the faster the motor start time, the smaller the value, the slower the motor start time
D-21	55 02 15 XX 00 00 AA 55		Clutch pull Angle	15	The larger the value, the larger the pull-in angle, and the smaller the value, the smaller the pull in angle
D-22	55 02 16 XX 00 00 AA 55		Duty cycle of clutch adjustment	10	
D-23	55 02 17 XX 00 00 AA 55		Physical anti-pinch reaction time	0	The slower the speed, the higher the number
D-24	55 02 18 XX 00 00 AA 55		Stall sensitivity	2	The larger the value, the longer the stall. The

					smaller the value, the shorter the stall
D-25	55 02 19 XX XX 00 AA 55		Current protection duration	2	The larger the value, the longer the protection duration. The smaller the value, the shorter the protection duration
D-26	55 02 1A XX 00 00 AA 55		Slave machine antagonism strength	9	1-20 The higher the value of 1-20, the greater the resistance. The smaller the value, the smaller the resistance
D-27	55 02 1B XX 00 00 AA 55		Deceleration stroke	8	The higher the number, the earlier the deceleration

H - Koblingsskjema og rekkeklemmetabell for landterminal

Rekkeklemmeliste

230V	-X1	Lask	
Automatsikring (1)	1		Landport (L)
Automatsikring (2)	2		Landport (N)
Automatsikring (1)	3		Ganglys (L)
Automatsikring (2)	4		Ganglys (N)

24V	-X2	Lask	
Pir rele (K1, 1)	1		Lys sensor (1)
Pir rele (K1, 2)	2		Lys sensor (2)
USB I/O module (Z2, DRY)	3		Stoppknapp (+)(S0)
	3		Manuellknapp(+)(S1)
USB I/O module (Z2, DI 0)	4		Stoppknapp (-)
USB I/O module (Z2, DO 0)	5		Port rele (+)
USB I/O module (Z2, GND)	6		Port rele (-)
PC (Z1, GND)	7		SAM R34 (GND)
PC com (Z1, RX-)	8		SAM R34 (TX-)
PC com (Z1, TX-)	9		SAM R34 (RX-)
PC com (Z1, RX+)	10		SAM R34 (TX+)
PC com (Z1, TX+)	11		SAM R34 (RX+)
	12		
	13		
	14		
	15		
	16		
	17		
	18		
	19		
	20		
	21		
	22		
	23		

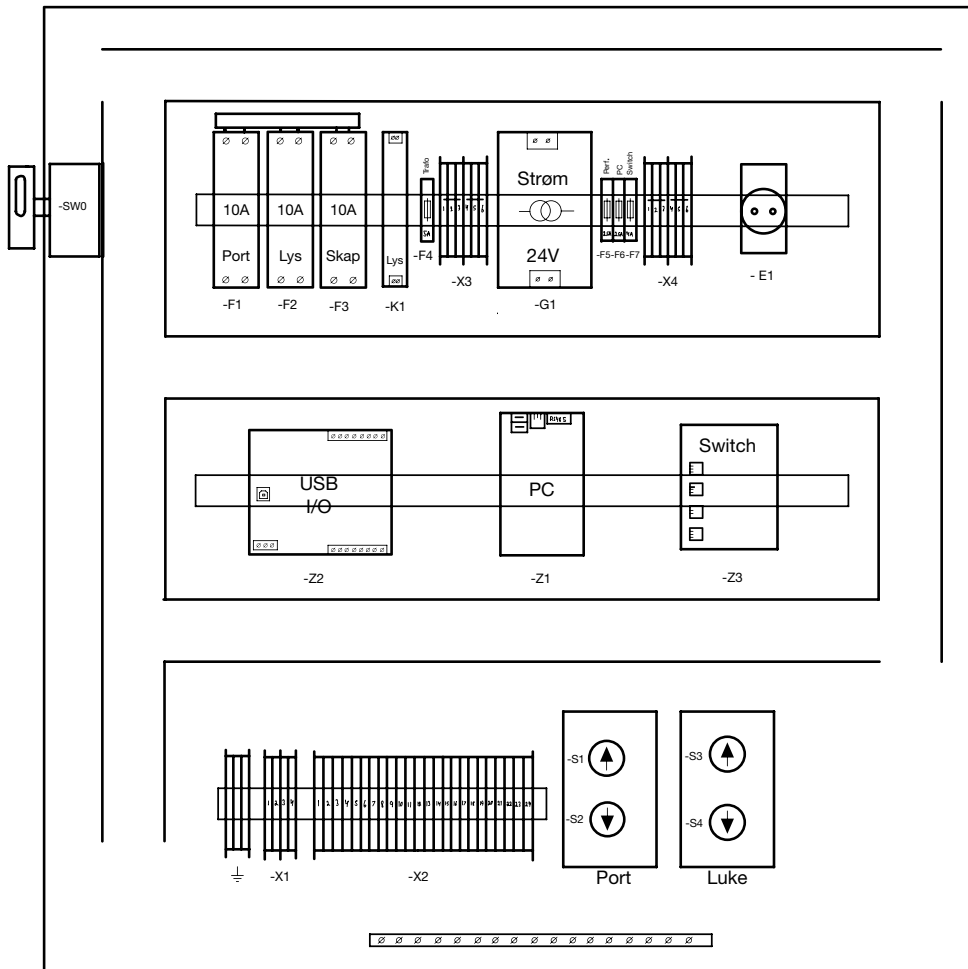
230V	-X3	Lask	
Automatsikring (F3,1)	1		Skumringsrele (L)
Glassikring Trafo (F4, 2)	2		-
Stikk på DIN (L)	3		-
Automatsikring (F3, 2)	4		Trafo (N)
Skumringsrele (N)	5		-
Stikk på DIN (N)	6		-

24V	-X4	Lask	
Glassikring (F5, 2)	1		USB I/O modul (+)
Rekkelemme (X2, 4)	2		USB I/O module (Z2, PC0)
Trykknapp (S1, 1)	3		-
Trafo (24v -)	4		USB I/O modul (-)
Pc (Z1, -)	5		Port rele GND
Switch (Z3, -)	6		-

Komponentliste

Utstyr	Referanse
Hovedbryter	SW0
Sikring C10A (Port)	F1
Sikring C10A (Lys)	F2
Sikring C10A (Skap)	F3
Sikring T5A (Trafo)	F4
Sikring T2.5A (Periferi)	F5
Sikring T2.5A (PC)	F6
Sikring T4A (Switch)	F7
Rele (Lys)	K1
Strømforsyning 24VDC	G1
PC	Z1
USB I/O	Z2
Switch	Z3
Trykknapp opp (Port)	S1
Trykknapp ned (Port)	S2
Trykknapp opp (Luke)	S3
Trykknapp ned (Luke)	S4
Stikkontakt skap	E1
Rekkeklemmer (230VAC)	X1
Rekkeklemmer (24VDC)	X2
Rekkeklemmer (230VAC Skap)	X3
Rekkeklemmer (24VDC Periferi)	X4

Arrangementstegning



Tilførsel
230VAC
25A

Port forsyning
230 VAC

ordgubg
sylinderqill

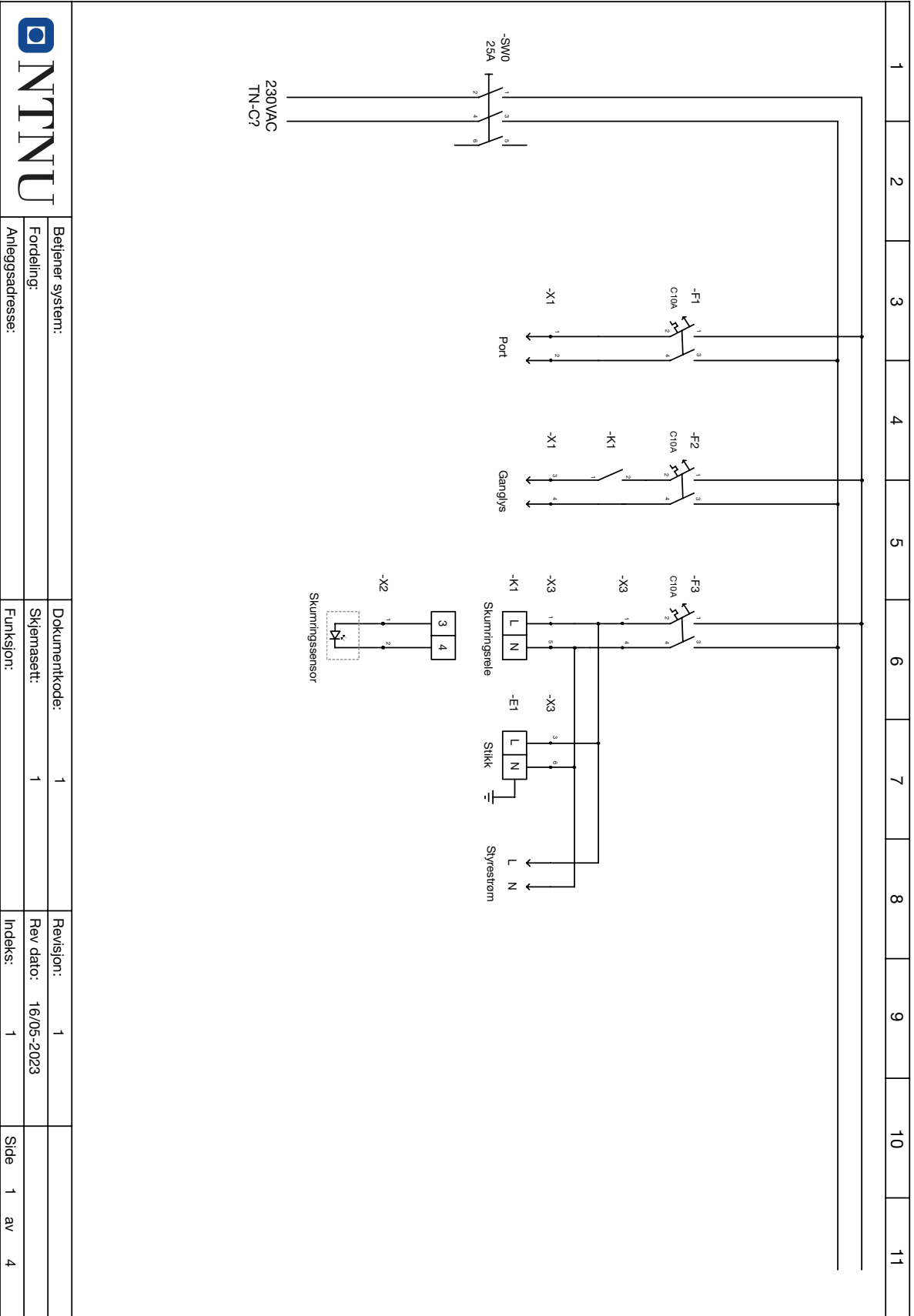
Skumringssensor

Hurtigstopp

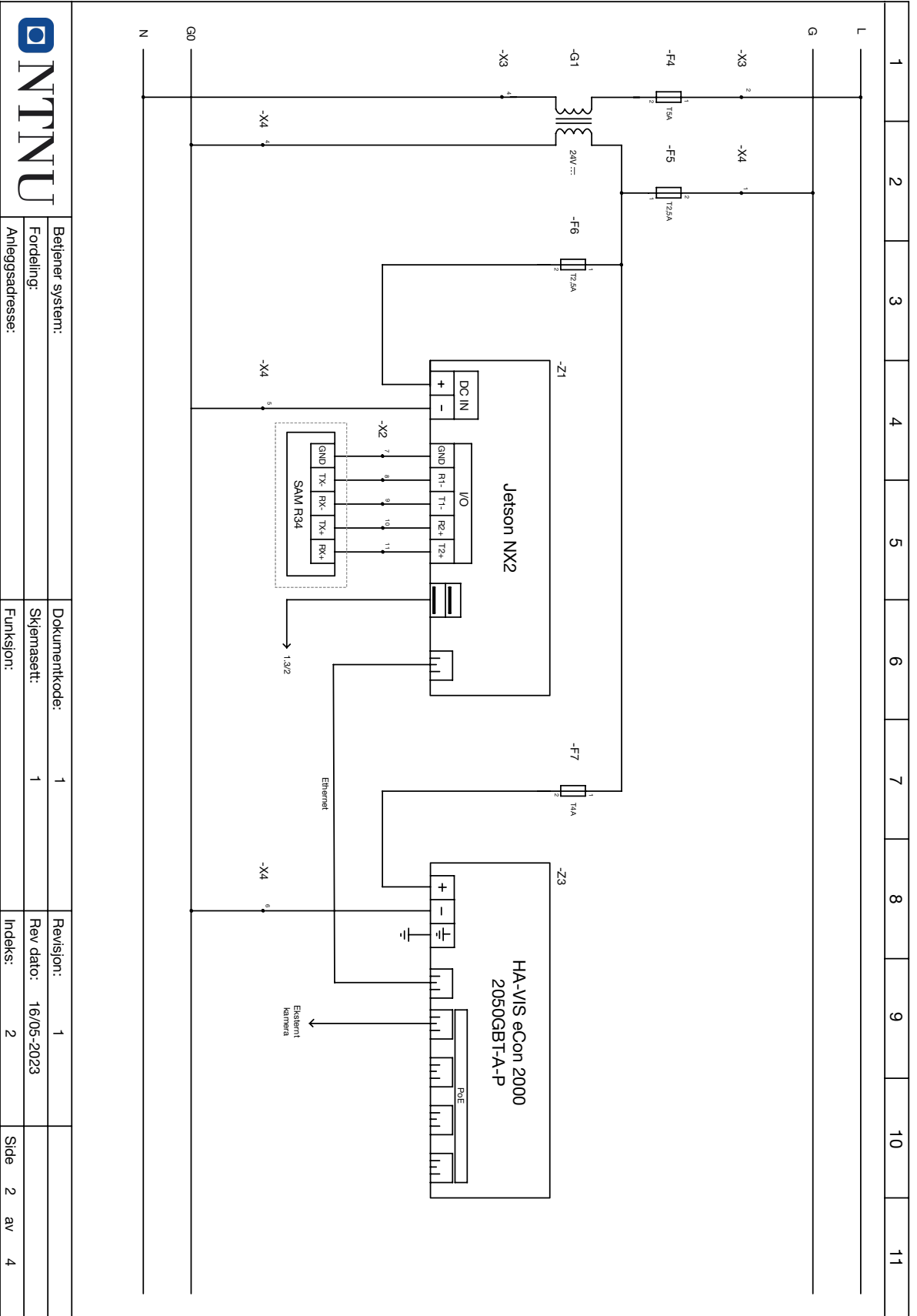
Port RS-485


Terminal
RS-422

Kamera

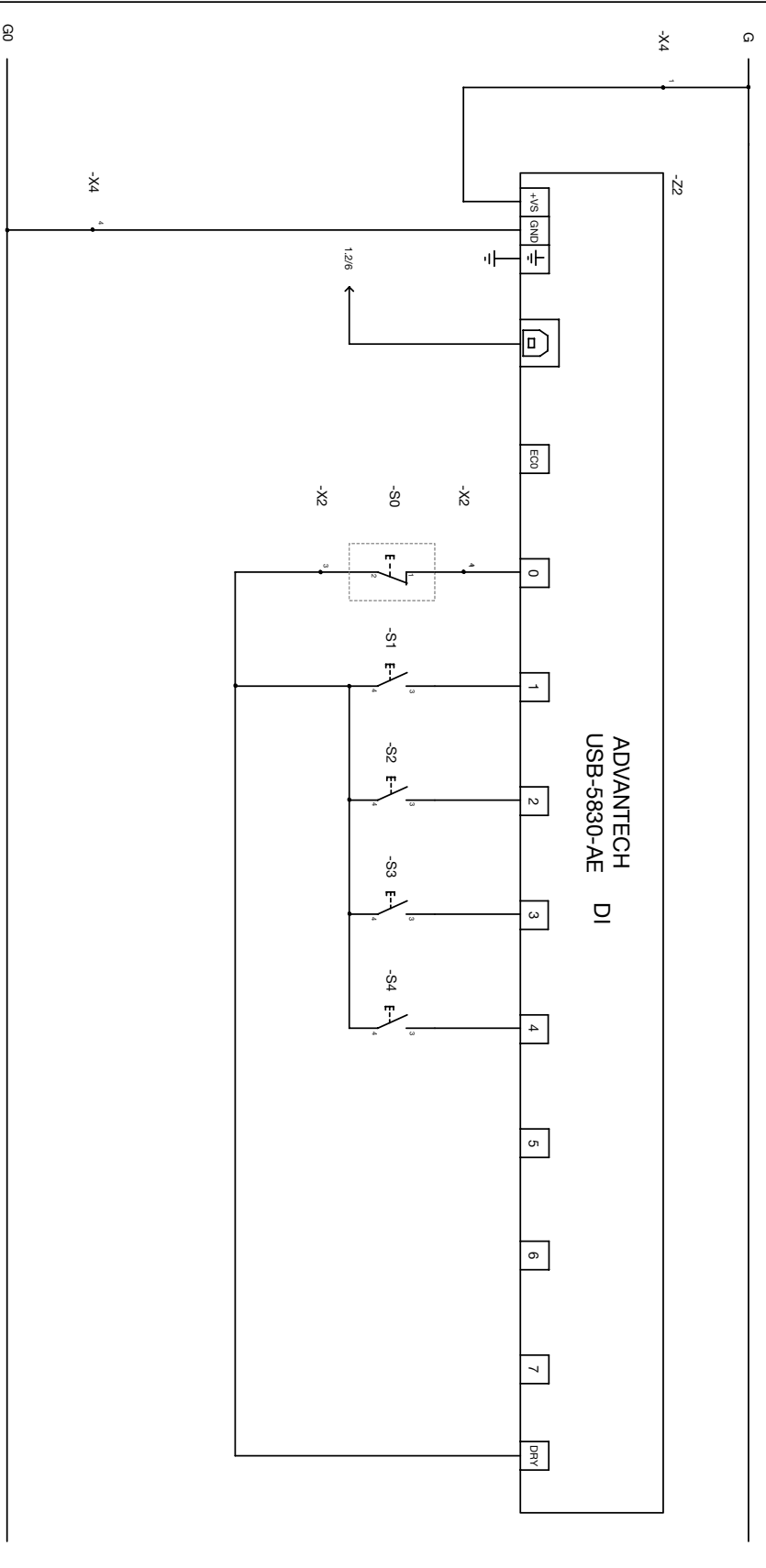



NTNU	Betjener system:	Dokumentkode: 1	Revisjon: 1
	Fordeling:	Skjemasett: 1	Rev dato: 16/05-2023
	Anleggsadresse:	Funksjon:	Indeks: 1
			Side 1 av 4



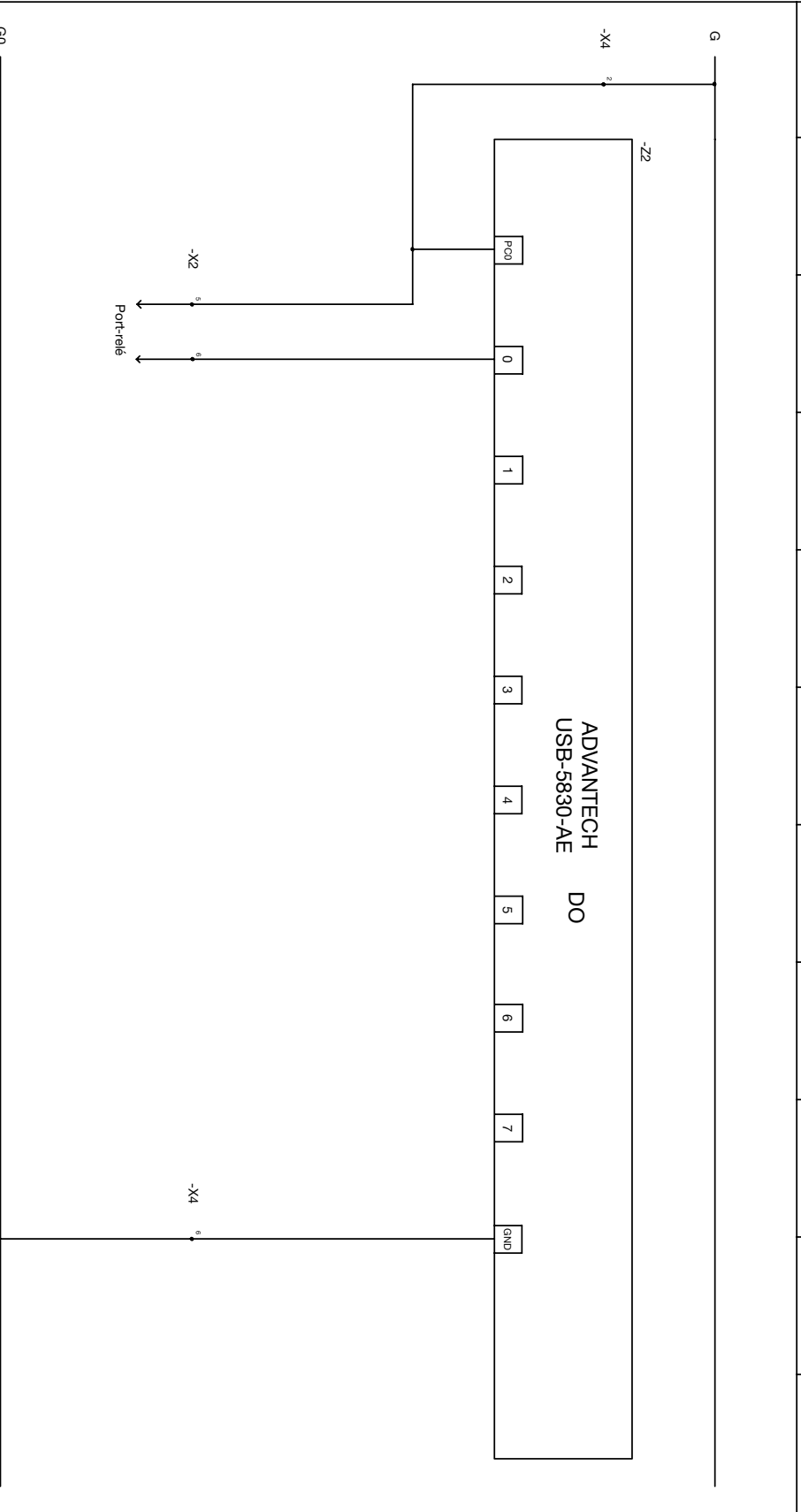
		Betjener system:		Dokumentkode: 1		Revisjon: 1			
Fordeling:		Anleggsadresse:		Skjemasett: 1		Rev dato: 16/05-2023			
				Funksjon:		Indeks: 2		Side 2 av 4	


1	2	3	4	5	6	7	8	9	10	11
---	---	---	---	---	---	---	---	---	----	----



	Beløper system:	Dokumentkode:	1	Revision:	1	
	Fordeling:	Skjemasett:	1	Rev dato:	16/05-2023	
	Anleggsadresse:	Funksjon:		Indeks:	3	Side 3 av 4

1	2	3	4	5	6	7	8	9	10	11
---	---	---	---	---	---	---	---	---	----	----



	Betjener system:	Dokumentkode:	1	Revision:	1	Side 4 av 4
	Fordeling:	Skjemasett:	1	Rev dato:	16/05-2023	
	Anleggsadresse:	Funksjon:		Indeks:	4	

I - Innkjøp, merket med hvor varen er bestilt

Arbeidsområde	Leverandør	Vare	Artikkelnr.	STK/meter
Fergeport	Solar	LED FLEX	3702336	1
Fergeport	Solar	ØLFLEX -CLASSIC-110 4 G 0,75	1090417	10
Fergeport	Solar	ØLFLEX-CLASSIC-110 7 G 0,75	1090419	10
Fergeport	Solar	ØLFLEX-CLASSIC-110 4 G 0,5	1090402	20
Fergeport	Solar	ALU.PROFIL	3200417	2
Fergeport	Solar	BELDEN 9842NH 2P RS-485	1097805	30
El-skap	Solar	REGNTAK FOR AE - 1010 - 1012	2430376	1
El-skap	Solar	STRIPSEFESTE MED LIM 28X28 SORT	7980883	1
El-skap	Solar	STRIPS SVART 200X4,6MM	1348759	1
El-skap	Solar	JORDINGSLISSE 170/4 5 STK	2430868	1
Fergeport	Lyskomponenter.no	Controller DMX512 CV RGBW 4Chx5A DIN	S25005	1
Sikkerhetssone	Solar	Stripline, ledstrip 24v (Lyktestolpe over inngang)	3201129	1
Sikkerhetssone	Solar	ØLFLEX Classic 110 2G1.5	1090460	20
El-skap	Solar	Stikk DIN	1513100	1
El-skap	Solar	Jord skinne	7920810	1
El-skap	Solar	Samle skinne	1638000	10
El-skap	Solar	Bryter panel	4303518	2
El-skap	Solar	Stopp knapp med boks	4303510	1
El-skap	Solar	Selvbårende skrue tykthode	1300706	1
El-skap	Solar	Selvbårende skrue tynthode	1300703	1
El-skap	Solar	Sikringsholder	1266417	4
El-skap	Solar	Glass sikring T5A	1600047	10
El-skap	Solar	Glass sikring T2.5A	1600044	10
El-skap	Solar	Glass sikring T4A	1600046	10
El-skap	Solar	Endestop rek.kl	1264904	10
El-skap	Solar	Endedeksel	7942118	4
El-skap	Solar	Patchekabel	6906634	1
El-skap	Solar	Signalkabel	1012885	10
El-skap	Solar	Mutter M16	1254760	100
El-skap	Solar	Mutter PG21	1248206	4
El-skap	Solar	Strømforsyning	7100364	1
El-skap	Solar	DIN skinne	7941239	1

Arbeidsområde	Leverandør	Vare	Artikkelnr.	STK/meter
El-skap	Solar	Rustfrit Skap	2430344	1
El-skap	Solar	Rekkeklemmer jording	1238747	3
El-skap	Solar	Rekkeklemmer	1238742	40
El-skap	Solar	Merking til rekkeklemmer	1218810	1
El-skap	Solar	Skilleplater til rekkeklemmer	1208343	100
El-skap	Solar	Lask til rekkeklemmer	1208329	25
El-skap	Solar	NIPPEL M16	1254710	8
El-skap	Solar	NIPPEL PG21	1248066	4
El-skap	Solar	Nitesett	2017315	1
El-skap	Solar	Endehyle 4mm2	2017297	1
El-skap	Solar	Endehyle 1.5mm2	2019404	1
El-skap	Solar	SORT 1,5 MM2	1022734	1
El-skap	Solar	SORT 4 MM2	1022753	1
El-skap	Solar	MØRKBLÅ 0,75MM2	1022717	1
El-skap	Solar	GUL/GRØNN 1,5MM2	1066603	1
El-skap	Solar	Sikring	1676065	3
El-skap	Solar	Nettverk switch	6906889	1
El-skap	Solar	Lys sensor, med rele	1475896	1
El-skap	Solar	Kabelkanal	1201156	1
El-skap	Solar	Kabelkanal lokk	1201149	1
El-skap	Solar	Hovedbrytter	1475420	1
El-skap	Solar	Krympestrømpe m/lim 12-3mm	1835763	1
Fergeport	LINAK	USB2LIN og BUSLINK-kabel	147999	1
Fergeport	Reichelt elektronikk	EX-DIN-RAIL	EXSYS EX-6099	1
Fergeport	Ahlsell	Innvendig Sekskantskrue DIN 912 / ISO 4762 A4 Syrefast	129896	1
Fergeport	Ahlsell	Låsemutter DIN 985 A4 syrefast	129105	1
Fergeport	Ahlsell	Skive ISO 7089 HV200 A4 Syrefast	384741	1
Fergeport	Ahlsell	Sekskantskrue DIN 931 delgjenget A4-70/A4-80 Syrefast	133560	1
Fergeport	Astrup	PMMA Satin DBL		1
Auto-PAX	Farnell	I/O modul pc	3994786	1
Fergeport	Oslo Gjerde AS	Rett U-RØR	ART-026	6
Fergeport	Oslo Gjerde AS	Gummi-U	Gummi-U	6

Arbeidsområde	Leverandør	Vare	Artikkelnr.	STK/meter
Fergeport	Oslo Gjerde AS	Rør-skjøt	ART-006-U	4
Fergeport	Oslo Gjerde AS	Endeløkk Flat	ART-32	4
Fergeport	Oslo Gjerde AS	Silikon	Silikon	1
Fergeport	Oslo Gjerde AS	T-Skjøt	ART- 004	4
Fergeport	Jens S, Transmisjoner AS	Tannhjul med 28 tenner	KM1,528SS	2
Fergeport	Jens S, Transmisjoner AS	Tannhjul med 60 tenner	KM1,560SS	2
Sikkerhetssone	RS Components	ABUS CCTV-kamera	218-4749	1
Sikkerhetssone	RS Components	Ethernetkabel, 10m	557-452	1
Sikkerhetssone	RS Components	Photoelectric sensor, laser	205-4828	1
Sikkerhetssone	RS Components	Reflektor	324-1620	1
Sikkerhetssone	ELFA Distrelec	DS-BOX-AA-NX2, 250gb, 8gb ram	302-38-571	1
Sikkerhetssone	ELFA Distrelec	DS-BOX-AA-250, 250gb, 4gb ram	302-38-568	1
Auto-PAX	ELFA Distrelec	FR 8 WP - 8 OHM, høytaler	301-97-306	4
Auto-PAX	Komplett Norge	Sound Blaster PLAY!3	1154839	1
Auto-PAX	Komplett Norge	II-MJACKM/M-B050	917243	1
Auto-PAX	Digikey	14 Flerpolet rekkeklemme Plugg, Phoenix Contact	1790344	2
Auto-PAX	Digikey	4 Flerpolet rekkeklemme Plugg, Phoenix Contact	1708595	2
Landport	Mairs Turnstile	Round pillar swing turnstile	MT311	1

J - I/O liste - WAGO I/O-maskin

Slot 2 - Serial module (DMX)

Channel	Description
D0 (RST/Z)	DMX +
D1 (TxD/Y)	DMX -
D2 (CTS/A)	DMX +
D3 (RxD/B)	DMX -
M (ground)	GND
S (Shield)	Shield

Slot 3 - DI

Channel	Description
DI1	Fore Hatch 0%
DI2	Fore Hatch 100%
DI3	Fore Hatch 50%
DI4	Aft Hatch 0%
DI5	Aft Hatch 100%
DI6	Aft Hatch 50%
DI7	Hatch EM stop
DI8	S8.1
DI9	S8.2
DI10	S8.3
DI11	S8.4
DI12	Fore Gate closed
DI13	Fore Gate open
DI14	Aft Gate closed
DI15	Aft Gate open
DI16	Fore/Aft ir gate sensor

Slot 4 - DO

Channel	Description
DO1	Horn
DO2	Fore Hatch close
DO3	Fore Hatch open
DO4	Aft Hatch close
DO5	Aft Hatch open
DO6	
DO7	
DO8	H8.1
DO9	H8.2
DO10	H8.3
DO11	Fore Gate close
DO12	Fore Gate open
DO13	Aft Gate close
DO14	Aft Gate open
DO15	
DO16	

Slot 5 - AI

Channel	Description
AI1	ZT1 (fore port dist)
AI2	ZT2 (fore stb. dist)
AI3	ZT3 (aft stb. dist)
AI4	ZT4 (aft port dist)

K - Populærvitenskapelig artikkel



Bachelorgruppen på milliAmpere2 ved Fosenkaia.

(Fv.: Lars Erik Langsåvold, Morgan Boe Johansen, Andreas Dollis Vik, Eline Berg)

Passasjerhåndtering for den selvkjørende fergen milliAmpere2

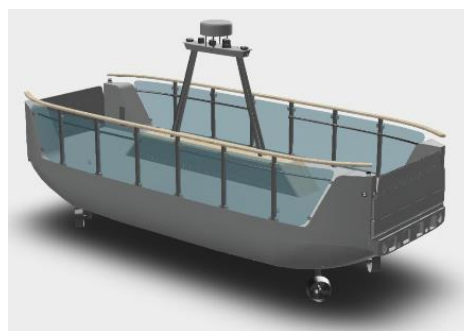
Av: Andreas Dollis Vik, Eline Berg, Lars Erik Langsåvold og Morgan Boe Johansen

Før i tiden brukte man fløttmenn til å ro passasjerer og gods over sund eller elver. Som passasjer kunne man da gå til fløttmennenes tilholdssted og ringe i en bjelle ved behov for transport. På 60-tallet så man på fløttmenn som vanlig og som en viktig del av infrastrukturen.

I dag, flere tiår senere, er fløttmennene mulig å erstatte ved hjelp av tekniske løsninger og selvkjøring. Det er kostbart å bygge broer og det kan ødelegge for skipstrafikk, og i allerede trange urbane strøk kan slike ferger være løsningen.

Egil Eide, førsteamanuensis ved Institutt for elektroniske styresystemer, kom i 2016, sammen med Kystlaget opp med en helt ny idé: «Hva med å utvikle en elektrisk selvgående ferge i stedet for å bygge en bro som kunne skape trøbbel for båttrafikken?». Med denne idéen kom startskuddet for utviklingen av Norges aller første selvkjørende ferge, milliAmpere1. Prosjektet har til hensikt å være et forskningsprosjekt for optimalisering og nyskaping av teknologiske løsninger.

Prosjektet har hatt en stor vekst siden 2016 og er nå på prototype nummer to, milliAmpere2. Etter sjøsetting var fergen i prøvedrift mellom Ravnkloa og Fosenkaia i tre uker høsten 2022. Fergen fraktet passasjerer i prøveperioden, og gav et bra inntrykk for muligheter innen selvkjørende ferger.

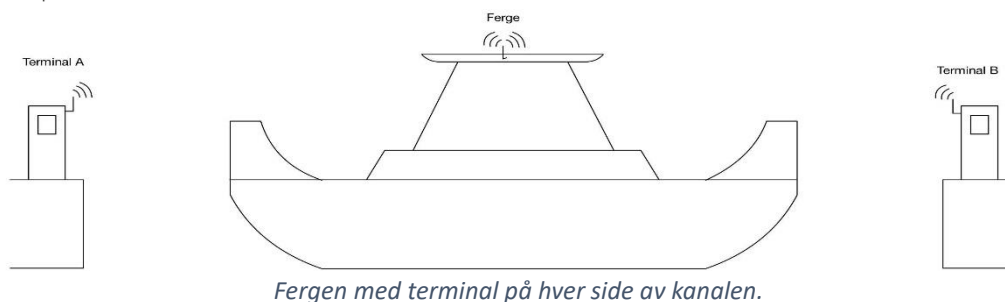


Digital skisse av milliAmpere2.

Fergen kjøres i dag med en sikkerhetsperson som skal starte fergen når alle er trygt om bord. Ansvarsområdet sikkerhetspersonen har er mulig å løse teknologisk, og målet er å flytte denne personen til et landbasert kontrollrom. Ved å gjøre dette kan vedkommende gjøre andre arbeidsoppgaver og være tilstede som er ekstra sikkerhet.

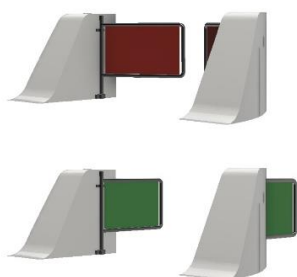
For å kunne lage et solid og robust system har gruppen samarbeidet med andre fagfelt, og kommet opp med et forslag som er mulig å realisere.

Per dags dato finnes det kun en lem som sørger for at passasjerene holder seg på båten eller på land. Dette vil for mange føles utrygt, og det er derfor nødvendig å beskytte passasjerer fra bevegelige deler.



Fergen med terminal på hver side av kanalen.

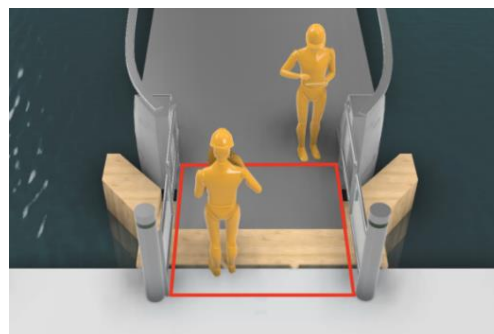
For å hindre nettopp dette er det designet og produsert porter. Det vil være fire porter på fergen, og fire på land, to på hver side. Portene skal være en fysisk hindring fra bevegelige deler og indikere med lys etter trafikklysmodellen. Ved bevegelse i porter eller lem, skal det indikeres fare med rødt blinkende lys og lydsignal. Når alt er klart, skal portene markeres med grønt lys. Portene er designet for å minimere sjansene for klemskader og ødeleggelse, og innsetting av porter er bare ett steg nærmere en sikker og trygg løsning.



Digital skisse av portene om bord på fergen. Rødt lys i bevegelse og grønt lys når portene er åpne.

En ønsker ikke at passasjerer havner mellom portene på land eller at bevegelige deler går hvis det kan være en fare. For å hindre dette er det definert en sikkerhetssone og et kamera som skal fange opp objekter i denne faresonen. Dersom det skjer en farlig hendelse varsles fergen og sender et signal til terminalen på den gitte landsiden. Det samme skjer om noe skulle varsles om på terminalsiden. Bevegelser av porter eller lem skal da stoppe momentant.

Systemet skal da varsle om fare, og stoppe eventuelle bevegelige deler. Fergen har også mulighet for å spørre landterminalen om det ligger objekter i vannet før fergen skal legge til kai. På denne måten får du en ekstra sikkerhet og kan unngå klemskader.



Faresonen markert i rødt, sett fra et fugleperspektiv.

Det er videreutviklet et styresystem for disse løsningene. Systemarkitekturen bak dette er en innviklet prosess med mange parametere som må tas hensyn til. I denne prosessen foregår alt som har med kommunikasjon mellom fergen og land-terminal, behandling av datapakker og all informasjonsflyt går igjennom dette systemet.

Når sikkerheten og portene er montert og testet, er det på tide å flytte sikkerhetsoperatøren til land. Det vil på fergen være plassert ut hurtigstopper som en kan aktivere ved ubehag.

Mellom portene blir det plassert en hurtigstopp i tillegg til hurtigstoppen om bord, som passasjerene kan trykke på om de føler ubehag. Sikkerhetspersonen på land vil da koble seg til over høytaleranlegget og berolige situasjonen, og eventuelt gjøre manuelle inngrep.

Denne løsningen er høyaktuell for å utvide til en sikrere og mer intuitiv opplevelse. Kanskje møtes vi på fergen om ikke lenge, og du kan få prøve denne høyt teknologiske oppfinnelsen!

