

Idar Evenstad
Magnus Dag Oseid
Petter Wandsvik
Thomas Jøntvedt Engum

Trådløs terminal for "on-demand" bestilling av autonom ferge

Bacheloroppgave i Elektroingeniør - BIELEKTRO

Veileder: Olav Myrvang

Medveileder: Egil Eide, Morten Breivik, Ole Andreas Alsos

Mai 2023



NTNU

Kunnskap for en bedre verden

Idar Evenstad
Magnus Dag Oseid
Petter Wandsvik
Thomas Jøntvedt Engum

Trådløs terminal for "on-demand" bestilling av autonom ferge



Bacheloroppgave i Elektroingeniør - BIELEKTRO
Veileder: Olav Myrvang
Medveileder: Egil Eide, Morten Breivik, Ole Andreas Alsos
Mai 2023

Norges teknisk-naturvitenskapelige universitet
Fakultet for informasjonsteknologi og elektroteknikk
Institutt for elektroniske systemer



Kunnskap for en bedre verden

Anerkjennelser

Denne oppgaven markerer avslutningen på et 3 år langt bachelorstudium innenfor elektroingeniør. Gruppen ønsker i den sammenheng å takke alle bidragsgivere til prosjektet.

Det rettes en takk til intern veileder Olav Myrvang, som har vært behjelpelig med både kunnskap og veiledning for å kunne gjennomføre oppgaven på en tilfredsstillende måte.

En takk rettes også mot Egil Eide, vår oppdragsgiver, samt med-veilederne Morten Breivik og Ole Andreas Alsos. Deres tilbakemeldinger, råd og tilgang til utstyr har vært avgjørende for prosjektets suksess.

En spesiell takk går til Glenn Angell som har bidratt til å designe, bestille og sveise innkapslingen til terminalene. Hans ekspertise og innsats har vært uvurderlig for å få terminalene realisert.

Til slutt ønsker vi å takke Anders Ziener Kristensen for god hjelp tilknyttet webdesign og et imøtekommende brukergrensesnitt.

Signaturer

Idar Evenstad Dato: 21.05.23

Idar Evenstad

idar.evenstad@ntnu.no

Thomas Jøntvedt Engum Dato: 21.05.23

Thomas Jøntvedt Engum

thomas.j.engum@ntnu.no

Magnus D. Oseid Dato: 21.05.23

Magnus Dag Oseid

magnus.d.oseid@ntnu.no

Petter Wandsvik Dato: 21.05.23

Petter Wandsvik

Petter.wandsvik@ntnu.no

Innholdet i denne oppgaven er fritt tilgjengelig. Publisering og gjengivelse gjøres med henvisning til forfattere og avhandlingen.

Tittelside Bacheloroppgave BIELEKTRO

Oppgavetittel (Norsk og Engelsk): Trådløs terminal for “on-demand” bestilling av autonom ferge Wireless terminal for “on-demand” ordering of autonomous ferry	
Forfattere: Idar Evenstad idar.evenstad@ntnu.no Magnus Dag Oseid magnus.d.oseid@ntnu.no Petter Wandsvik petter.wandsvik@ntnu.no Thomas Jøntvedt Engum thomas.j.engum@ntnu.no	Prosjektnummer: E2318 Innleveringsdato: 22.05.2023 Gradering: [x] åpen [] lukket
Studium: Elektroingeniør - BIELEKTRO	
Studieretning: Elektronikk og sensorsystemer	
Veileder internt: Olav Myrvang Institutt: Institutt for Elektroniske Systemer	
Oppdragsgiver: NTNU Institutt for Elektroniske Systemer Kontaktperson: Egil Eide, Epost: egil.eide@ntnu.no , til.: 416 91 512	
Sammendrag (Norsk og Engelsk): I dette prosjektet har det blitt utviklet et trådløst bestillingssystem for den autonome fergen milliAmpere2. Systemet er lagd for å kunne fungere på et 150 m langt strekke mellom Ravnkloa og Vestre Kanalkai, i Trondheim. Det er i den sammenheng utviklet et sett terminaler, med hver sin bestillingsknapp og skjerm, som skal operere utendørs. Det medfører at terminalene er utviklet til å tåle vann, hvor det i tillegg foreligger forslag til ytterligere forbedringer, for å sikre terminalenes elektronikk. Skjermen gir brukeren direkte tilbakemelding på sin bestilling, oppdatering på fergens status og dens neste destinasjon. I tillegg er det utviklet et ferge-modem, som kommuniserer trådløst med terminalene, og dirigerer beskjeder til og fra fergen. In this project, a wireless ordering system has been developed for the autonomous ferry milliAmpere2. The system is designed to function across a 150 m stretch between Ravnkloa and Vestre Kanalkai in Trondheim. As part of this, a set of terminals have been developed, each with an ordering button and a display, which are to be operated in an outdoor environment. This entails that the terminals are waterproof, where additionally there are proposals for further improvements, for the purpose of protecting the electronic devices within. The display provides the user with direct feedback on their order, updates on the ferry's status and its next destination. Additionally, a ferry-modem has been developed to route messages between the ferry and the terminals, using wireless communication.	
Stikkord norsk: Trådløst kommunikasjonssystem, LoRa, bestillingsterminal, brukergrensesnitt, vandalsikker, vanntett	Stikkord engelsk: Wireless communication system, LoRa, order terminal, user interface, vandal proof, waterproof

Innholdsfortegnelse

Forkortelser	iv
Definisjoner	v
1 Innledning	1
1.1 Bakgrunn for oppgaven	1
1.2 Oppgavebeskrivelse	2
1.3 Mål for prosjektet	6
1.4 Utforming av rapporten	6
2 Teoretisk rammeverk	7
2.1 Trådløs kommunikasjon	7
2.1.1 LoRa - Long Range	8
2.1.2 Rekkeviddefaktorer	8
2.1.3 Antenner	9
2.1.4 RSSI - Received Signal Strength Indicator	10
2.1.5 Linkbudsjett	10
2.1.6 Koaksialkabel	11
2.2 API - Application Programming Interface	12
2.3 FIFO - First-In-First-Out	12
2.4 Seriell-kommunikasjon	12
2.4.1 UART - Universal Asynchronous Receiver and Transmitter	13
2.4.2 RS485	14
2.5 Brukergrensesnitt	15
2.5.1 Trykk-knapp	15
2.5.2 SBC - Single Board Computer	16
2.5.3 JavaScript	16
2.5.4 Node.js	16
2.5.5 React	17
2.6 Standarder for klassifiseringer av innkapsling for elektriske systemer	17
2.6.1 IP-rangering - Ingress Protection	17
2.6.2 IK-rangering - Impact Protection	18
3 Metode	19
3.1 Prosjektoppgavens utgangspunkt	19
3.1.1 Skisser og funksjonsbeskrivelse	20
3.2 Valg av trådløs kommunikasjon	21
3.2.1 Valg av protokoll	21
3.2.2 Valg av av utviklingskort	24
3.2.3 Valg av antenner	25
3.3 Utvikling av programvare	28
3.3.1 Programvare tilknyttet LoRa- og SAM R34-funksjonalitet	28
3.3.2 API for kommunikasjon i systemet	31
3.3.3 Oppsett og konfigurering av UART	34
3.3.4 Systembeskrivelse av bestillingssystem	35
3.3.5 Implementering av FIFO-struktur	37
3.3.6 Grafisk brukergrensesnitt - GUI	38
3.3.7 Nettsidedesign	39
3.3.8 Server og klient	42

3.3.9	Implementering og oppsett av UART på RPi 4B	44
3.4	Designprosess for terminal	44
3.4.1	Skjerm	45
3.4.2	Skjermdriver	46
3.4.3	Temperaturtest av RPi 4B	47
3.4.4	Trykk-knapp	49
3.4.5	Vann- og vandalsikring	49
3.4.6	Vanntest på terminal	54
3.5	Modem på ferge	55
3.6	Utvidelseskort	56
3.7	Oppsett og konfigurering av RS485	59
3.8	Signalstyrke og parametere	62
3.9	Utstyrliste	65
3.9.1	Ferge	65
3.9.2	Terminal	65
3.9.3	Programvare	66
3.10	Oppsett, test og simulering av systemet	67
3.10.1	LoRa-rekkevidde	68
3.10.2	Trådløs og seriell signalflyt	71
3.10.3	Signalflyt til/fra Autopax/PHS	73
3.10.4	Programflyt ved bestilling av ferge	75
3.11	Simulering av bestilling av ferge	78
4	Resultater	79
4.1	Trådløs kommunikasjon mellom Ravnkloa og Vestre Kanalkai	79
4.2	Brukergrensnitt	81
4.3	Terminal	83
4.3.1	Vannsikring av terminal	83
4.3.2	Vandalsikring av terminal	84
4.3.3	Organisering av elektronikk i terminalene	89
4.3.4	Ferdigstilling av terminal	90
4.4	Innkapsling av ferge-modem	91
4.5	Fysisk simulering og tilbakemeldinger på bestillingssystemet	91
5	Drøfting	94
5.1	Vurdering av resultater	94
5.1.1	Trådløs kommunikasjon og signalstyrke	94
5.1.2	Vanntest	95
5.1.3	Oppkobling av elektronikk i terminalene	96
5.1.4	Tilbakemeldinger fra brukertesten	96
5.2	Produkt levert til oppdragsgiver	97
5.2.1	Backupløsning tilknyttet strømforsyning	97
5.2.2	Installasjon av terminaler	99
5.2.3	Installasjon av modem	99
5.2.4	Vedlikehold av maskinvare	100
5.2.5	Oppgradering av programvare	100
5.3	Optimering og fremtidig arbeid	100
5.3.1	Effektivisering av programvare	100
5.3.2	Tilfelle med mulighet for unødvendige seilas	101
5.3.3	Meldingsbekreftelse ved mottatt trådløst signal	101
5.3.4	API	101

5.3.5	Vann- og vandalsikring	102
5.3.6	Universell utforming	104
5.4	Prosjektgjennomføringen	105
6	Konklusjon	106
A	AFT - API	112
B	Guide for oppsett av Raspberry Pi 4 med GUI	127
C	Design av terminal	134
D	Pidestall og terminal	135
E	Risikoanalyse	136
F	Heksadesimalt til dBm	138
G	Brukeropplevelse undersøkelse	139
H	Mail ang. datablad for skjerm	140
I	Fullstendig deleliste	141
J	Budsjett for utstyrsinnkjøp	142
K	Populærvitenskapelig artikkel	143

Forkortelser

Forkortelse	Forklaring
API	Application Programming Interface
dB	Decibel
dBi	Decibel relative isotropic
dBm	Decibel relative milliwatts
DC	Direct Current
DE	Driver Enable
FFD	Full Functional Device
FSPL	Free Space Path Loss
GPIO	General Purpose Input Output
GUI	Graphical User Interface
HDMI	High-Definiton Multimedia Interface
HTML	HyperText Markup Language
I/O	Input/Output
ID	Identification
IDE	Integrated Development Environment
IK	Impact Protection (K = Kinetic, for å skilles fra IP-rating)
IoT	Internett of Things
IP	Ingress Protection
JS	JavaScript
LED	Light Emitting Diode
LoRa	Long Range
LSB	Least Significant Bit
MSB	Most Significant Bit
NC	Normally Closed
NO	Normally Open
NTNU	Norges Teknisk-naturvitenskapelige Universitet
NVM	Non-Volatile Memory
P2P	Peer to Peer
PAN	Personal Area Network
PHS	Passasjer-håndteringssystem
RE	Reciever Enable
RPi	Raspberry Pi
RSSI	Received Signal Strength Indicator
SBC	Single Board Computer
UPS	Uninterruptible Power Supply
USB	Universal Serial Bus
UTP	Unshielded Twisted Pair
WiFi	Wireless Fidelity

Definisjoner

Definisjon	Forklaring
Argument	Data som sendes til en funksjon når funksjonen kalles.
Avbrudd	Brudd i normal programflyt for en mikrokontroller.
Data Rate	Antall bits per sekund.
Klient	Programvare som brukes til å få tilgang til web-innhold.
Medlemsvariabel	Variabler i en struct.
Node	Enhet i et nettverk som kan motta, sende eller videreformidle data og informasjon til andre noder i nettverket.
Parameter	Variabel som mottar et argument som blir matet til en funksjon.
Peker	En variabel som inneholder adressen til en variabel.
Protokoll	Refererer til en samling av regler og standarder som angir hvordan enheter skal kommunisere og utveksle data.
Server	Enhet som håndterer forespørsler fra klient, behandler data og returnerer resultatene tilbake til klient via nettverk.
Struct	En samling av en eller flere variabler i programmeringsspråket C.

1 Innledning

1.1 Bakgrunn for oppgaven

milliAmpere2, som illustreres i figur 1, er et pågående prosjekt ved NTNU (Norges teknisk-naturvitenskapelige universitet) som fokuserer på utviklingen av en autonom ferge. Prosjektets mål er å tilby samfunnet en pålitelig transportløsning for å krysse kanalen i Trondheim, spesifikt fra Ravnkloa til Vestre Kanalkai, ved bruk av milliAmpere2.

Prosjektet ble først satt i gang i 2016, samtidig som Trondheimsfjorden ble innviet som det første testområde i verden for autonome og førerløse fartøy (Haugan 2022). Det utledet prototypen milliAmpere, forgjengeren til milliAmpere2, som fokuserte på utvikling av teknologi tilknyttet autonome fartøy. milliAmpere2 er nå en fullskala prototype og inkluderer forskning rundt hvordan mennesker skal fraktes ved hjelp av autonome fremkomstmidler (Norwegian Schitech New 2023).

milliAmpere2 ledet til spinnoff-selskapet Zeabuz, som utvikler teknologi for maritime autonome fremkomstmidler (Zeabuz 2023). Tjenesten Zeam vil i samarbeid med Zeabuz og Torghatten, sommeren 2023, implementere en autonom ferge, med navn MF Estelle, som vil frakte passasjerer fra Kungsholmen til Söder Mälarstrand, i Stockholm. Fergen vil være i skytteltrafikk og ha faste avganger hvert 15. minutt (Zeam 2023). Det vil altså ikke være menneskelig interaksjon for å benytte fergen, i motsetning til milliAmpere2 hvor personer skal bestille fergen. milliAmpere2 vil på mange måter fungere som en horisontal heis, hvor du trykker for å indikere at du vil bruke heisen.

Fergen i Stockholm og milliAmpere2 kan gi verdifull informasjon til hvordan ferger og andre autonome fremkomstmidler bør videreutvikles for å bli en sentral del av bymiljø og samfunn i fremtiden.



Figur 1: Fergen milliAmpere2 (Faculty of Information Technology and Electrical Engineering NTNU's photos 2021).

1.2 Oppgavebeskrivelse

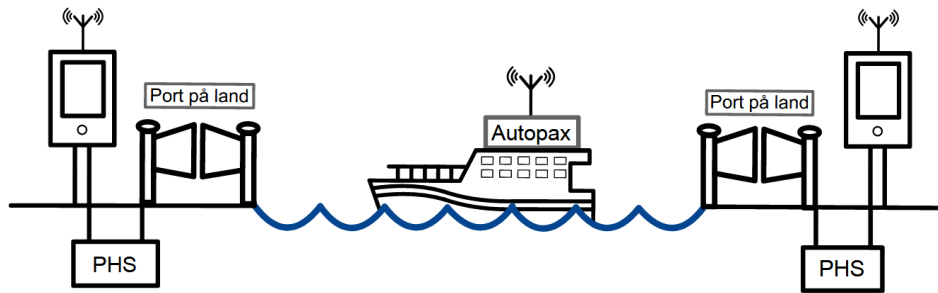
Rapporten tar for seg bacheloroppgaven gitt ved studieprogrammet Elektroingeniør hvor gruppen har valgt spesialisering innenfor Elektronikk og Sensorsystemer.



Figur 2: Forenklet systembeskrivelse som illustrerer kommunikasjonen mellom terminalene og milliAmpere2.

For at mennesker skal kunne benytte seg av milliAmpere2 må det foreligge et system hvor man bestiller og indikerer at man vil ta i bruk fergen for å komme over til den andre siden av kanalen. Gruppens bacheloroppgave bygger på denne ideen hvor hensikten er å utvikle et bestillingssystem. Systemet består av et modem plassert på milliAmpere2 og to bestillingsterminaler plassert på hver side av kanalen, heretter omtalt individuelt som terminal A og terminal B, slik som illustrert i figur 2. Terminal A representerer Ravnkloa, og terminal B, Vestre Kanalkai. Disse terminalene vil kommunisere trådløst med ferge-modemet og videre indikere når personer ønsker å benytte seg av tjenesten.

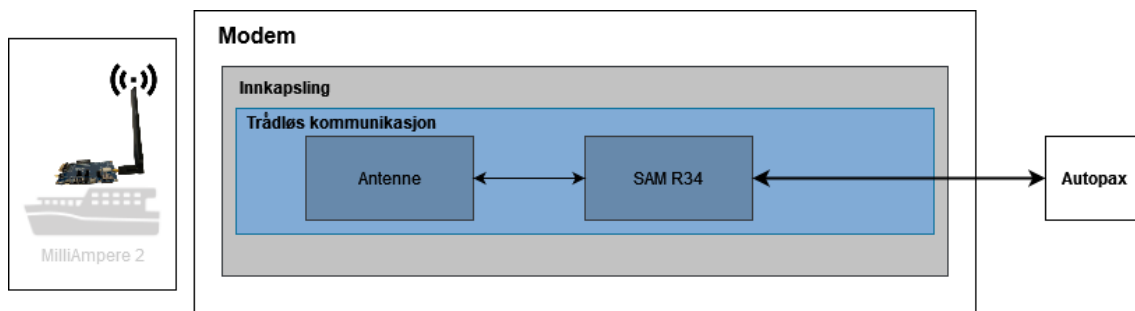
Terminalene har i tillegg oppgaven å viderefremde signaler mellom fergen og passasjerhåndteringssystemet (PHS) som er plassert på land. PHS omfatter blant annet sikkerhetsporter, som vil åpnes og lukkes ved på- og avstigning. PHS administreres av en bachelorgruppe fra Institutt for teknisk kybernetikk (ITK), heretter referert til som ITK-gruppen (Berg mfl. 2023).



Figur 3: Illustrasjon av oppgavens grensesnitt mot ITK-gruppen.

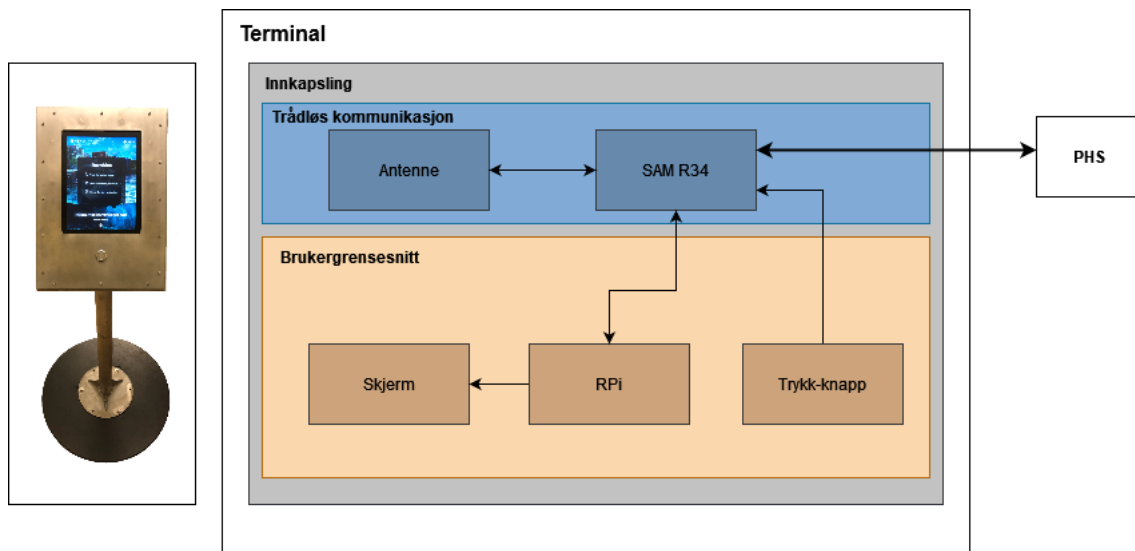
Det er ferge-modemet som står for mesteparten av kontrollen i kommunikasjonssystemet. Det har også i oppgave å kommunisere med en datamaskin om bord i fergen. Denne datamaskinen, heretter omtalt som Autopax, mottar og sender signaler som åpning og lukking av portene på land til ferge-modemet. Autopax-maskinen er også en del av ITK-gruppens bacheloroppgave.

Illustrasjonen i figur 3 viser hvordan systemene henger sammen. Terminalene på land kommuniserer med PHS som igjen styrer portene på land. På fergen er ferge-modemet koblet opp til Autopax som igjen kommuniserer med resten av fergen. Figur 4 viser ferge-modemets systemarkitektur.



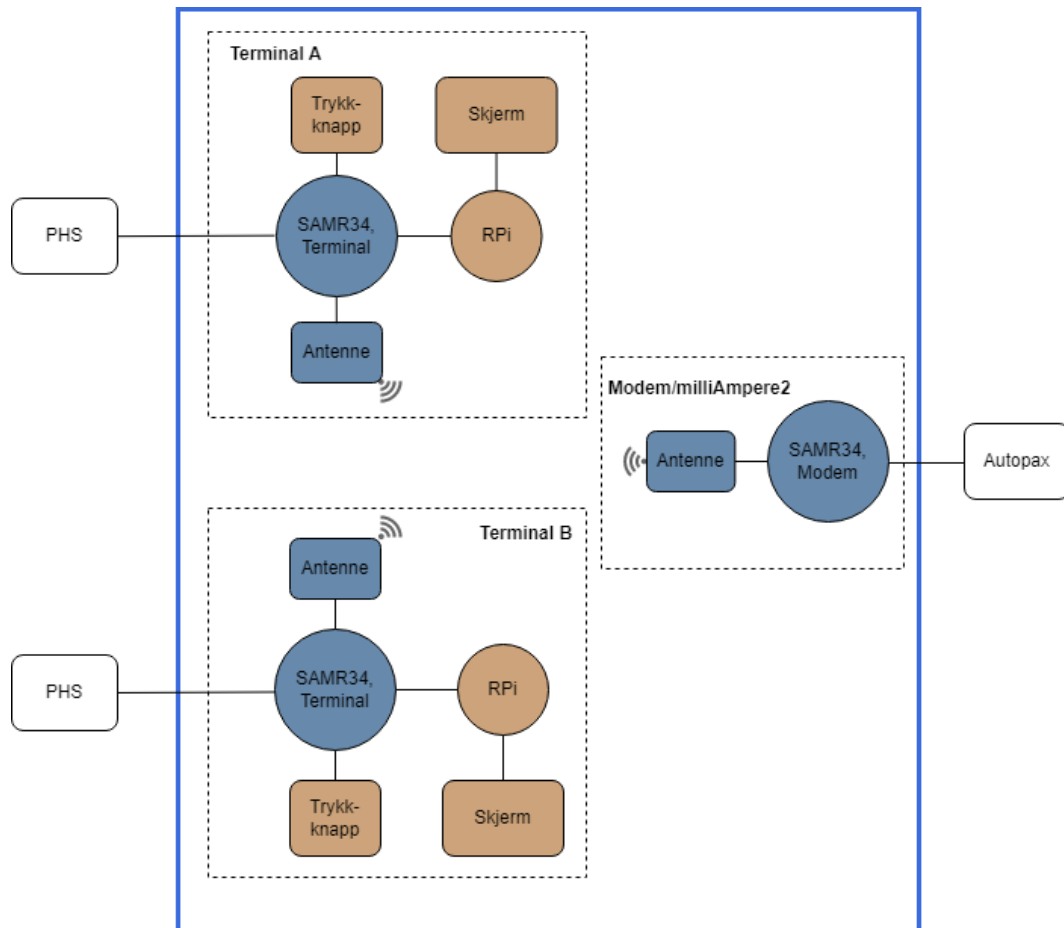
Figur 4: Ferge-modemets forenklete systemarkitektur.

Terminalene er nødvendige for at mennesker skal kunne bestille fergen. De inkluderer en knapp som kan brukes til å bestille fergen, samt en skjerm som kan fremstille og gi beskjed om fergens status til ventende passasjerer. Sammen med en liten datamaskin utgjør disse komponentene terminalens brukergrensesnitt. Terminalene er tenkt til å skulle stå ute og har derfor et behov for å tåle vær og vind. De må også være vandalsikre. Figur 5 viser terminalenes systemarkitektur.



Figur 5: Terminalenes forenklete systemarkitektur.

Avgrensningen til bacheloroppgaven ligger i grensesnittet mot PHS og Autopax-maskinen, hvor et klart skille vil gå frem til, men ikke med, behandling av informasjon som kommer inn til PHS og Autopax. Det kan derfor stadfestes at bacheloroppgavens grense er i enden av transmisjonsmediet som benyttes til den serielle informasjonen til Autopax-maskinen og PHS. Oppgaven omfavner med andre ord hverken porter på land eller ferge, som tidligere er beskrevet.



Figur 6: Illustrasjon av nettverkets hovedkomponenter med avgrensning av oppgave markert med blått rektangel.

Figur 6 viser en forenklet systemoversikt med systemets hovedkomponenter. Komponentene som ligger innenfor det blå rektangelet inngår i denne oppgaven og som figuren viser kommuniserer systemet med Autopax-maskinen og PHS som ligger utenfor gruppens ansvarsområde. Det er totalt tre noder i nettverket som kommuniserer trådløst med hverandre. Terminalene og modemmet kommuniserer serielt med hver sin Raspberry Pi, som i sin tur fungerer som skjermdrivere for terminalenes skjermer. Terminalene har videre hver sin trykk-knapp for å kunne motta bestillinger fra passasjerer som ønsker å benytte ferden.

Med utgangspunkt i at systemet opererer som bestillingsterminaler for passasjerer, samt en mellommann for informasjonsstrøm mellom Autopax og PHS, vil det totale bestillings- og passasjer-håndteringssystemet være avhengig av begge gruppens funksjonalitet for å skape en komplett løsning.

1.3 Mål for prosjektet

- Tilrettelegge for bestilling av milliAmpere2.
- Utvikle trådløst kommunikasjonssystem mellom ferge-modem og terminalene.
- Samhandling mellom terminalene og passasjer-håndteringssystemet.
- Samhandling mellom ferge-modem og Autopax-maskin.
- Utvikle terminalløsning som tilrettelegger for brukervennlighet.
- Utvikle løsninger i tråd med omgivelsene systemet skal operere i. Det inkluderer å sikre elektronikk og komponenter mot vær og vandalisme.
- Utvikle et intuitivt brukergrensesnitt.

1.4 Utforming av rapporten

Rapporten er strukturert i henhold til en klassisk vitenskapelig struktur som inkluderer teoretisk rammeverk, metode, resultat, drøfting og konklusjon. Det teoretiske rammeverket gir en grundig gjennomgang av relevant teori som kreves for å forstå metodene som ble anvendt for å oppnå de beskrevne resultatene. Metode-delen beskriver systematisk fremgangsmåte, valg, vurderinger og delresultat som ble benyttet for å produsere de oppnådde resultatene, mens resultat-delen presenterer prosjektets hovedresultater. Drøftingsdelen gir en kritisk evaluering av resultatene og foreslår mulige forbedringer for fremtidig arbeid. Rapporten avsluttes med en konklusjon som oppsummerer hovedfunn og gir anbefalinger for videre forskning.

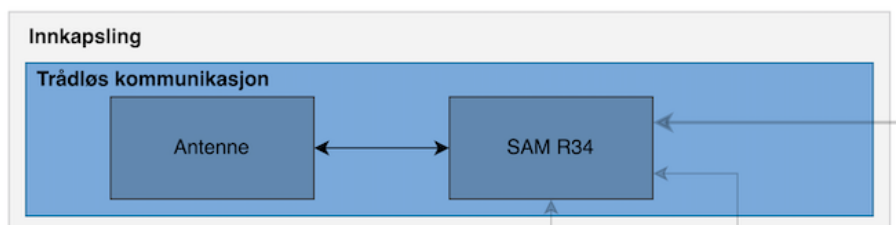
Det er gjennom prosjektet utviklet kode samt kretskort i forbindelse med både terminalene og modemmet. Alle filer er å finne på Github-repository-et ved følgende link: https://github.com/Ocheid/Wireless_network_ordering_milliAmpere2. Det er også et Github-repository for oppsett av brukergrensesnitt på Raspberry Pi, som er å finne her: <https://github.com/Engum/E2318-Terminal>.

2 Teoretisk rammeverk

For å forstå arbeidet presentert i rapporten er det nødvendig å ha kjennskap til de underliggende komponentene, prinsippene og konseptene som er anvendt. Det teoretiske rammeverket tar for seg teorien knyttet til den trådløse kommunikasjonen som er valgt, samt faktorer som påvirker signalstyrken. Videre presenteres nødvendig teori som underbygger valget av antenner, og seriell kommunikasjon for å forstå hvordan de ulike delsystemene kommuniserer. I prosjektet er det behov for å designe et brukergrensesnitt. For å oppnå dette anvendes teknikker og metoder hentet fra web-design. Teorien forklarer til slutt grunnleggende rangeringer for vann- og vandalsikring, for å legge grunnlaget for valg av komponenter og designprosess av terminalene.

2.1 Trådløs kommunikasjon

Et av de mest sentrale problemene i dette prosjektet ligger i kommunikasjon mellom tre individuelle noder som ikke kan kommunisere via kabel. Trådløs kommunikasjon er en teknikk for å overføre informasjon over en avstand uten bruk av ledning eller kabel (Frenzel Jr 2023). For å oppnå tiltenkt funksjonalitet må en rekke valg og faktorer tas i betraktning, slik som valg av kommunikasjonsprotokoll og antenner, samt rekkeviddefaktorer. Det er derfor nødvendig å vurdere disse faktorene nøye når man planlegger og implementerer trådløs kommunikasjon i et system. Figur 7 viser systemets trådløse arkitektur bestående av en antenne og sender/mottaker.



Figur 7: Modem og terminalenes trådløse arkitektur.

I en fysisk kabel representeres ofte informasjonen ved spenningspulser som indikerer 0 eller 1 i binære verdier. Dette bruker datamaskiner til å behandle data og informasjon. Det er ikke mulig å representere trådløse signaler med spenning på samme måte, hvor dette problemet løses ved bruk av modulasjonsteknikker. Signalmodulasjon betyr endring av et radiosignal, for eksempel ved å justere amplituden på bølgene, eller frekvensen til signalet. Mottaker av et trådløst signal kan hente ut informasjonen radiosignalet representerer, ved å se på hvilke endringer som er gjort (Frenzel Jr 2023).

2.1.1 LoRa - Long Range

LoRa er en trådløs modulasjonsteknikk som tillater å sende signaler over store avstander (Frenzel Jr 2023, s. 821). Den utkonkurrerer andre protokoller som Bluetooth og WiFi når det kommer til rekkevidde (The Things Network 2023b). Avhengig av omgivelsene kan LoRa-enheter kommunisere med hverandre fra avstander på få meter til flere kilometer. En annen viktig karakteristikk ved LoRa er at den krever lite effekt for å operere, og det gjør den ideell for applikasjoner som krever batteri (Semtech 2023b).

LoRa opererer i det lisensfrie “sub-gigahertz” spekteret som inkluderer 433 MHz, 868 MHz og 915 MHz (The Things Network 2023b).

LoRa-teknologien muliggjør P2P-kommunikasjon (Peer-to-Peer). Det innebærer trådløs kommunikasjon direkte mellom noder uten behov for et aksesspunkt som videreformidler signalet til riktig mottaker (SMART Modular Technologies 2022). LoRa P2P oppretter et eget PAN (Personal Area Network) som er et lite nettverk med direkte kommunikasjon mellom en eller flere noder over korte avstander (Frenzel Jr 2023, s. 489).

2.1.2 Rekkeviddefaktorer

LoRa har i hovedsak tre faktorer som spiller inn på hvor langt det er mulig å sende og motta et signal uten tap av informasjon. Det er sendereffekt, spredningsfaktor og båndbredde (Microchip Technology 2021c). Sendereffekt er hvor mye effekt som sendere produserer og leverer på utgangen (Frenzel Jr 2023).

Spreading Factor (For UL at 125 KHz)	Bit Rate	Range (Depends on Terrain)	Time on Air for an 11-byte payload
SF10	980 bps	8 km	371 ms
SF9	1760 bps	6 km	185 ms
SF8	3125 bps	4 km	103 ms
SF7	5470 bps	2 km	61 ms

Figur 8: Ulike verdier av spredningsfaktor som påvirker bit rate, rekkevidde og hvor lenge senderen bruker på å sende et trådløst signal (time on air), ved båndbredde lik 125 kHz (Semtech 2023b).

Spredningsfaktoren, også kjent som “spreading factor”, er en parameter som regulerer varigheten av en signaloverføring, det vil si den regulerer symboltiden (The Things Network 2023a). Et symbol er en samling av en eller flere bits i en dataoverføring (Frenzel Jr 2023, s. 906). Symboltiden utledes med følgende formel:

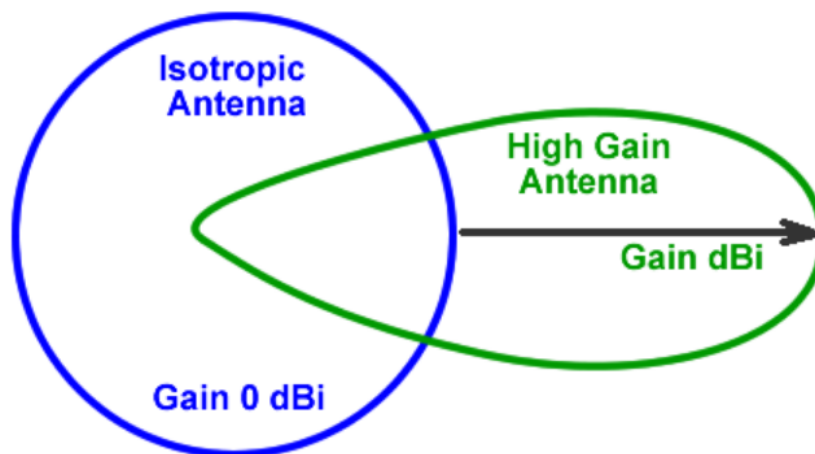
$$\text{Symbol tid} = 2^{\text{Spredningsfaktor}} \quad (1)$$

(Microchip Technology 2021c). Det gjør at en lavere spredningsfaktor gir kortere symboltid og fører til høyere datahastighet. Det går derimot på bekostning av rekkevidden til signalet. Spredningsfaktor er også med på å bestemme sensitiviteten til mottaker, hvor høyere faktor gir høyere sensitivitet. Spredningsfaktor programmeres til verdier mellom 7 og 12, avhengig av hva som er nødvendig for nettverket (The Things Network 2023a). Spredningsfaktoren og hvordan den påvirker ulike parametere kommer frem av figur 8.

Båndbredde er et valg mellom rekkevidde, energieffektivitet og data rate. En lavere båndbredde øker rekkevidden, men fører også til lavere effektivitet og data rate. Det er økt sannsynlighet for opphopning av data ved større båndbredde. Det betyr at det kan oppstå kø og data må vente til kanalen er ledig (Microchip Technology 2021c).

2.1.3 Antenner

Antenner er komponenter som omdanner elektriske signaler til elektromagnetiske bølger, og motsatt, for å overføre informasjon gjennom et trådløst medium (Frenzel Jr 2023, s. 530). Antenner oppgis til å ha en egen vinning, gitt i dBi, som beskriver vinning relativt til en isotropisk antenne. En isotropisk antenne er en teoretisk antenne som radierer likt i alle retninger (Perez 1998). En antenne i seg selv har ikke direkte vinning ettersom det er en passiv elektrisk komponent. Ved å stille inn direktiviteten, altså over hvor bredt område den fokuserer strålingen, omtales det som at den har vinning. Enheten dBi indikerer hvor mye antennen fokuserer strålingen i en retning og hvilken vinning det medfører (Frenzel Jr 2023, s. 552). Figur 9 illustrerer en isotropisk antenne og en antenne med direktivitet, og hvordan direktiviteten påvirker vinningen representert ved "Gain" i figuren.



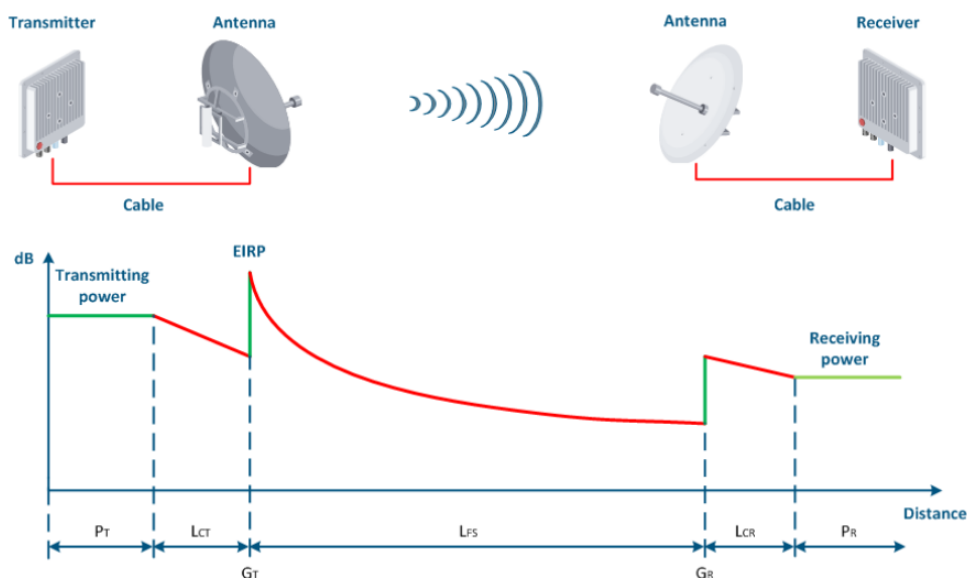
Figur 9: Stråling hos en isotropisk antenne i forhold til en direktiv antenne (*Understanding Antenna Gain, Beamwidth, And Directivity* 2023).

2.1.4 RSSI - Received Signal Strength Indicator

RSSI er et mål på styrken av et radiosignal i et trådløst miljø, for å kartlegge om det er tilstrekke signalnivå for å uthente informasjonen fra radiosignalet pålitelig. Det brukes ofte til å måle signalstyrken ved mottaker hvor høyere RSSI tilsvarer sterkere signal (Bhargava 2017).

2.1.5 Linkbudsjett

For å kartlegge hvorvidt det teoretisk er tilstrekkelig signalstyrke for dataoverføring mellom terminalene og fergen, benyttes et linkbudsjett. Det er en beregning som tar hensyn til alle tap og forsterkninger i signalstyrken fra en radiosender til en mottaker. Det inkluderer blant annet senderens effekt, kabeltap, antennevinning og tap som skyldes trådløs overføring, se figur 10. Linkbudsjettet utgjør et viktig verktøy for å undersøke om signalet har tilstrekkelig styrke, slik at mottaker kan motta informasjonen uten tap. Mottakerens sensitivitet beskriver minimum signalstyrke som kreves for å hente ut informasjonen fra et radiosignal (*Radio Link Budget: details & formula* 2023).



Figur 10: Signalstyrke fra sender til mottaker, hvor det oppstår forsterkninger og tap (InfiNet Wireless 2023). Grønt representerer et stabilt eller økt signalnivå, mens rødt representerer områder hvor det oppstår tap i signalnivået.

Det er vanlig å måle signalstyrke i decibel (dB). Decibel er en logaritmisk skala, og er hensiktsmessig siden man finner sluttstyrken ved å legge sammen forsterkningene og trekker fra tapene. Sendereffekt er ofte i milliwatt og representeres av den grunn med dBm, hvor 0 dBm er 1 mW (Frenzel Jr 2023, s. 40).

Sendereffekt, kabeltap og antennevinning finnes i datablad for de respektive kompo-

mentene. For å regne ut hvilket tap som oppstår når signalet sendes over et trådløst medium anvendes formelen for “Free Space Path Loss” (Blaunstein og Christodoulou 2007, s. 98):

$$FSPL [dB] = 10 \log \left(\frac{4\pi R}{\lambda} \right)^2, \quad (2)$$

$$FSPL [dB] = 10 \log \left(\frac{4\pi Rf}{c} \right)^2. \quad (3)$$

I (2) og (3) er

$FSPL$ = Tap i trådløst medium [dB],

R = Avstand fra senderantenne [m],

λ = Bølgelengde [m],

f = Frekvens [Hz],

c = Lyshastighet [m/s].

Det totale linkbudsjettet regnes ut ved å inkludere alle forsterkninger og tap i systemet (*Radio Link Budget: details & formula 2023*):

$$P_R = P_T + G_T - L_T - FSPL + G_R - L_R. \quad (4)$$

Parameterene i (4) er:

P_R = Styrken på signalet ved mottaker [dBm],

P_T = Styrken på signalet ved sender [dBm],

G_T = Senderantenne-vinning [dBi],

G_R = Mottakerantenne-vinning [dBi],

L_T = Sender kabeltap [dB],

L_R = Mottaker kabeltap [dB].

2.1.6 Koaksialkabel

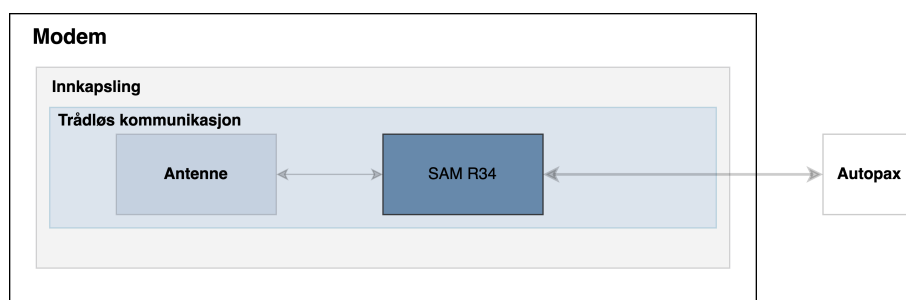
En koaksialkabel er en elektrisk ledende kabel som overfører signaler via strøm og spenning, fra et sted til et annet, gjerne mellom sender/mottaker og en antenne. Når et signal overføres gjennom en koaksialkabel vil det føre til at signalstyrken reduseres. Den vanligste grunnen til tap er resistive elementer i kabelen som genererer varme når det går strøm gjennom den. Dette fører til effekttap som vil ha innvirkning på linkbudsjettet. Hvor mye kabeltap, også kalt demping, som oppstår, avhenger av kvaliteten på kabelen, i tillegg til lengden (Frenzel Jr 2023, s. 457-458, 531).

2.2 API - Application Programming Interface

Prosjektets behov for å utveksle data mellom ulike plattformer og subsystemer krever en standardisert metode for informasjonsutveksling. Et API er et sett med definerte regler og standarder som kan brukes for å utveksle data mellom to programmer eller applikasjoner. API gir mulighet for en applikasjon å kommunisere med en annen, uten å måtte kjenne til alle detaljene i implementeringen av den andre applikasjonen. API-er gir også en måte for tredjepartsutviklere å bygge applikasjoner som kan utnytte funksjonaliteten til en eksisterende applikasjon, ved å benytte API-ene som er gjort tilgjengelige (IBM 2020).

2.3 FIFO - First-In-First-Out

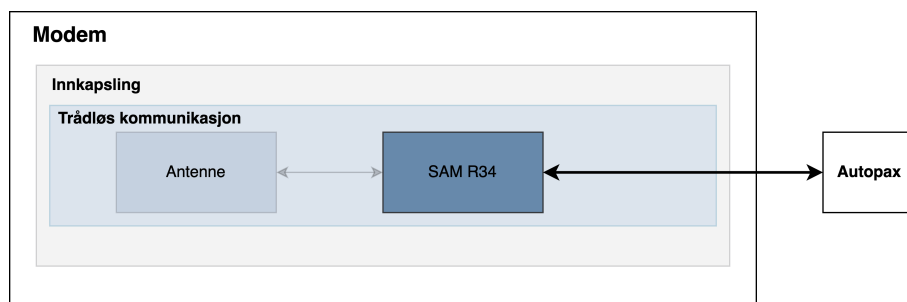
På bakgrunn av at fergen skal kunne bestilles fra begge sider av kanalen, er det behov for å ha oversikt over de ulike bestillingene som mottas og behandle de i ønsket rekkefølge. En FIFO-struktur er en vanlig datastruktur i programmering som organiserer elementer i rekkefølgen de ble lagt inn. Datastrukturen omtales også som en kø, og det er modemmet på fergen som har i oppgave å administrere strukturen, som fremhevet i figur 11. Elementene legges til i enden av køen, og fjernes fra starten (Cormen mfl. 2009). Dette gjør FIFO-strukturer nyttige i situasjoner der det er viktig å behandle data i samme rekkefølge som det mottas.



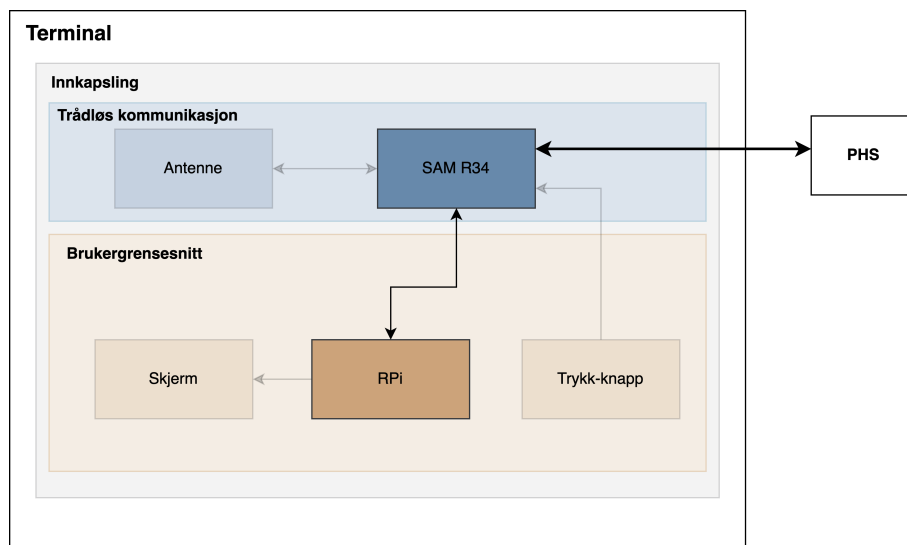
Figur 11: Enheten på ferge-modemet som administrerer bestillingskøen.

2.4 Seriell-kommunikasjon

I tilfeller hvor et system består av flere separate kontrollenheter som er avhengige av å utveksle informasjon med hverandre, kan seriell kommunikasjon benyttes. Det gjøres ved å implementere en protokoll som definerer kommunikasjonsflyten mellom enhetene. Figur 12 og 13 fremhever grensesnittene på modem og terminal hvor seriell kommunikasjon utveksles.



Figur 12: Modemets enheter som kommuniserer serielt.



Figur 13: Terminalenes enheter som kommuniserer serielt.

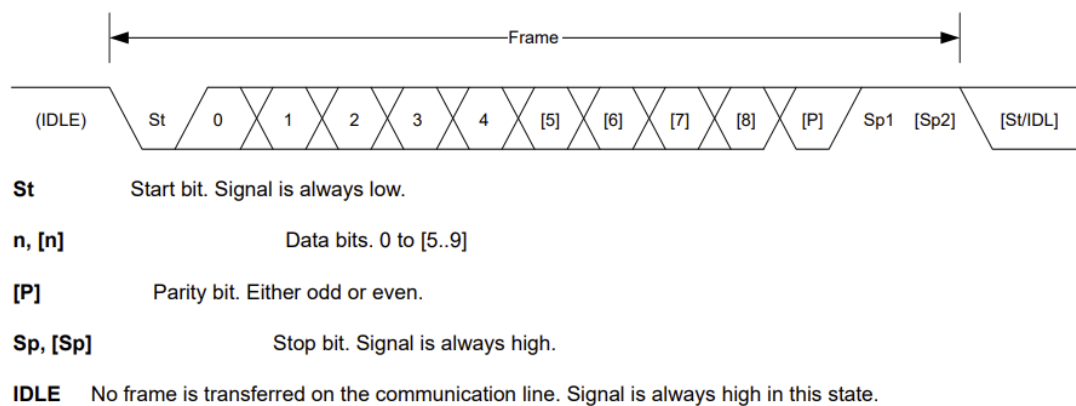
2.4.1 UART - Universal Asynchronous Receiver and Transmitter

UART er en velkjent kommunikasjonsprotokoll som brukes for å overføre seriell data mellom enheter. Protokollen har et bredt spekter av anvendelser, men brukes ofte for å etablere kommunikasjon mellom en mikrokontroller og andre enheter, for eksempel sensorer, motorer, skjermer og andre mikrokontrollere (Fang og Chen 2011). UART-protokollen er ansvarlig for å etablere og opprettholde pålitelig seriell kommunikasjon mellom enheter som bruker denne rammebaserte protokollen (Kurose og Ross 2013).

UART-protokollen kan operere i både synkron og asynkron modus. I synkron modus bruker kontrolleren en klokke og datalinje for å sende data. Kontrolleren sender et klokkesignal til periferienheten som gjør at den synkroniseres med datahastigheten til avsenderen, og data overføres over datalinjen. Dette gjør at hverken kontrolleren eller periferienheten trenger å konfigureres til lik datahastighet i forkant av kommunikasjonen. I asynkron modus opererer UART uten separat klokke, kun med en datalinje. Kontroller og periferienhet må derfor konfigureres til samme datahastighet før kommunikasjonen starter.

UART-rammen består av en rekke protokollinformasjon; startbit, databits, en valgfri

paritetsbit og en eller to stoppbit(s). Startbiten brukes til å indikere starten på en ny ramme, mens stoppbitene brukes til å indikere slutten. Paritetsbit brukes til feilkontrollering av databitene (Microchip Technology 2019). Figur 14 illustrerer protokollens rammeformat med startbit, databits, paritetsbit og stoppbit(s).



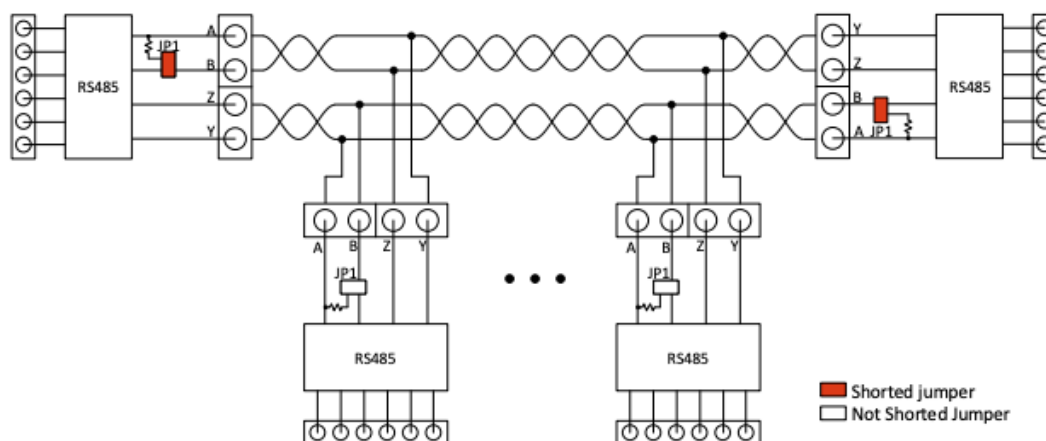
Figur 14: UART-protokollens rammeformat som illustrerer startbit, databits, paritetsbit og stoppbit(s) (Microchip Technology 2020a).

UART-støtter full duplex-kommunikasjon, som betyr at data kan overføres samtidig i begge retninger. Dette oppnås ved hjelp av separate sende- og mottaksbuffer. Kommunikasjonen inkluderer protokollinformasjon i hver ramme. Data i begge retninger lagres midlertidig i en buffer, som gjør det mulig med kontinuerlig dataoverføring uten forsinkelser og mulighet for avbruddsstyrt kommunikasjon (Microchip Technology 2019).

2.4.2 RS485

I systemet er det nødvendig å overføre seriell informasjon over betydelige avstander, som i sin tur kan introdusere støy i de serielle UART-rammene. RS485 (Recommended Standard #485) er en kommunikasjonsstandard som ofte er benyttet innenfor innebygde systemer for overføring av seriell data over lengre avstander. Bus-standarden tillater bi-direksjonell kommunikasjon ved sending av databits. En av egenskapene til bussen er at den tilfører differensiell overføring av data som øker dens resistans mot støy og videre senker støy ut av systemet (Analog Devices 2008). Det betyr at spenningsnivået på en linje er lik det inverse av spenningen på den andre linjen.

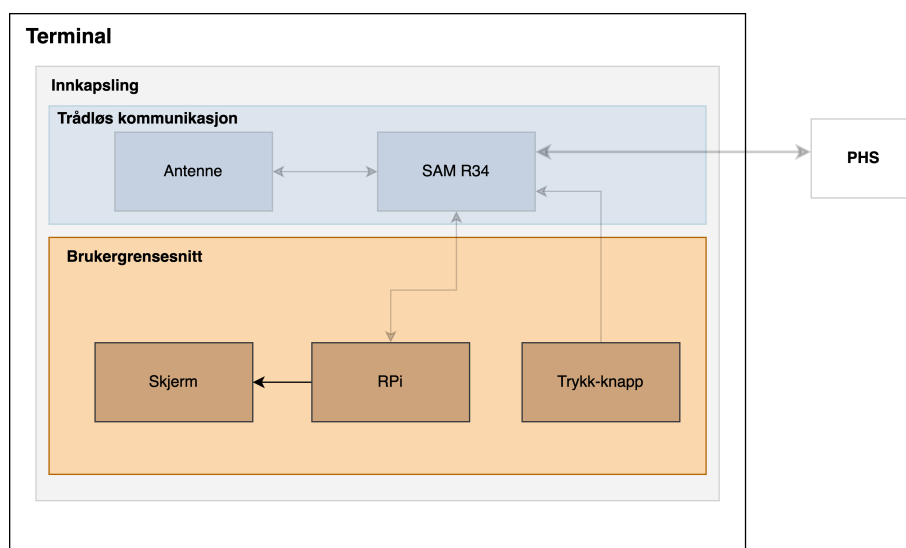
For å oppnå full-duplex dataoverføring med RS485 trengs fire buss-linjer; et par for hver retning av dataoverføring, hvor en leder av hvert par er invertert. For best resultat skal tvistede parkabler brukes (Texas Instruments 2016). Dette fordi støy fra eksterne kilder vil da induseres likt inn i begge signaltrådene, som igjen avvises av den differensielle mottakerinngangen (T. 2021). Figur 15 viser en oppkobling av fire RS485-moduler med full-duplex dataoverføring. Figuren illustrerer de fire buss-linjene som tillater toveis kommunikasjon, samt anbefalingen med tvistede parkabler.



Figur 15: RS485 med full-duplex dataoverføring mellom fire enheter (Digilent 2021).

2.5 Brukergrensesnitt

For å muliggjøre intuitiv og effektiv kommunikasjon mellom bruker og system, kreves utvikling av et brukergrensesnitt. Manuell interaksjon kan være en enkel måte for brukeren å kommunisere med systemet, mens en visuell form for kommunikasjon kan være mer hensiktsmessig for å kommunisere fra systemet til brukeren. Figur 16 fremhever terminalens komponenter som til sammen utgjør brukergrensesnittet.



Figur 16: Terminalens komponenter som til sammen utgjør systemarkitekturs brukergrensesnitt.

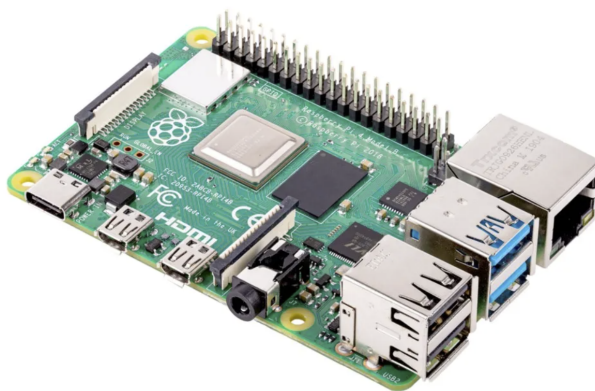
2.5.1 Trykk-knapp

For å gi brukerne av terminalen en intuitiv måte å bestille fergen på, er det integrert en trykk-knapp i hver terminal. I elektronikk refererer en trykk-knapp til en komponent som brukes til å opprette eller bryte en elektrisk forbindelse når den aktiveres.

Trykk-knapper er tilgjengelig med konfigurasjonene NO (Normally Open) og NC (Normally Closed) kontakter som endrer tilstand når trykk-knappen opereres.

2.5.2 SBC - Single Board Computer

For å muliggjøre visning av relevant informasjon om fergen på systemets skjermer, som er en integrert del av brukergrensesnittet til terminalene, kreves det en kompakt maskin med kapasitet til å fungere som både server og klient. En SBC er en kompakt integrert krets med en prosesseringsenhet, tilgang til minne og GPIO-enheter (General Purpose Input/Output - enheter). En SBC er hovedsakelig bærbar, og er et godt alternativ for ferdigmonterte enheter og prototyping. Noen av de mest kjente SBC-ene på markedet inkluderer Raspberry Pi, som vist i figur 17, Ordroid, Jetson Nano og NVIDIA XAVIER (G. R. Kanagachidambaresan 2021).



Figur 17: Raspberry Pi 4 Model B (Kjell & Company 2023).

2.5.3 JavaScript

JavaScript, heretter omtalt som JS, er et høynivå objektorientert programmeringsspråk med opphav fra C/C++. Språket ble utviklet med tanke på bruk sammen med andre web-verktøy og er ikke ment å fungere som et selvstendig språk. Hensikten med JS er å samarbeide med HTML (HyperText Markup Language) for å lage interaktive nettsider. JS er ofte brukt til å skrive klient-side applikasjoner (Brooks 2007).

2.5.4 Node.js

Node.js er en åpen kildekode-plattform som kjører JS-kode utenfor en nettleser. Den gir utviklere muligheten til å kjøre JS på serversiden, som kan brukes til å bygge skalerbare nettverksapplikasjoner og servere. Node.js bruker en enkel og effektiv asynkron modell som gjør det mulig å behandle mange parallelle forespørsler uten å blokkere tråden (Shah 2018).

2.5.5 React

React er et JS bibliotek, som kan brukes til å bygge brukergrensesnitt og utvikle webapplikasjoner. React er komponentbasert, som vil si at det er bygd opp av forskjellige gjenbrukbare byggeklosser, som settes sammen til en webapplikasjon (Bugl 2019).

React “Hooks” ble introdusert i versjon 16.8 av React-biblioteket for å gjøre det enklere å bruke avanserte funksjoner i React uten å måtte skrive komplekse klassekomponenter. Tidligere var det vanlig å bruke klassekomponenter for å håndtere tilstand og livssyklus hendelser, men med introduksjonen av hooks kan man nå enkelt legge til denne funksjonaliteten i funksjonskomponenter (Bugl 2019).

’**useState**’ er en hook i React som lar deg opprette og oppdatere tilstanden i en komponent. Denne funksjonen tar inn en initialverdi, og returnerer et par med to elementer: tilstandsvariabelen (data) og en funksjon som lar deg oppdatere tilstanden (setData).

Når tilstanden endrer seg, vil React automatisk oppdatere komponenten for å gjenspeile den nye tilstanden. Dette kan for eksempel inkludere å oppdatere innholdet på en nettside når en bruker utfører en handling, eller å endre fargen på en knapp når den klikkes på (*Using the State Hook* 2023).

’**useEffect**’ er en hook i React som lar deg utføre sideeffekter i en komponent etter at den er endret, eller når visse tilstandverdier endrer seg. Dette kan for eksempel inkludere å hente data fra en server, oppdatere dokumenttittelen, eller endre CSS-egenskaper på en HTML-element (*Using the Effect Hook* 2023).

2.6 Standarder for klassifiseringer av innkapsling for elektriske systemer

Siden terminalene er ment å bli plassert på et utendørs offentlig område, vil de være utsatt for påvirkning fra værforhold og andre eksterne faktorer. Det er derfor nødvendig å velge komponenter og innkapslinger som er robuste nok til å tåle de miljømessige betingelsene ved Ravnkloa og Vestre Kanalkai.

2.6.1 IP-rangering - Ingress Protection

IP-rangering, heretter omtalt som IP, er en internasjonal standard, utviklet av IEC (International Electrotechnical Commission), for hvor beskyttet en elektronisk innkapsling er for inntrengning av vann og faste stoffer (IEC 2022).

IP-rangeringen starter med bokstavene IP etterfulgt av 2 siffer. Det første sifferet beskriver hvor resistent enheten er for inntrengning av faste stoffer, mens det siste indikerer beskyttelsen for vann. Inntrengning av faste stoffer kan være alt fra fingre

og kroppsdelene ved lave tall, til støv for de høyeste tallene. Sifferet for faste stoffer er et tall mellom 0 og 6. Tallet for vannbeskyttelse, har en rangering fra 0 til 9. Graderingen tilsvarer varierende beskyttelse fra ingen (0), til høyt trykk og høyt tempererte vannjetter fra flere vinkler (9) (IEC 2022). Figur 18 viser en utfyllende tabell for IP-klassifisering.

First Number	Protection From Solid Objects	Second Number	Protection From Water
0	No protection	0	No protection
1	Protected from solid objects over 50mm	1	Protected from vertically falling drops of water
2	Protected from solid objects over 12mm	2	Protected from direct sprays of water up to 15 degrees from vertical
3	Protected from solid objects over 2.5mm	3	Protected from direct sprays of water up to 60 degrees from vertical
4	Protected from solid objects over 1mm	4	Protected against sprays from all directions - limited ingress permitted
5	Protected from dust-limited ingress	5	Protected from low pressure jets of water from all directions - limited ingress permitted
6	Totally protected from dust	6	Protected against strong jets of water - limited ingress permitted
NA		7	Protected from the effects of temporary immersion between 15cm and 1m
NA		8	Protected against long periods of immersion under pressure

Figur 18: Rangeringstabell for IP-klassifisering (BUD Industries Inc. 2023).

2.6.2 IK-rangering - Impact Protection

IK-rangering, heretter omtalt som IK, er en internasjonal standard for klassifisering av hvor robust en enhet er for mekanisk påvirkning. Standard graderingen EN 62262 for IK, vurderes fra IK00-IK11 (Interelectronix 2022). IK00 tilsvarer ingen beskyttelse, mens IK11 tilsvarer en beskyttelsesgrad mot ytre mekaniske påvirkninger med en energi på 50 joule.

3 Metode

Ved prosjektstart av bacheloroppgaven oppstod det behov for å utvikle en arbeidsplan og tilnærming som ville sikre en tilfredsstillende løsning av oppgaven og leveranse av det ønskede produktet til arbeidsgiver. Dette kapitlet beskriver fremgangsmåten og arbeidet fra start til slutt, og inkluderer diskusjon av utfordringer gruppen støtet på, samt vurderinger og beslutninger som førte til det endelige sluttproduktet.

3.1 Prosjektoppgavens utgangspunkt

Et av de første elementene som ble gjennomført var en fullstendig oversikt over oppgaven og funksjonsbeskrivelsen. Kartlegging av systemets utseende, hvilke enheter skulle gjennomføre hvilke funksjoner og den totale program- og systemflyten ble utarbeidet.

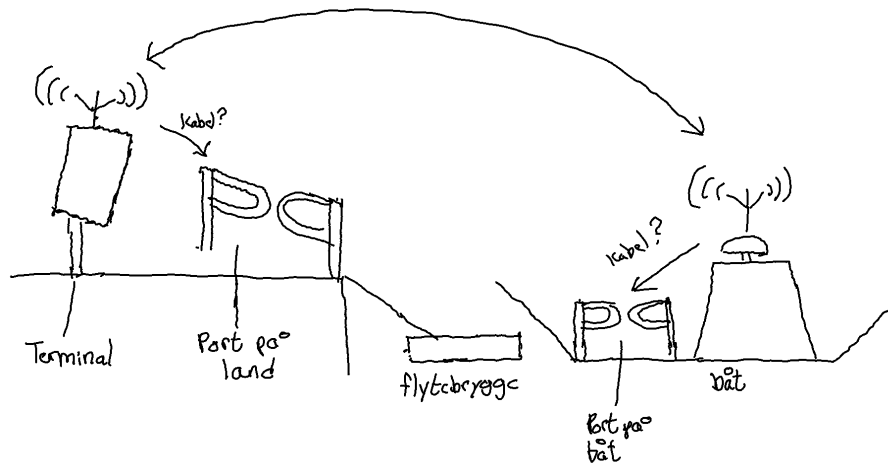
Utarbeidelse av blokkdiagram, tegninger og funksjonsbeskrivelse var et nyttig verktøy for å se oppgaven i sin helhet, samtidig som det gjorde det enklere å bryte prosjektet ned i mindre deler. Nedbrytelsen av prosjektet gjorde det oversiktlig for gruppen hvilke deloppgaver som måtte arbeides med for å fullføre det totale systemet. Alle delene måtte fungere hver for seg, samtidig som alt måtte fungere sammen.

Gruppen tok utgangspunkt i oppgave- og funksjonsbeskrivelsen presentert i seksjon 1.2, for å sette opp skisse og plan for gjennomføring. De spesifikke delsystemene og enhetene som utgjorde bakgrunn for de sentrale arbeidsoppgavene ble satt til å være:

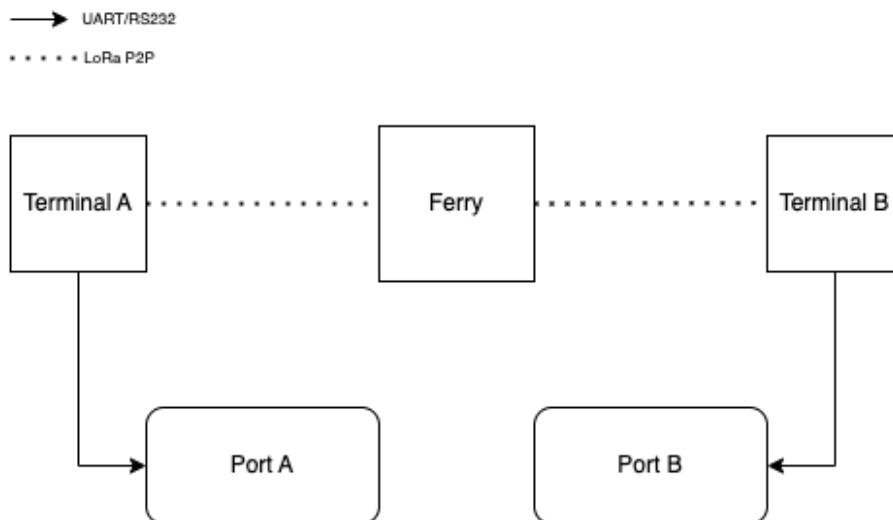
- Trådløs kommunikasjon.
- Seriell kommunikasjon.
- Vann- og vandalsikker fysisk terminal.
- Hardware og elektronikk.
- Menneskelig interaksjon og brukervennlighet.

3.1.1 Skisser og funksjonsbeskrivelse

Skissene og flytdiagrammene, som illustrert i figur 19, 20, 21 og 22, ble konstruert ved starten av prosjektet og var ment som et utgangspunkt og en pekepinne på hvordan det endelige system skulle se ut. Skjemaene fungerte som utgangspunkt for arbeid med program og hvordan bestillinger skulle gjennomføres. Det har gjennom prosjektarbeidet blitt gjort endringer utfra de originale ideene for å oppfylle krav og levere et best mulig sluttprodukt.

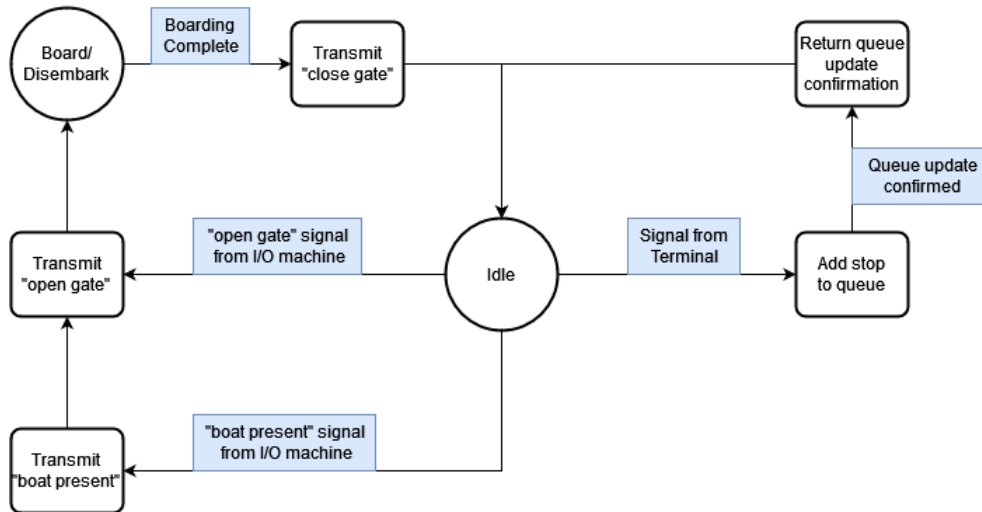


Figur 19: Tidlig illustrasjon av tenkt system.



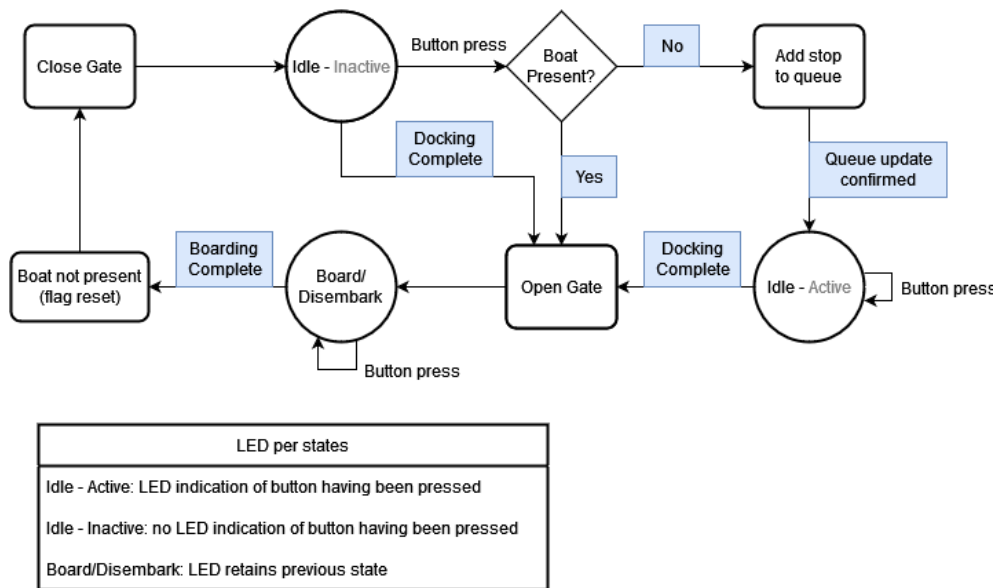
Figur 20: Tidlig illustrert kommunikasjon mellom fergen og terminalene, og mellom terminalene og portene på land.

Modem Flowchart



Figur 21: Modemets hovedfunksjoner og kriterier for å endre tilstand slik det forelå ved prosjektstart.

Terminal Flowchart



Figur 22: Terminalenes hovedfunksjoner og kriterier for å endre tilstand slik det forelå ved prosjektstart.

3.2 Valg av trådløs kommunikasjon

3.2.1 Valg av protokoll

Startfasen av prosjektet besto mye av å kartlegge informasjon rundt trådløs kommunikasjon og protokoller for å velge protokollen som var best egnet for oppgaven.

Det viktigste kravet var at protokollen måtte gjennomføre trådløs kommunikasjon på 150 meter, ettersom det er avstanden mellom Ravnkloa og Vestre Kanalkai. Figur 23 viser at det er 105 meter mellom Ravnkloa og bryggen fergen skal legge til ved. Det er derimot litt usikkerhet rundt akkurat hvor terminalene skal stå plassert. 150 meter brukes derfor som utgangspunkt gjennom prosjektoppgaven og utvikling av systemet.

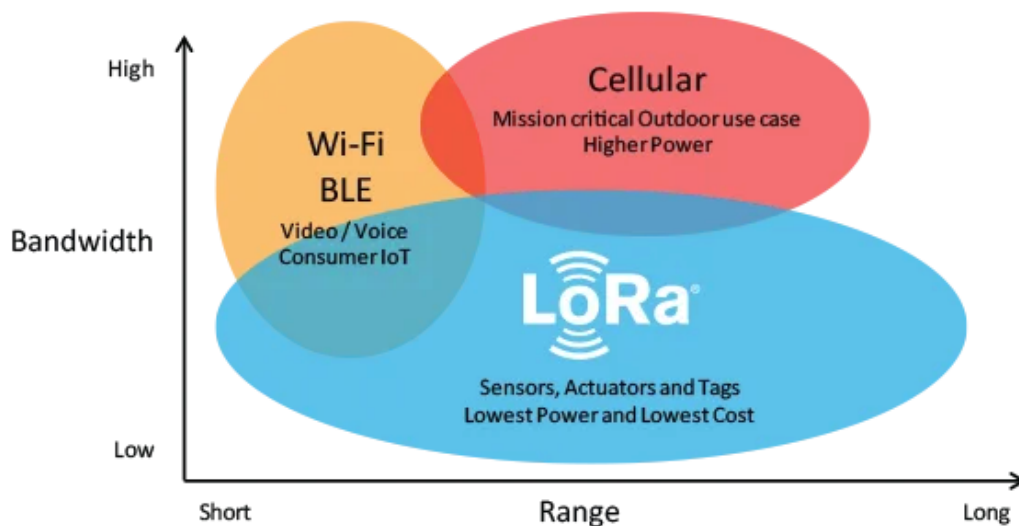


Figur 23: Tilnærmet avstand i luftlinje mellom terminalene på Ravnkloa og Vestre Kanalkai (Finn kart 2023).

Selv om avstanden maksimalt skulle være 150 meter, vurderte gruppen det som fordelaktig at protokollen hadde en sikkerhetsmargin, slik at 150 meter ikke var absolutt maksimal rekkevidde protokollen kunne overføre data. Gruppen undersøkte i hovedsak fire protokoller og koblet de opp mot nødvendig rekkevidde. Det var

Bluetooth, WiFi, cellular IoT (Internett of Things) og LoRa, hvor figur 24 illustrerer en sammenligning mellom noen av protokollene i forhold til båndbredde og rekkevidde. Det ble tidlig klart at WiFi ikke ville tilby god nok rekkevidde ettersom den ikke når over 100 meter utendørs (Frenzel Jr 2023, s. 801).

Resterende protokoller oppnår alle 150 meter. Av den grunn ble skalerbarheten til protokollen vurdert for å avgjøre hvilken protokoll som skulle benyttes i oppgaven. Hvis arbeidsgiver i fremtiden har lyst til å utvikle og skalere prosjektet ut i et større system, med flere terminaler og større avstander for trådløs kommunikasjon, er det gunstig at gruppen legger til rette for det allerede nå.



Figur 24: Sammenligning av protokoller i forhold til båndbredde og rekkevidde (The Things Network 2023b).

Bluetooth har varierende rekkevidde basert på hvilken versjon og hvorvidt det er fritt rom mellom nodene. Den nyeste versjonen, Bluetooth 5, kan utendørs i fritt rom, oppnå rekkevidde på omtrent 1000 meter, men 400 meter oppgis som mer realistisk (Sponås 2023). Det tilsier at protokollen har god sikkerhetsmargin, men skalerbarheten er begrenset hvor 400 meter ikke tilbyr stor utvidelse. Det er basert på at Bluetooth 5 kun gir en økning på 250 meter fra avstanden mellom Ravnkloa og Vestre Kanalkai.

Cellular IoT og LoRa har begge fordel ved at de oppnår kommunikasjon over store avstander. Cellular IoT bruker det allerede eksisterende mobilnettverket, for eksempel 4G og 5G, til å drive kommunikasjon mellom noder (Tata Communications 2022). Det betyr at det allerede foreligger en betydelig infrastruktur i samfunnet for bruk av cellular IoT.

MilliAmpere2 benytter allerede 4G, som eliminerer behovet for å sette opp et eget 4G-aksesspunkt. Derimot vil det medføre sikkerhetsproblemer hvis man kobler seg på 4G-nettverket til fergen. Autonomisystemet til fergen er tilkoblet 4G. Det krever derfor omfattende programvare for å forhindre at personer skal hacke seg inn og overta styringen av fergen. En slik sikkerhetsprogramvare ligger utenfor gruppens

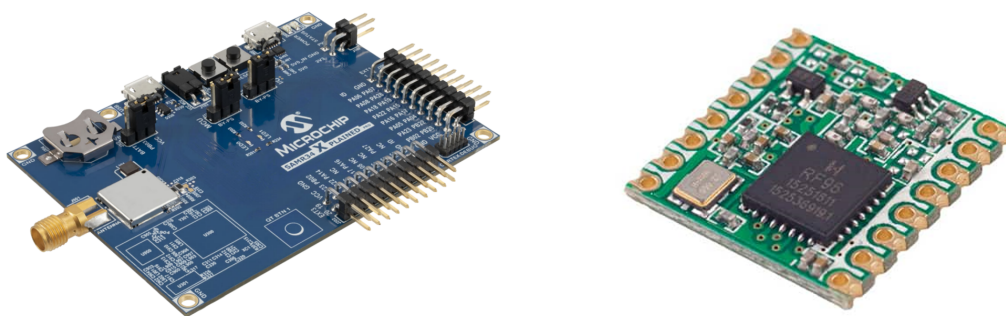
kompetanseområde og det ville vært behov for assistanse.

LoRa på sin side tilbyr allerede forhåndprogrammert sikkerhet i kommunikasjonen for å forhindre at uønskede parter skal koble seg på nettverket (Microchip Technology 2021b). LoRa benytter seg ikke av 4G-nettverket og det brukes ikke av andre systemer på fergen. Protokollen vil derfor ikke spille inn på systemene eller sikkerheten til fergen og personene om bord. I tillegg er LoRa fordelaktig ettersom protokollen tilbyr P2P funksjonalitet. Kommunikasjonen i prosjektet foregår over en kort avstand på maksimalt 150 meter, og av den grunn vil det være unødvendig å gå via et aksesspunkt. Det er enklere og mer pålitelig at det er direkte kommunikasjon mellom nodene i systemet. LoRa har også betydelig rekkevidde hvor protokollen kan kommunisere opp mot 15 km (Frenzel Jr 2023, s. 821).

Med bakgrunn i vurderingene bestemte gruppen seg for å benytte LoRa som trådløs kommunikasjonsprotokoll. Det ble sett på som hensiktsmessig å benytte en protokoll som ikke berører autonomisystemet til fergen, samtidig som det foreligger direkte kommunikasjon mellom nodene og god mulighet for skalering av systemet basert på rekkevidden til LoRa.

3.2.2 Valg av av utviklingskort

Etter at gruppen var kommet til enighet om hvilken protokoll som skulle benyttes i oppgaven, ble det sett på hvordan man skulle sette opp en prototype. Det vil si hvilke produsenter og leverandører som tilbyr krets- og utviklingskort som støtter LoRa. Et utviklingskort er et kort bestående av krets og komponenter for å eksperimentere, evaluere og utvikle program for en bestemt mikrokontroller (Elprocus 2023). Gruppen fant to alternativer, SAM R34 fra Microchip Technology og RFM95 fra Seeed Technology, som vises i figur 25.



Figur 25: SAM R34 XPLAINED Evaluation Board til venstre (Microchip Technology 2020b) og RFM95 til høyre (Hoperf 2006).

Det oppsto uenighet i forhold til hvilken modul som skulle velges for å utvikle systemet. Teknologiene er i utgangspunktet helt like ettersom begge er bygget på

Semtech sin LoRa teknologi (Semtech 2023a). Hovedforskjellen mellom kretskortene er hvordan de kobles opp og programmeres. SAM R34 er basert på mikrokontrolleren SAM L21 fra Microchip Technology og har i tillegg funksjonalitet for LoRa (Microchip Technology 2019). Det gjør at kortet kan programmeres og utvikles uten behov for en ekstern mikrokontroller. RFM95 har kun modulen for LoRa integrert på kortet og krever derfor en ekstern mikrokontroller for å programmeres. I tillegg var det en stor prisforskjell på kortene. SAM R34 kostet omtrent 1300.00 NOK per enhet, mens RFM95 lå på 85.00 NOK per enhet. Etersom RFM95 krever en ekstern mikrokontroller, ville den totale prisen være omtrent 200.00 NOK per enhet.

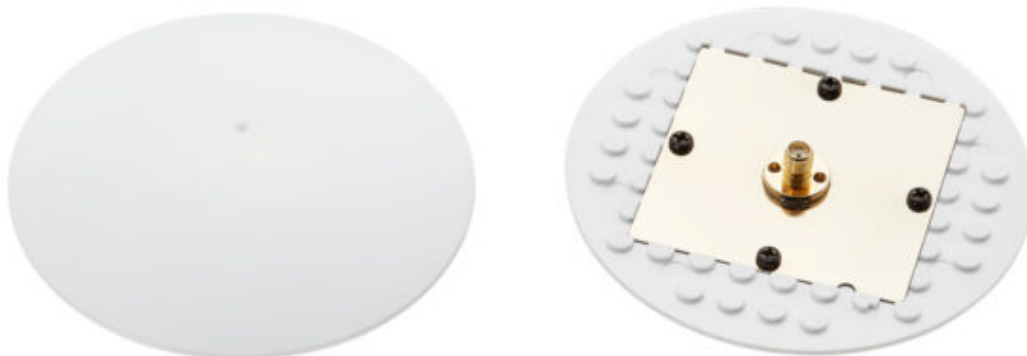
Diskusjonen omhandlet hvilket alternativ som ville generere det beste resultatet i oppsettet av en prototype. Halvparten av gruppen hevdet at det ville være enklere å konstruere en prototype med SAM R34, mens den andre halvparten hevdet at resultatene ville være likegyldige. Et av argumentene som også sto sentralt for Microchip var at det var bedre dokumentasjon, både i forhold til programmering og kretskortutlegg. I starten var det planlagt å designe kretskort og derfor var det viktig med god dokumentasjon i forhold til design og utlegg.

Alternativene ble presentert til veileder hvor veileder påpekte og la vekt på at Microchip hadde større sikkerhet, siden den benytter seg av en nøkkel for kryptering (Microchip Technology 2021b). Gruppen ble også gjort oppmerksom på at det var unødvendig å designe eget kretskort ettersom det hadde blitt tidkrevende. Det ville være lettere å gå til innkjøp av ferdig elektronikk. Av den grunn vurderte gruppen det dit at alternativet som ville gi lettest implementasjon i fergen og terminalen var SAM R34, hvor hele systemet kunne programmeres på kortet og bruke kortet som modem og noder i systemet.

SAM R34 EXPLAINED Evaluation Kit gir tilgang til funksjonaliteten og teknologien til chippen SAM R34. Kortet støtter LoRa kommunikasjon med frekvenser mellom 862 - 1020 MHz, samtidig som den har funksjonalitet for seriell kommunikasjon (Microchip Technology 2020b). Spesifikasjoner knyttet til strømforsyning kan leses i seksjon 3.5 og utviklingskortets funksjonalitet vurderes i seksjon 3.3.1.

3.2.3 Valg av antenner

Valget av antenne hadde bakgrunn i hvilke som støtter LoRa og frekvens på 868 MHz. I Europa benyttes 868 MHz når man operer i “sub-gigahertz” spekteret (Meanev 2023). Det foreligger to forskjellige antenner i systemet, en for modemmet på fergen og en antenne som brukes i terminalene. Antennene på terminalene er valgt ut i fra at de skal være vandalsikre og ikke invitere til hærverk. For eksempel hvis antennen hadde stukket ut av terminalen, hadde den vært lett å ødelegge for personer som går forbi. Av den grunn vurderte gruppen antenner som fremstår mer nøytrale og diskre når de monteres på terminalen, i tillegg til at de må ha kvalifisert IP-rangering og være vandalsikre, for å tåle det miljøet de skal operere i.



Figur 26: Roadmarker-antenne hvor man ser koaksialkonnektoren under antennen på høyre side (TAOGLAS 2023).

Det første forslaget som forelå var å bruke “Roadmarker RI.08.A.0121”. Fordelen med antennen er at den er omtrent helt flat og det vil derfor ikke være innlysende at det er en antenne når den monteres på terminalen. Den støttet ønsket frekvens og protokoll, i tillegg til å være IK10 rangert (TAOGLAS 2023). Svakheten til antennen er, som man kan se av figur 26, at den ikke har en pålitelig måte å festes, og det må benyttes teip eller lignende for å holde den fast. Det medfører en lite optimal løsning på en terminal som skal stå ute i offentligheten, ettersom personer lett vil kunne rive av antennen.



Figur 27: Antenne på AtB sine terminaler.

Transportselskapet AtB har terminaler stående rundt i Trondheim, på baksiden av disse finnes en antenne som vist i figur 27. Gruppen undersøkte om det var mulig å finne lignende antenner som oppfylte nødvendige krav, som også kunne monteres mer praktisk. Resultatet ble puck-antennen ANT-8/9-SPS1 fra Linx, vist til venstre i figur 28. Den oppfylte krav til ønsket frekvens og støtte for LoRa-protokollen (Linx Technologies 2021). Med en høyde på 23.3 mm vil enheten stikke noe ut av terminalen. Det ble ikke sett på som et stort problem ettersom den totalt sett er liten og tilsynelatende nøytral.

Oppdragsgiver Egil Eide hjalp gruppen med å finne en antenne som kunne benyttes ombord på fergen og hvordan den skulle monteres. Gruppen så ingen problemer med forslaget til Egil og bestemte seg for å bruke foreslått antenne, Delta 5A, en fleksibel “whip” antenne, vist til høre i figur 28, ettersom den støtter protokoll, frekvens og EU-kriterier (Siretta 2023). Antennen er rett, men siden oppdragsgiver foreslo å montere den i masten på fergen, så gruppen på det som usannsynlig at personer ville klatre opp for å vandalisere den.



Figur 28: Antennener brukt i prosjektet. ANT-8/9-SPS1 til venstre (Linx Technologies 2021) og Delta 5A til høyre (Siretta 2023).

Mer informasjon om antennenenes spesifikasjoner som for eksempel vinning og strålingsmønster kan leses i seksjon 3.8.

3.3 Utvikling av programvare

3.3.1 Programvare tilknyttet LoRa- og SAM R34-funksjonalitet

Microchip har utviklet en manual tilknyttet teori og hvordan man skal sette opp LoRa P2P med SAM R34 (Microchip Technology 2021a). Manualen viser til en eksempelkode som er tilgjengelig å laste ned. Utviklingen av programvare tilknyttet trådløs kommunikasjon baserer seg mye på denne eksempelkoden, ettersom den tilbyr mange nyttige funksjoner og oppsett av programflyt.

Utvikling av programvare foregikk i Microchip Studio, som er Microchip Technology sitt eget utviklingsverktøy. Verktøyet inneholder “data visualizer” hvor man kan se signaler inn og ut av kortet, i tillegg til å måle strømforbruk (Microchip Technology 2022).

Da utviklingskortene ankom, gikk det mye tid på å sette seg inn i eksempelkoden for å forstå de ulike funksjonene og programflyten. Det ble kartlagt hvilke funksjoner som var nødvendige for å etablere et nettverk og sende data mellom noder.

Basert på eksempelkoden og dens funksjonalitet utviklet gruppen egne funksjoner og programflyt, slik at programmet fulgte ønsket funksjonalitet til det overordnede systemet slik det forelå ved prosjektstart, som vist i figur 21 og 22. Programmene for ferge og terminal skilles fra hverandre ved at det er forskjell i hvilke signaler som mottas trådløst og serielt. Det ble videre tenkt at modemmet på fergen skulle være

et FFD (Full Functional Device), som beskrevet i seksjon 3.3.1. Det innebærer at modemet etablerer og administrerer nettverket der terminalene fungerer som noder og enkelt kan koble seg til.

I henhold til dokumentasjonen fra Microchip (Microchip Technology 2021b), har utviklingskortene evnen til å sende trådløse meldinger enten ved å “broadcaste” eller “unicaste” signalene. Når en melding broadcastes, har den ingen bestemt mottaker, og alle enheter som er koblet til nettverket vil motta meldingen. En melding som unicastes, sendes direkte til en bestemt mottaker. Ved vurderingen av om nodene i nettverket skulle broadcaste eller unicast meldinger, ble det lagt vekt på at terminalene hele tiden må få informasjon om fergens status for å kunne gi brukerne av terminalen tilbakemeldinger om systemet. Derfor bestemte gruppen at meldingene skal broadcastes, slik at alle nodene i nettverket mottar dem. Det vil gi nodene i systemet en kontinuerlig oversikt over hendelser og fergens status.

Det foreligger EU-direktiv som bestemmer hvor lenge en sender kan okkupere en kanal, altså sende et signal, i løpet av en time. Reguleringen presiserer at ved 868 MHz kan ikke en sender okkupere kanalen i mer enn 1% av en time (Microchip Technology 2021c). Det tilsier at det måtte kartlegges hvor lang tid senderen bruker på å sende data og dermed undersøke hvor mange signaler som kan sendes. Eksempelkoden medfølger funksjonen `calculate_ToA()` som regner ut hvor lang tid en sender bruker på å sende et gitt antall bytes (Microchip Technology 2021b).

Det ble bestemt at gruppen skulle sende 1 byte for å representere handlinger. Funksjonen “calculate_ToA” oppga at senderen brukte 25 ms på å sende 1 byte, se figur 29. 1 time = 3600 s. En prosent av 3600 s er 36 s, som tilsvarer 36 000 ms. Antall signaler som kan sendes per time er da $\frac{36000 \text{ ms}}{25 \text{ ms}} = 1440$.

```
My Address: 0x00042519238b015e PANID: 0x1234 Channel: 5

Connection      PeerLongAddress      PeerInfo      RxOnStatus
00              000425191801e599    01            01
BTime-on air calculated : 25

Data Confirm: Handle: 0 status:0
BTime-on air calculated : 25

Data Confirm: Handle: 1 status:0
```

Figur 29: Tiden sender bruker på å sende 1 byte ved å benytte `calculate_ToA`.

Det betyr at man kan sende data totalt 1440 ganger i løpet av en time. Gruppen så på det som usannsynlig at det ville sendes mer enn 1440 signaler per time i bestilling- og operasjonssystemet med milliAmpere2, og systemet overholdt av den grunn EU-direktivene.

SAM R34 støtter sovefunksjonalitet som tillater prosessoren å gå i sovemodus, og det forelå vurderinger rundt hvorvidt det skulle implementeres i bestillingssystemet

for å redusere strømbruken. Sovemodus betyr at prosessoren slår seg tilnærmet helt av og blir inaktiv, hvor kun funksjonalitet for å vekke prosessoren kjører (Microchip Technology 2021b). Et P2P-nettverk krever at en av nodene er et FFD, fordi det konstant må være en aktiv enhet som opprettholder nettverket og tilkoblingen mellom nodene (Microchip Technology 2021c). Gruppen tok utgangspunkt i at modemmet ble forsynt med batteristrøm fra fergen og optimalt sett burde den sove for å bruke minst mulig batterikapasitet. Derimot skal modemmet i fergen behandle og videresende mye informasjon og det vil ikke være hensiktsmessig at systemet må vente til den våkner for å kunne sende informasjon. Det ble derfor bestemt at modemmet i fergen skulle være FFD. Det medfører at den konstant trekker strøm fra batteriet. På den annen side er strømtrekket lavt med et gjennomsnitt på 16.8 mA, som illustrert i figur 30.



Figur 30: Strømforbruk ved vanlig operasjon. x-aksen beskriver tid i sekunder fra målingene startet, y-aksen er strømtrekket til modemmet gitt i milliampere.

Terminalene på sin side vil være koblet til strømmettet på land, og batterisparing er derfor ikke relevant. Det kan argumenteres for at inaktivt modus vil føre til totalt sett mindre strømforbruk fra strømmettet, men det vil være en såpass liten andel at det vil føre til større utfordringer med bestillingssystemet enn nytten det vil ha på strømforbruket.

Et annet moment som ville gjort det omfattende å implementere sovmodus er at mikrokontrolleren kun kan motta og sende trådløs data når den er aktiv. Hvis en enhet prøver å sende data til en inaktiv enhet, må den vente til enheten er aktiv og etterspørre data på ny (Microchip Technology 2021b). Dersom det foretas en bestilling, men terminalen er inaktiv, vil bestilleren måtte vente til terminalen er aktiv igjen. Dette kan føre til en betydelig grad av usikkerhet for brukeren av systemet ettersom det ikke er garanti for at bestillingen er registrert.

Det ble også undersøkt om det var mulig å vekke mikrokontrolleren med et avbrudd. Det ville fungert ved for eksempel en bestilling, knappetrykk, på terminalen ettersom det vil sende et fysisk singal som kan vekke terminalen. Mikrokontrolleren kan

derimot ikke bli vekket med et radiosignal, ettersom funksjonaliteten for å motta signaler ikke er operativ. På bakgrunn av at oppgaven går ut på å etablere et bestillingssystem, setter det krav til at det tar kort tid fra en bestilling, til brukeren gjøres oppmerksom på hva som skjer. Det ble av den grunn valgt at både modem og terminalene skal være aktive til enhver tid, slik at systemet blir pålitelig og brukervennlig.

“Network freezer” eller nettverkfryser er en funksjonalitet ved SAM R34 som lagrer nettverksinformasjon i et NVM (Non-Volatile Memory) (Microchip Technology 2021b). Data lagret i et NVM vil ikke slettes dersom strømmen til en mikrokontroller kuttes (M. A. Mazidi, J. G. Mazidi og McKinlay 2006, s. 418). Nettverkfryser lagrer informasjonen for nettverket mellom fergen og terminalene ved første oppkobling. Mister systemet strømtilførsel vil nettverkinformasjonen være lagret og kan hentes ut når systemet kobles til strøm igjen. Enhetene trenger av den grunn ikke å bruke lang tid på å etablere et nytt nettverk mellom nodene, men setter opp det forrige nettverket med en gang, som sparer energi og tid (Microchip Technology 2021b).

3.3.2 API for kommunikasjon i systemet

For å oppnå kommunikasjon mellom de ulike systemene på tvers av bachelorgruppene, ble det utviklet et API som definerer en protokoll for slik kommunikasjon. API-et fungerer som en bro mellom de ulike systemene og gjør det mulig for dem å utveksle data. API-et benyttes videre internt i kommunikasjonen i LoRa-nettverket.

Med bakgrunn i at alle nodene i LoRa-nettverket broadcaster meldinger på PAN, ble meldingsstrukturen i API-et definert til å ha en mottaker-ID, avsender-ID og en kommando, totalt er meldingen på 8 bit. Dette ble gjort for å forsikre at nodene i nettverket til en hver tid kunne se hvilke meldinger som var ment for hvilken node, samtidig som de forble synkronisert ved at hver terminal ville kjenne til fergens status. API-et ble utviklet med tanke på at hver node i nettverket ville forstå hvilken datapakke som var ment for hvilken node basert på datapakkens 2 MSB (Most Significant Bit), som representerer en mottaker-ID. Bit 5 og 4 i datapakken representerer ID til avsender av meldingen og de resterende 4 LSB (Least Significant Bit), representerer kommando. Vedlegg A viser en komplett oversikt over API-et samt en beskrivelse av hver kommando.

Generelt følger API-et nevnt struktur, med unntak av informasjon som er ment for alle nodene i nettverket. Et eksempel på det vil være siste kommando, PPLCOUNT, fra figur 31 som indikerer passasjerantallet på fergen. Her vil avsender være irrelevant på bakgrunn av at signalet ikke skal utføre en operasjon annet enn å informere ferge-modem om passasjerantall. For å videre bygge opp under valget så vil fergens lokasjon allerede være kjent basert på foregående signaler som er utvekslet og det er derfor ikke nødvendig å kjenne til avsender av PPLCOUNT-kommando.

For at API-et skal fungere som tiltenkt må hver node i nettverket ha en unik ID bestående av to bit for å kunne kontrollere om datapakken mottatt er ment for den aktuelle noden. Dersom den mottatte datapakkens 2 MSB samsvarer med nodens

identifikator, gjennomfører noden i nettverket den aktuelle kommandoen bestemt av de 4 LSB i datapakken. Dersom datapakkens 2 MSB ikke samsvarer med nodens ID vil den kun oppdatere sin status på fergen og videre forkaste kommandoen. Hver node vil også legge igjen en avsender-ID ved sendt datapakke for å identifisere til resten av nettverket hvem som sendte pakken.

Command identifier summary

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:]			
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]			

Figur 31: Oversikt over alle kommandoer i API-et.

Figur 31 viser en fullstendig oversikt over de mulige kommandoene i API-et. Tabellen viser fra venstre, navn på kommando, bit-posisjonene kommandoen opptar, mottaker-ID gitt av bit 7:6, avsender-ID gitt av bit 5:4 og kommando/melding gitt av de fire LSB, bit 3:0. Fra figur 32 ser vi definisjonen av en kommando, GSTATE. Figuren viser en oversikt over kommandoens egenskap, gyldige mottaker-ID, avsender-ID og meldingsinnhold. Her vil bruker av API-et se de mulige kombinasjonene av bits som vil gi en gyldig 8-bit datapakke.

Gate States

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

Bit	7	6	5	4	3	2	1	0
	REID[7:6]		SEID[5:4]		STATE[3:0]			

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

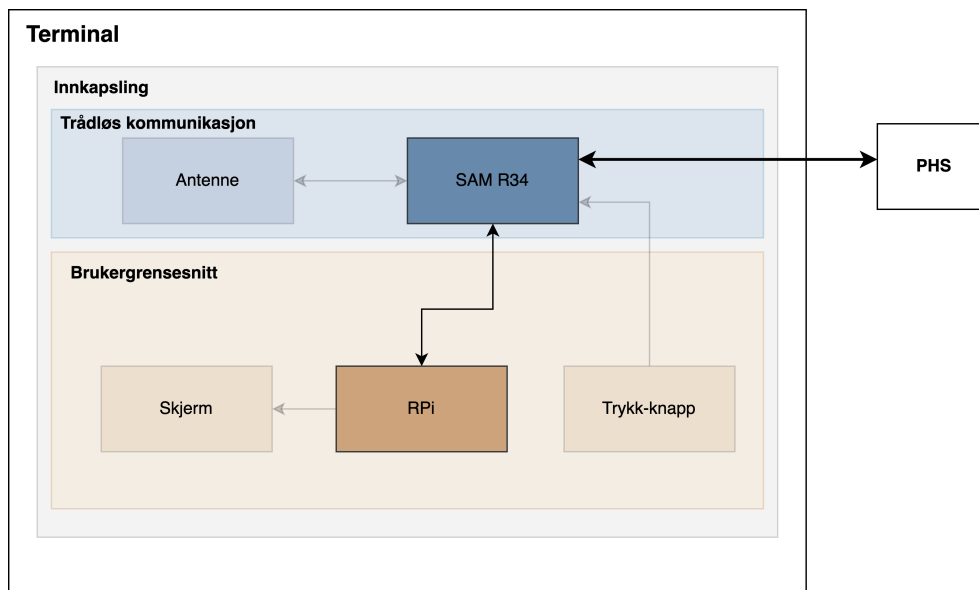
Bit 3:0 - STATE[3:0] Tilstand på port

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

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

Figur 32: Eksempel på API struktur.

3.3.3 Oppsett og konfigurering av UART



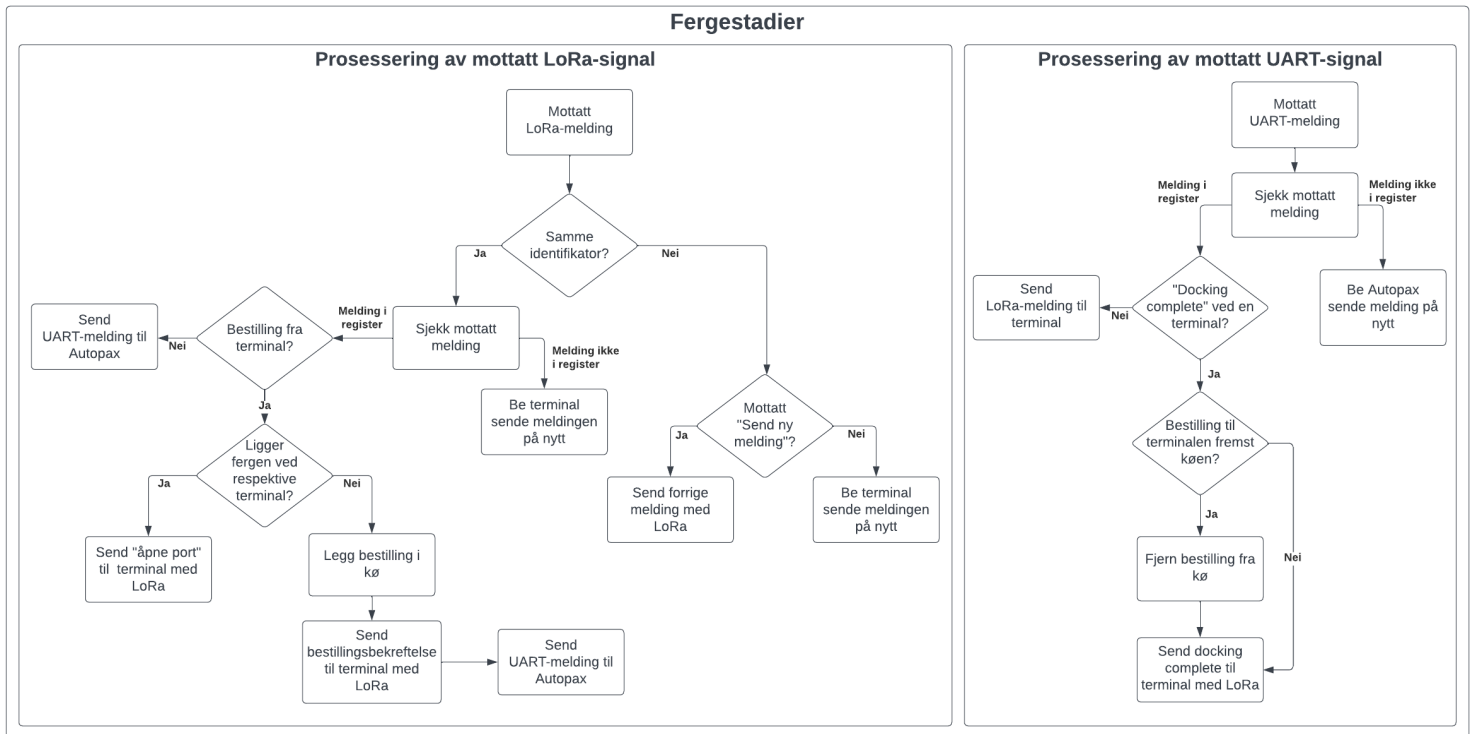
Figur 33: Systemarkitektur som fremhever seriell informasjonsflyt mellom SAM R34, RPi og PHS.

For å etablere kommunikasjon mellom SAM R34 og RPi, samt mellom SAM R34 og PHS/Autopax, som fremhevet i figur 12, ble UART-protokollen valgt. UART-protokollen ble valgt på bakgrunn av gruppens kompetanse innenfor protokollen og dens mulighet for full duplex kommunikasjon. Totalt ble det opprettet to UART-porter på SAM R34, hvor det i tillegg forelå en tredje som ble brukt som UART-meny for feilsøking. For å konfigurere UART-protokollen på RPi, henvises det til vedlegg B.

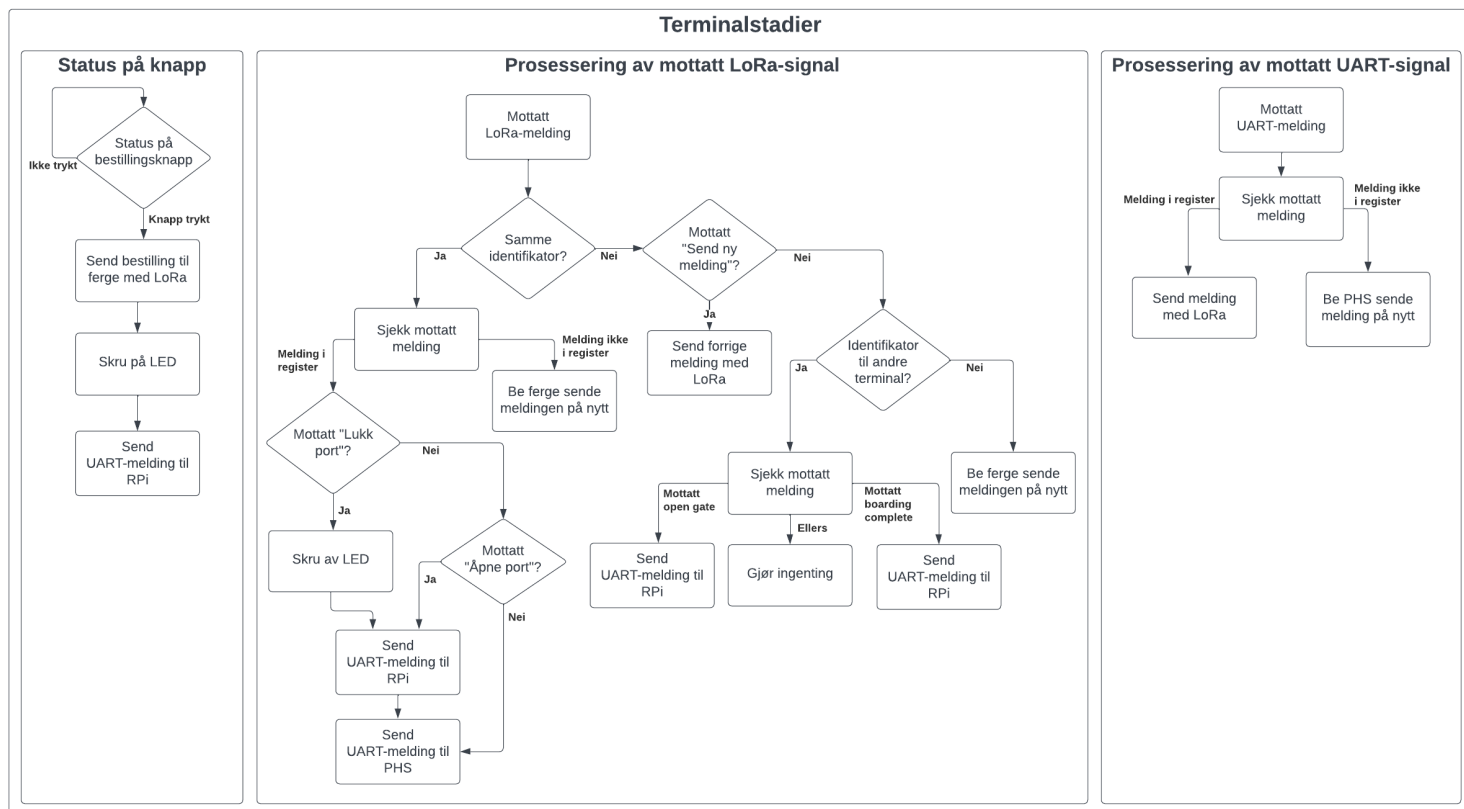
UART-protokollen på SAM R34 ble konfigurert med Microchips ASF (Advanced Software Framework), som er et rammeverk som tilbyr programvare-drivere og biblioteker for Atmel og Microchip mikrokontrollere (Atmel 2012). Baudraten ble satt til 115200, og ingen paritetsbit ble benyttet. Det ble videre benyttet én stopp-bit. Konfigurasjonen ble som følger: 115200/N/1. Det ble aktivt valgt å ikke benytte paritetsbit, ettersom programvaren skulle kontrollere alle mottatte meldinger mot et register av definerte datapakker som stemte overens med API-et beskrevet i seksjon 3.3.2.

3.3.4 Systembeskrivelse av bestillingssystem

Den utviklede koden fletter sammen LoRa og UART til et komplett kommunikasjonssystemet som tillater bestilling av ferge, kommunikasjon mellom terminalene og PHS, modemmet og Autopax, og trådløs kommunikasjon mellom terminalene og fergen.



Figur 34: Stadier fergen går gjennom ved mottak og sending av LoRa og UART signaler.



Figur 35: Stadier terminal går gjennom ved mottak og sending av LoRa og UART signaler, og behandling av knappetrykk.

Programvaren initialiserer alle nødvendige systemer og etablerer et PAN mellom fergen og terminalene. Hver node vil kontinuerlig sjekke om det er mottatt en UART- eller LoRa-ramme. Ved mottatt LoRa-ramme vil noden først kontrollere om meldingen er ment til seg ved å sammenligne identifikator-bitene i meldingen mot sin egen identifikator. Deretter kontrollerer den meldingen opp mot forhåndsdefinerte signaler i filen “customCommandsTerminalFerry.h”, som finnes i følgende github (Bachelorgruppe E2318 2023). Er meldingen en av de definerte signalene, videresender noden meldingen til riktig instans. Modemet og terminalene gjennomfører også særegne handlinger basert på noen signaler som kommer frem av figurene 34 og 35. Nodene sjekker ikke identifikator ved mottatt UART-ramme, ettersom en UART-melding i systemet kun blir sendt til en bestemt mottaker. Dersom nodene ikke gjenkjenner meldingen mottatt, forespør den avsender om å sende meldingen på nytt. Det sikrer systemet mot eventuelle bitfeil eller andre signalproblemer.

Dersom en UART-melding fra Autopax mottas av ferge-modemet med en indikasjon om at porten ved terminal A skal åpnes, vil fergen “broadcaste” meldingen ut i nettverket. Terminal A vil gjenkjenne meldingen som tiltenkt til seg, basert på identifikatoren, og åpner porten. Tilsvarende vil Terminal B også motta meldingen, men etter å ha verifisert at den ikke er tiltenkt for seg, vil den kun oppdatere sin

informasjon om at fergen befinner seg ved Terminal A.

Terminalene vil også kontinuerlig sjekke om det foreligger et knappetrykk, altså en bestilling av fergen. Terminalen leser statusen på knappen og vil sende en bestilling til fergen dersom den trykkes. Et aspekt ved bestillingsknappen var å sikre den mot uønsket menneskelig adferd, hvor et eksempel ville vært gjentatte knappetrykk. Det var ikke ønskelig at terminalen skulle sende et trådløst signal hver gang knappen ble trykt, spesielt ikke hvis den ble trykt gjentatte ganger. Dette kunne ført til et betydelig antall bestillinger og signaler som kunne tømte signal-kvoten, som forklart i seksjon 3.3.1. Av den grunn ble det lagt inn en variabel, **orderStatus**, som kontrollerer hvorvidt det allerede foreligger en bestilling eller ikke og forhindrer repeterende signaler utløst ved knappetrykk.

Dersom det foreligger en bestilling evalueres **orderStatus** til “true”. Personer kan fremdeles trykke på knappen, men terminalen foretar ingen handlinger og det vil derfor ikke bli lagt til en ny bestilling. Når portene på land ved den respektive terminalen lukkes, blir variabelen “false” og det er igjen mulig å legge inn en ny bestilling.

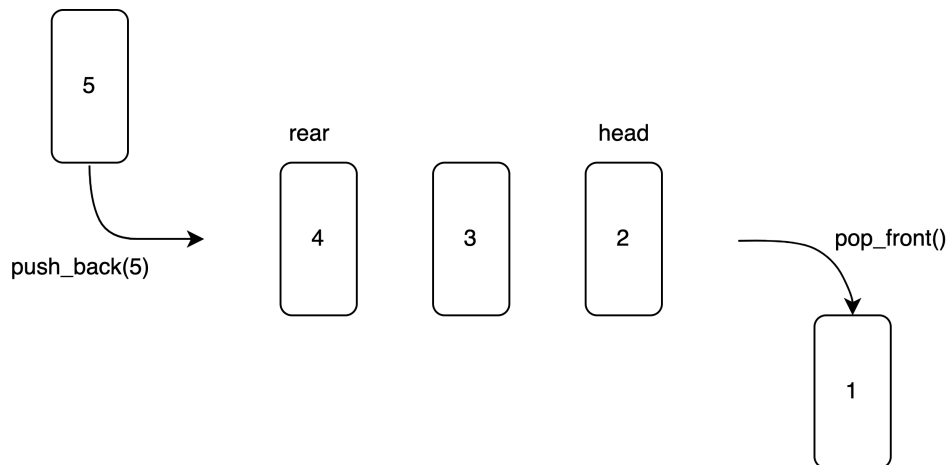
Mottar fergen en bestilling fra enten terminal A eller B, vil programmet kontrollere hvilken terminal det gjelder og hvilken status fergen har. Fergen gir beskjed om at det er mulig å gå om bord dersom fergen allerede ligger til kai ved den respektive terminalen. Hvis ikke legger fergen bestillingen i køen og gir en bestillingsbekreftelse til terminalen.

Når fergen ankommer en terminal og modemmet får beskjed om at “docking” er fullført ved terminalen, vil modemmet sjekke det første elementet i bestillingskøen. Dersom elementet tilsvarende terminalen fergen legger til kai ved, vil modemmet fjerne bestillingen fra køen. Dette sikrer at modemmet ikke sletter en bestilling fra den andre terminalen.

3.3.5 Implementering av FIFO-struktur

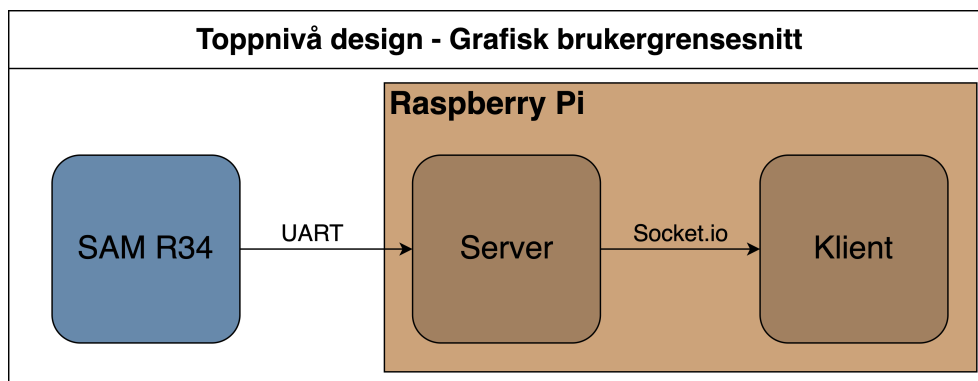
Bestillingskøen, som beskrevet i seksjon 3.3.4, er strukturert i en FIFO-struktur og er basert på prinsippene gitt i boken “Introduction to Algorithms, Third Edition” (Cormen mfl. 2009), hvor det foreligger visse modifikasjoner. Kø-strukturens medlemsvariabler, **head**, **rear** og **queue**, er strukturert i en **struct** for å gjøre koden modulerbar. Strukturen vil videre ha tilgang på en rekke funksjoner som å sjekke om køen er tom eller full samt å legge til eller fjerne elementer. Hver funksjon tar inn et instans av kø-strukturen som argument, hvor funksjonene for å legge til og fjerne elementer tar inn en peker til instansen for å kunne gjøre permanente endringer som å legge til eller fjerne elementer i køen. Funksjonen **push_back** implementerer operasjonen ”legg bakerst”. Funksjonen legger til et nytt element bakerst i køen basert på argumentet som blir sendt til funksjonens parameter. Funksjonen **pop_front** utfører en operasjon for å ta ut det første elementet i køen. Figur 36 illustrerer en vilkårlig FIFO-struktur med dens funksjoner **push_back** og **pop_front** samt medlemsvariablene **head** og **rear**.

FIFO-strukturen er definert i filen 'UpdateQueue.h' og deklartert i filen 'UpdateQueue.c', som finnes i Github-repository-et (Bachelorgruppe E2318 2023).



Figur 36: En vilkårlig FIFO-struktur som illustrerer funksjonene **pop_front** og **push_back** samt posisjonene **head** og **rear**.

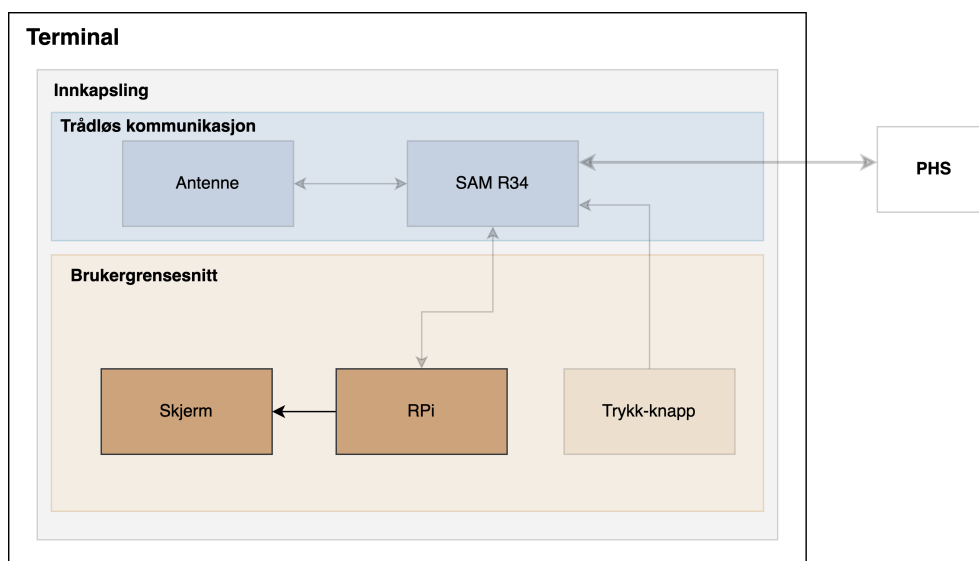
3.3.6 Grafisk brukergrensesnitt - GUI



Figur 37: Toppnivå blokkdiagram over overordnet system for GUI.

For å kunne tilby brukeren informasjon om bestillinger og status på fergen, ble det utviklet et GUI til skjermen. Figur 37 viser overordnet signalflyt for GUI, og koblingen opp mot resten av terminalsystemet. For å designe GUI ble det tidlig valgt å bruke en struktur med React som klient og Node.js som server, en løsning anbefalt av React (*Start a New React Project* 2023). React ble valgt da det er et rammeverk som gruppen allerede hadde erfaring med. I tillegg har React et stort fellesskap, med forum og mye dokumentasjon, som bidrar til enklere og mindre tidkrevende utvikling (*React Community* 2023).

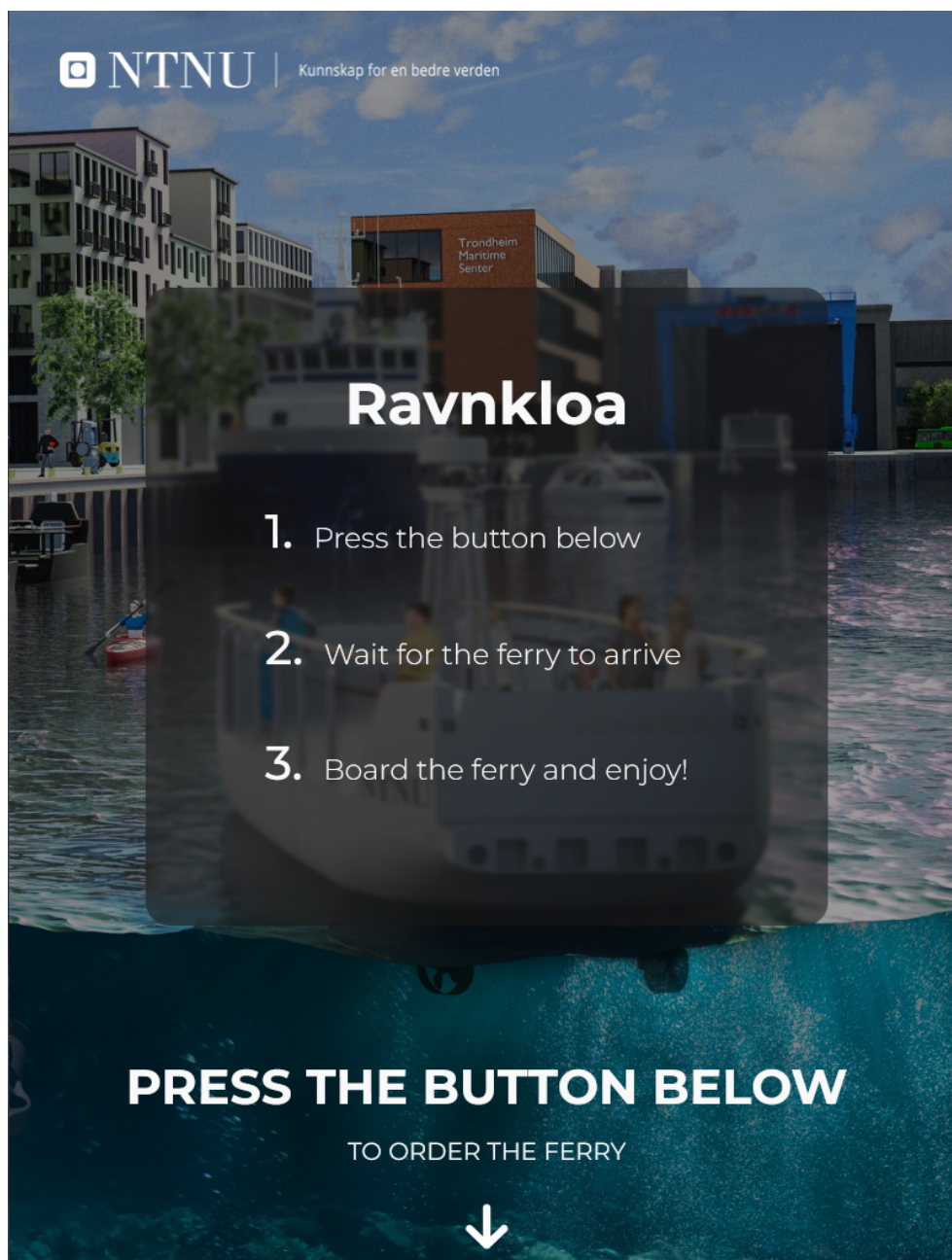
3.3.7 Nettsidedesign



Figur 38: Raspberry Pi og skjerm fremhevet i terminalens arkitektur.

Gruppen hadde begrenset erfaring innenfor grafisk design og for å imøtekomme denne utfordringen ble det inngått et samarbeid med en web-designer ved navn Anders Ziener Kristensen. Kristensen besitter betydelig ekspertise innenfor web-design, og har tidligere utført både frivillig og profesjonelt arbeid innenfor feltet (*My Digital Portofolio. By - Anders Kristensen 2023*). Gjennom et nært samarbeid ble det utviklet en rekke forslag til grafiske design ved hjelp av designprogrammet Figma, som er en programvare spesielt utviklet for grafisk design. Forslagene ble senere presentert for og godkjent av oppdragsgiver og med-veiledere. Etter dette ble designet justert, forbedret og implementert i klientkoden på Raspberry Pi. Figur 38 fremhever RPi-en og skjermens posisjon i arkitekturen.

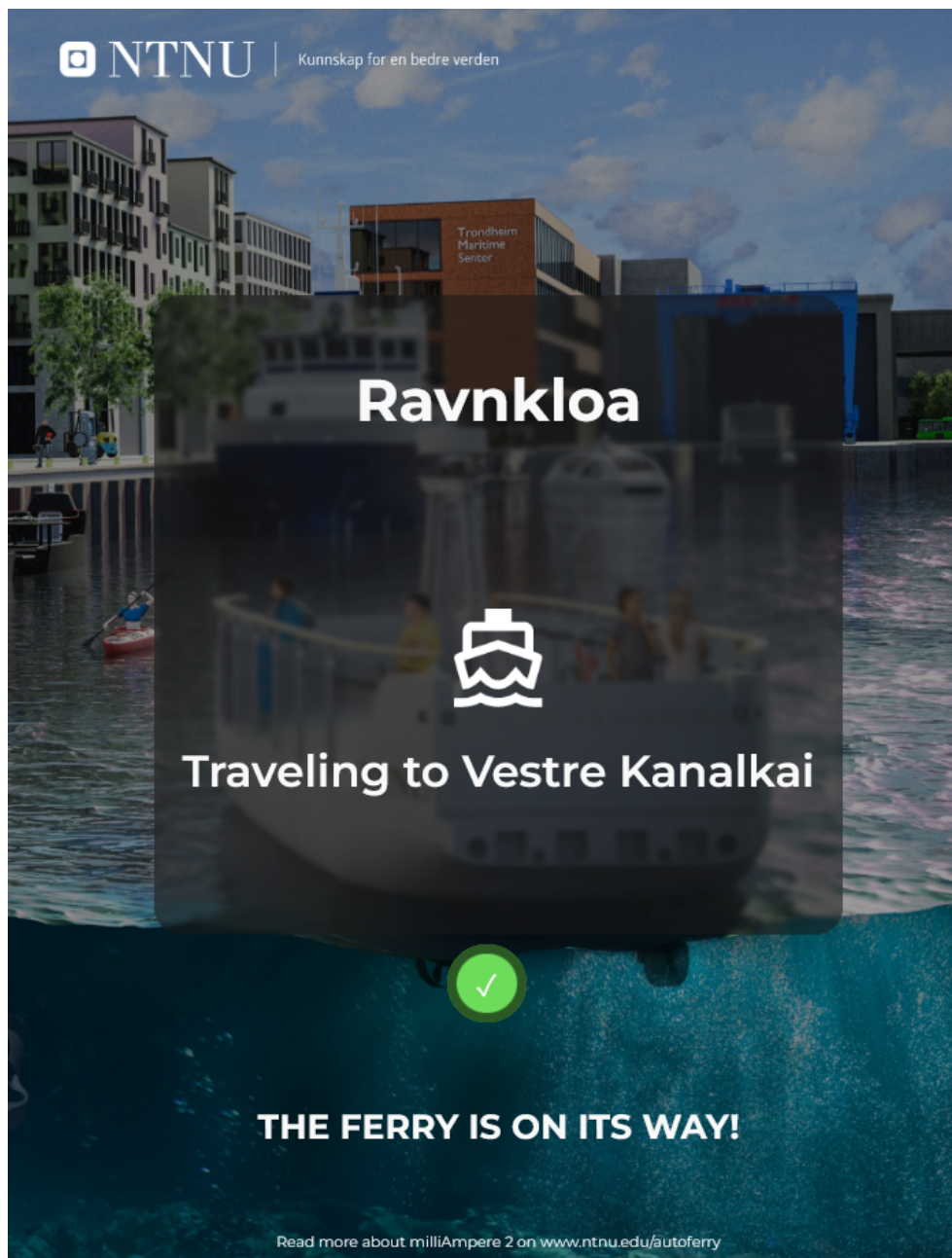
Det var viktig å gjøre interaksjonen enkel og intuitiv for å oppmuntre brukeren til å bestille fergen, spesielt for de som ikke er kjent med terminalen eller området rundt. For å oppnå dette, var det nødvendig å ha en tydelig og enkel brukeropplevelse med minimalt antall interaksjoner.



Figur 39: Grensesnitt som fungerer som terminalens forside. Bildet i bakgrunnen krediteres Mikael Sætereid (C) NTNU.

Skjermen vil alltid vise NTNU-logo i øvre venstre hjørne, på begge terminalene, som vist i figur 39. For forsiden er hovedfokuset rettet mot instruksjonen ”PRESS THE BUTTON”, som er uthevet med størst skriftstørrelse relativt til resten av teksten, og er plassert nederst på siden. Instruksjonen peker ned på terminalens trykknapp, plassert rett under skjermen, for å gi en tydelig visuell henvisning til brukeren om hvor de skal trykke. Navnet på terminalen vil også alltid være synlig, Ravnkloa eller Vestre Kanalkai, avhengig av hvilken terminal brukeren befinner seg ved. Det er viktig at brukeren enkelt kan identifisere stedet de befinner seg på, for å bedre forstå informasjonen på bestillingssiden om hvor fergen befinner seg eller er på vei. En bruker som ikke er kjent i Trondheim kan ha vanskeligheter med å identifisere at

de befinner seg på Ravnkloa, og derfor er det viktig å tydeliggjøre dette på skjermen. Under terminalens navn står en enkel og intuitiv forklaring på terminalens bruk. Her ble det bevisst gjort et valg om å begrense mengden tekst for å oppfordre til lesning.



Figur 40: Grensesnitt som vises på skjermen etter at fergen er bestilt via knappetrykk.

For å skape en klar og tydelig forskjell mellom to mulige tilfeller, ble GUI-en delt opp i to separate sider: "Frontpage" og "Orderpage". Frontpage, som vist i figur 39, skulle fungere som en innledende startside som viser brukeren en enkel beskrivelse om bruk av terminalen. Orderpage, som vist i figur 40, skulle være en bekreftelse på bestillingen og en forsikring om at brukeren nå kan forberede seg på ombordstigning. Dermed ligger fokuset på denne siden på den statiske teksten "THE FERRY IS ON

ITS WAY”. For å understreke dette finnes det et pulserende grønt avkryssningsmerke rett over den uthevede teksten. På midten av skjermen under ferge-ikonet vises fergens status som inkluderer informasjon om hvor fergen er nå eller hvor den er på vei. Denne statuteteksten endrer seg ut i fra utvalgte kommandoer som sendes i nettverket og deles med RPi, se tabell 1. Disse kommandoene styrer også bytting av sider, se figur 41.

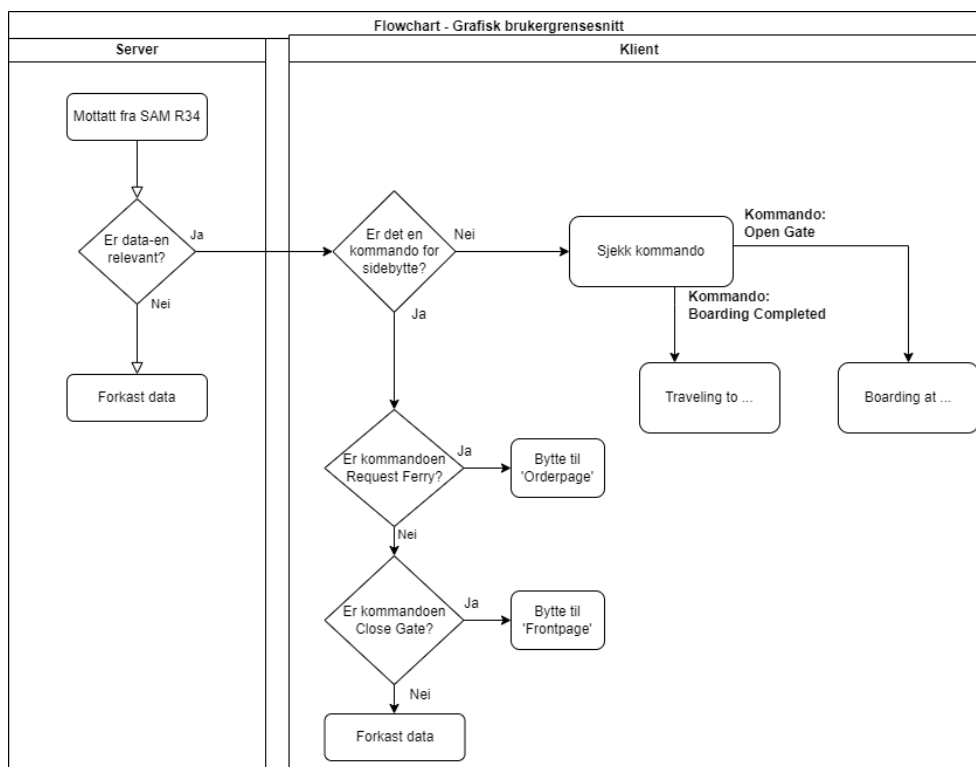
3.3.8 Server og klient

Server-siden av brukergrensesnittet er ansvarlig for å overføre informasjon mottatt fra SAM R34 via RPi-ens UART til klienten. For å oppnå dette er serveren strukturert med to moduler, hvor den ene er ansvarlig for å lese UART-rammer og den andre for å sende meldinger videre til klient-siden. For å lese av UART, ble JS-biblioteket SerialPort brukt. SerialPort-biblioteket gir Node.js tilgang til maskinvare-UART på RPi-en (Gulotta 2022). For å kunne overføre meldinger mellom server og klient, ble det først laget et system hvor serveren genererte en respons hver gang det kom et nytt UART-signal. Klienten ble da satt opp til å hente responsen ved et satt intervall på 1 sekund. Problemet med denne løsningen var at klienten måtte spørre serveren om ny respons, selv om responsen ikke hadde blitt oppdatert av en ny UART-melding. Dette medførte unødvendige antall forespørsler og responser mellom server og klient, noe som over tid kunne belastet RPi-en unødig. Et annet problem som oppstod var at responsen fra serveren ble oppdatert mellom intervallene klienten spurte på, som førte til at klienten mistet meldingen.

For å løse dette ble Socket.io tatt i bruk på server-siden. Socket.io er et JS-bibliotek som muliggjør toveis sanntidskommunikasjon mellom en server og klient (*What Socket.IO is* 2023). Dette ga muligheten til å skape et event-basert system, hvor klient-siden ikke trengte å forespørre en respons, men heller lå og lyttet på om responsen fra serveren ble oppdatert. Dette medførte at systemet ikke trengte å oppdatere kontinuerlig med et fast intervall, men heller oppdateres når det var kommet inn en ny UART-melding. Dette løste både belastningsproblemet med at RPi-en måtte oppdatere oftere enn den trengte og fjernet risiko rundt å miste meldinger.

Kommando navn:	Kommando beskrivelse:
Open gate (Ravnkloa)	Skriver: "Boarding at Ravnkloa" på skjermen
Open gate (Vestre Kanalkai)	Skriver: "Boarding at Vestre Kanalkai" på skjermen
Close gate (Ravnkloa)	Bytter til 'Frontpage' på terminal Ravnkloa
Close gate (Vestre Kanalkai)	Bytter til 'Frontpage' på terminal Vestre Kanalkai
Boarding Complete (Ravnkloa)	Skriver: "Traveling to Vestre Kanalkai" på skjermen
Boarding Complete (Vestre Kanalkai)	Skriver: "Traveling to Ravnkloa" på skjermen
Request Ferry (Ravnkloa)	Bytter til 'Orderpage' på terminal Ravnkloa
Request Ferry (Vestre Kanalkai)	Bytter til 'Orderpage' på terminal Vestre Kanalkai

Tabell 1: Beskrivelse av relevante kommandoer sendt fra SAM R34 til RPi 4B, som blir behandlet av klient slik som vist i figur 41.



Figur 41: Kodelogikk server og klient. Tabell 1 viser kommandosignalene som styrer logikken.

I seksjon 3.4.2, blir problematikken med bruken av RPi 3B+ som skjermdriver beskrevet. Før gruppen byttet over til RPi 4B som skjermdriver, ble det først prøvd å gjøre tiltak med programvaren. I tillegg til å bytte fra å hente inn meldinger ved gitte intervaller til å hente inn på event med Socket.io, ble det satt opp et filter for ønskede signaler med bruk av en switch-case struktur, som vist på serversiden i figur 41. Dette førte til at dersom det oppstod støy eller uønskede beskjeder over UART, ville disse bli forkastet på server-siden før de ble sendt videre til klient-siden. Signalet sorteres også i to eventer, som klienten lytter på. Det ene eventet er for oppdatering av 'Orderpage' og 'Frontpage', mens det andre er for oppdatering av status-teksten på skjermen.

Data som blir sendt fra server-siden blir mottatt av "useEffect-hook", som beskrevet i seksjon 2.5.5. "useEffect-hooken" blir kjørt når ny data mottas på socket.io-event. Eventet for sidebytte utløser en useEffect-hook, som omdirigerer signalet til en funksjon som bytter mellom "Frontpage" og "Orderpage", se seksjon 3.3.7. Eventet for oppdatering av status-teksten blir overført til en egen status-komponent, med en "useState-hook", se seksjon 2.5.5. I status-komponenten blir kommandosignalene for status-teksten behandlet i et "dictionary". Dictionary er en datatype i JS som lar deg lagre og hente ut data ved å bruke nøkler og verdier (*Dictionary Data Structure* 2023), og brukes til å hente ut riktig status basert på kommandoen. Denne logikken beskrives på klientsiden i figur 41.

3.3.9 Implementering og oppsett av UART på RPi 4B

En beskrivelse av RPi-oppsettet finnes i vedlegg B. RPi-ene må igjennom en rekke innstillinger for å fungere som ønsket. For å bruke UART på RPi må maskinvare-UART aktiveres, hvor Bluetooth-modul og diverse servicefiler som kan forstyrre UART, må deaktiveres. Det er også nødvendig å installere node.js for å kunne bruke ”server”. Vedlegget beskriver også trinnene for å klonе koden for både klient og server, og hvordan man setter opp servicefiler for å sørge for at systemet starter automatisk ved RPi-oppstart. Skjermen må roteres til høyre siden den er installert vertikalt på terminalen. Til slutt er det viktig å sette opp ”Chrome” til å starte i kiosk-modus ved oppstart, og sette innstillinger som forhindrer dvalemodus og gjør musepekeren usynlig.

3.4 Designprosess for terminal

For å kunne fullføre oppgaven var det nødvendig å designe og produsere en innkapsling for systemet som kunne beskytte mot vær og vind, og samtidig fungere som et brukergrensesnitt ovenfor brukerne av fergen. For brukeren skal terminalen primært være et medium for å bestille fergen, og i teorien krever dette kun en enkel knapp. Imidlertid vil det i praksis være nødvendig å inkludere en viss mengde informasjon rundt knappen for å gjøre bruken så intuitiv og brukervennlig som mulig.

Gruppen fikk anledning til å delta på et online-møte med Einar Hareide, som er en anerkjent ekspert innen design (*Hareide Design Norway 2023*). Under møtet fikk gruppen verdifulle innsikter og perspektiver om design av terminalen, og mulighet til å stille spørsmål direkte til ham angående design av terminalen. Tidlig i utviklingsfasen ble det utarbeidet diverse ideer i form av håndtegninger for å illustrere forskjellige sammensetninger av komponenter som knapper, skjerm, høyttalere og liknende. Hareide kom med kommentarer og innspill rundt design, designprosess, komponenter og materiale. Slik kom gruppen frem til et forslag til oppbygging av terminalen. Det var ønskelig å holde mengden nødvendig brukerinteraksjon med systemet lavt, slik at det alltid ville være så enkelt som mulig å bruke og forstå. Optimalt skulle det inkludere en trykk-knapp som lyser opp ved hjelp av en lysring, på samme måte som en heis-knapp gjør når den bestilles til ønsket etasje, og en skjerm som kunne vise ytterligere informasjon og samtidig løfte brukeropplevelsen betraktelig.

Det ble designet en rekke forslag til design gjennom CAD (Computer Aided Design) i programmet Fusion 360. Forslagene ble presentert for oppdragsgiver og medveiledere, og godkjent slik at gruppen videre kunne kontakte verksted for bestilling av stål. Gruppens endelige design presenteres i vedlegg C.

Designet måtte justeres før bestilling for å ta hensyn til aspekter som oppkutting av plater, bøyning og sveising, samt skruehull og vannsikring. Glen Angell, avdelingsingeniør på ITK med mye erfaring innen arbeid med stål og sveising, var behjelpelig med denne prosessen. I vedlegg D vises terminalen etter disse praktiske justeringe-

ne. Deretter ble terminalen bestilt og produsert. Terminalen ble bestående av tre hovedkomponenter; et frontpanel, et kabinett og en pdestall.

3.4.1 Skjerm

I startfasen av oppgaven forelå det mye diskusjon rundt hvorvidt terminalen skulle innholde en skjerm for å visualisere informasjon eller ikke. Oppgaven og systemet i seg selv setter ingen krav til at det skal foreligge en skjerm på terminalene. Det vil fungere etter sin hensikt kun ved å ha en knapp som kan trykkes for å bestille fergen. Gruppen diskuterte også tidsbruken som ville gått til implementasjon av skjerm, i tillegg til at det skapte nye problemstillinger til vann- og vandalsikring.

Oppdragsgiver uttrykte ønske om å ha en skjerm på terminalene i fremtiden. En skjerm vil kunne gi verdifull tilbakemelding til brukere om statusen til fergen, som for eksempel "Er bestillingen mottatt?", "Hvor er fergen?", og "Er det klart til å gå om bord?". Dette vil bidra til å øke brukervennligheten til systemet og dermed forbedre brukeropplevelsen. Det blir også lettere for oppdragsgiver å videreutvikle systemet hvis gruppen leverer et produkt som inneholder en skjerm. Det vil unngå behovet for å designe en ny terminal med skjerm i fremtiden. På grunn av disse fordelene besluttet gruppen å inkludere en skjerm på terminalene ved innlevering av prosjektet, for å gjøre systemet mer skalerbart og brukervennlig.

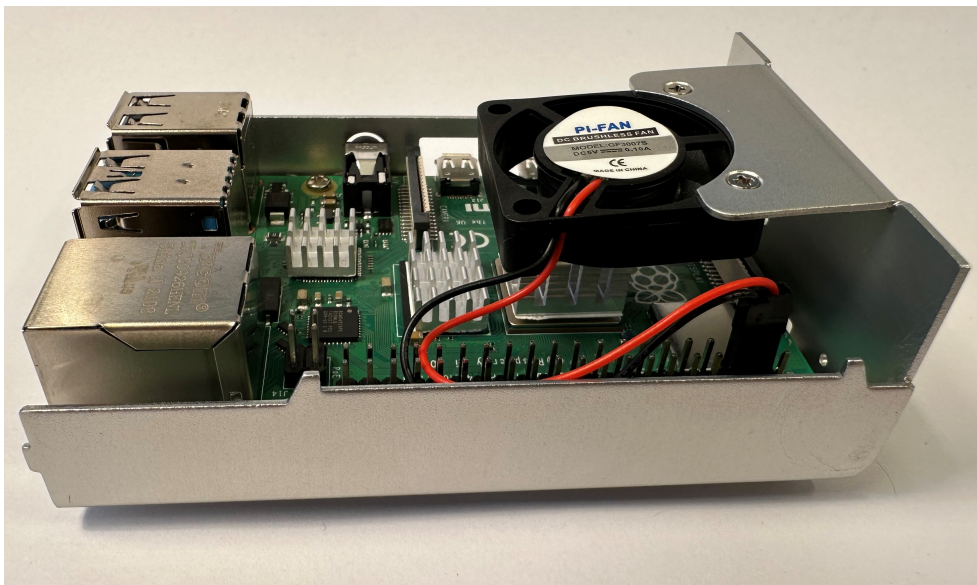


Figur 42: Vanntett og vandalsikker utendørsskjerm mottatt fra oppdragsgiver, med vanntette konnektorer.

Oppdragsgiver hadde allerede to skjermer tilgjengelig for tiltenkt bruk om bord i fergen, som gruppen har lånt og brukt til utvikling og testing av terminalene. Figur 42 viser skjermen med konnektorene som blant annet gjør at skjermen er vanntett. Fra oppdragsgiver ble gruppen henvist til et datablad for en lignende skjerm, da det ikke var datablad å oppdrive for gjeldene skjerm som ble brukt i prosjektet,

se vedlegg H. Skjermene er designet til å være vandalsikre, IP67 og beregnet for utendørs bruk med 1000 nits lysstyrke på displayet (Sihovison 2023). Terminalene er designet med tanke på størrelsen på skjermen, slik at det er tilstrekkelig plass til både skjerm og annet nødvendig utstyr for terminalene. Gruppen har gjennomført omfattende testing og bruk av terminalen, og har ikke oppdaget noen grunn til å tvile på kvaliteten på skjermen.

3.4.2 Skjermdriver



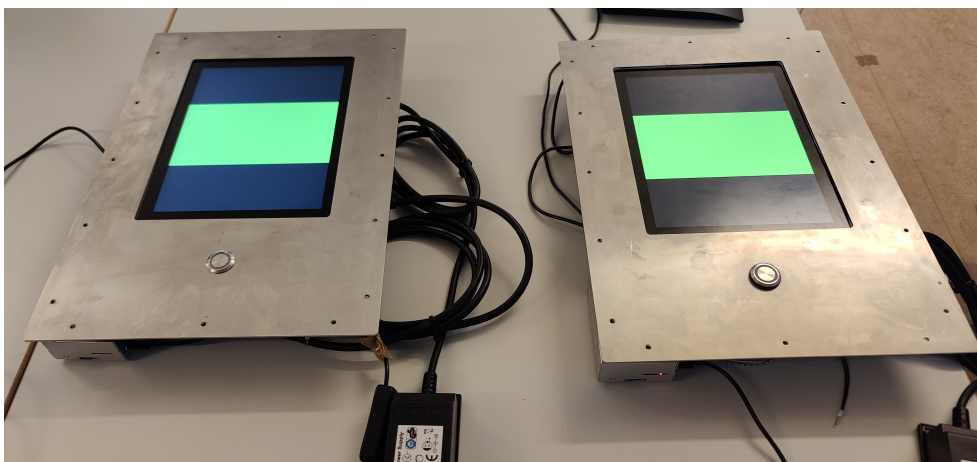
Figur 43: RPi 4B som brukes som skjermdriver.

Skjermen i terminalen skulle vise diverse informasjon og gi tilbakemelding til bruker om bestilling og fergens posisjon. Det var derfor nødvendig å inkludere en enhet som kunne drive skjermen, som beskrevet i seksjon 3.3.6. Gruppen hadde allerede en del erfaring med RPi og det ble derfor tidlig valgt å bruke RPi som driver av GUI og skjerm. For testing og utprøving fikk gruppen låne to RPi 3B+ gjennom IES (Institutt for Elektroniske Systemer). Senere når systemet nærmet seg ferdig utviklet ble det klart at RPi 3B+ hadde for lite prosessorkraft for å drive systemet, da CPU-en ble svært varm og systemet ofte hang seg opp. Gruppen undersøkte andre løsninger og fikk i den sammenheng lånt en RPi 4B fra “Omega Verksted NTNU”. Denne klarte å drive systemet uten store problemer og gruppen gikk derfor til innkjøp av to enheter som ble brukt i det endelige systemet. Det ble også gjort innkjøp av vifte, kjøleribber og kabinett til RPi 4B, da gruppen erfarte høy varmegang igjennom testing av den lånte RPi 4B. Figur 43 viser RPi 4B oppkoblet i kabinett med tilhørende vifte og kjøleribber på de mest kritiske komponentene.

3.4.3 Temperaturtest av RPi 4B

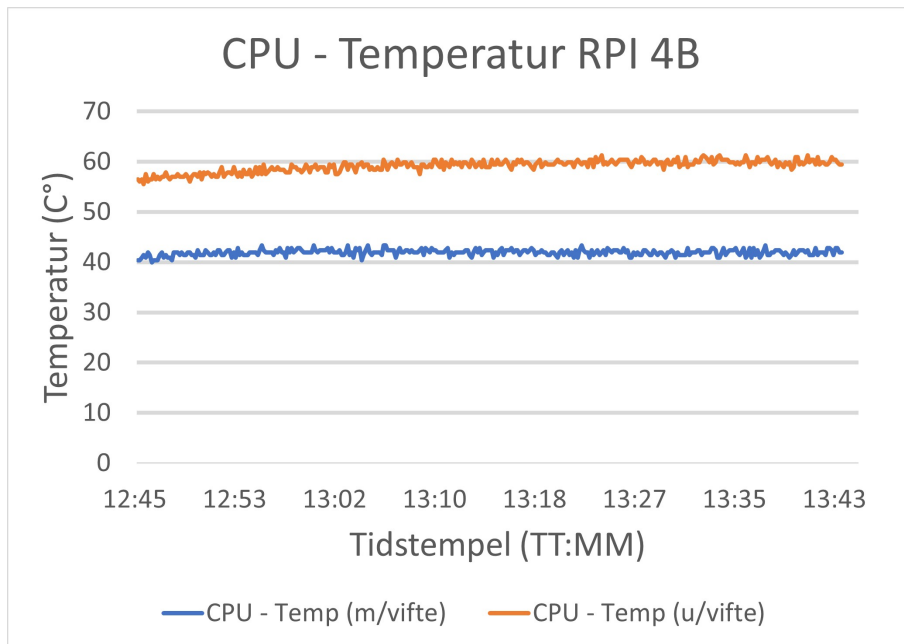
For å undersøke effekten ved bruk av vifte, gjennomførte gruppen en test for å se om viften bidro til å kjøle ned CPU-en til en lav nok arbeidstemperatur. Siden RPi 3B+ sin "soft limit" vanligvis er satt til 60°C, hvor klokkehastigheten blir redusert fra 1.4GHz til 1.2 GHz, ønsket gruppen å holde seg under denne verdien. Selv om RPi 4B ikke har en slik "soft limit" aktivert som standard, vurderte gruppen det til at det var en god pekepinn å holde seg under RPi 3B+ sin "soft limit" (Raspberry Pi 2023).

Testen ble gjennomført ved å la en RPi stå med vifte og den andre uten. Begge RPi-ene hadde kjøleribbe installert på de mest kritiske komponentene. For å la testene være mest mulig lik, ble det brukt de samme skjermene som skulle brukes til terminalene, og RPi-ene ble satt opp på samme sted iforhold til skjermen. I stedet for å kjøre den faktiske GUI-en på skjermen, ble det spilt av en test video for å kunne simulere oppdatering av skjermen uten å selv måtte fysisk oppdatere skjermen. Test video-en var en video som oppdaterte farger på skjermen, noe som kunne gi en jevn belastning på prosessorene til RPi-ene (Alex Voxel 2023). I figur 44 vises det fulle oppsettet av testen.



Figur 44: Testoppsett for temperaturtest av RPi 4B, med og uten vifte.

For å logge CPU-temperaturen til RPi-ene under testen ble det satt opp et Python-program. Python-programmet logget CPU temperatur, tidstempel og CPU forbruk hvert 10. sekund og lagret det i en .csv-fil. Testen ble gjennomført i en time for å kunne samle inn nok data til å kunne få et jevnt og informativt resultat.

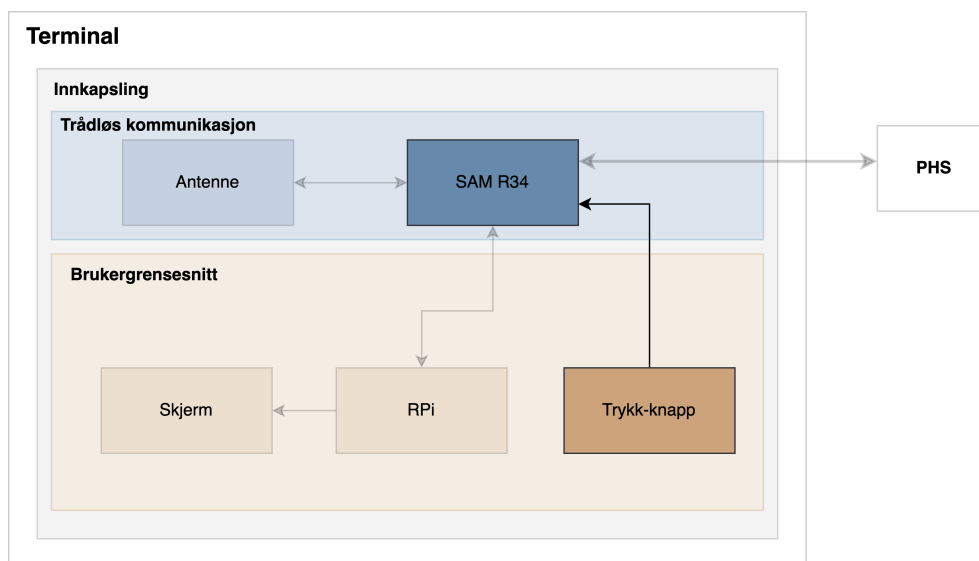


Figur 45: Resultat fra temperaturtest av RPi 4B, med og uten vifte.

Figur 45 viser resultatet fra testen. Selv om CPU-forbruket også ble logget valgte gruppen å ekskludere dette resultatet da CPU-forbruket lå jevnt rundt 11% for begge RPi-ene gjennom hele testen. Resultatet viser at til tross for tilnærmet likt CPU-forbruk ligger RPi-en uten vifte omtrentlig 18°C over RPi-en med vifte. I tillegg ligger temperaturen til tider over 60°C, som gruppen hadde satt som sin øvre grenseverdi, som beskrevet over.

Basert på testen valgte gruppen at oppsettet med bruk av vifte var nødvendig for at RPi-ene skulle fungere som ønsket for sitt bruk. Det er her viktig å merke seg at testen ble gjennomført utenfor innkapsling for å lage en mest mulig lik test for de to terminalene. Dette kan derimot skape usikkerhet rundt resultatene, da det kunne vært annerledes om RPi-ene var innkapslet. Andre usikkerheter verdt å nevne er at sol, vind og ytre temperatur kan også ha en påvirkning på klima inne i boksen, og derfor ha effekt på temperaturen til RPi-en.

3.4.4 Trykk-knapp



Figur 46: Systemarkitekturen med fokus på trykk-knappens posisjon og tilknytning til SAM R34.

Trykk-knappen ble koblet opp for å gi brukere av systemet en manuell måte å bestille fergen på. Oppkoblingen ble gjort ved å koble den positive terminalen for LED-ringen til digital port PA06, og den negative terminalen til jord på SAM R34. Porten evaluerer til høy ved mottatt bestilling, og lav når fergen lukker porten ved den respektive terminalen, som forklart i seksjon 3.3.4. Trykk-knappens tilstand ble koblet til å være normalt lukket, NC, og ved trykk på knappen vil den gi høy verdi inn på porten. NC og C (Changeover) ble koblet til henholdsvis port PA07 og GND. Figur 46 fremhever knappens posisjon i systemarkitekturen.

3.4.5 Vann- og vandalsikring

Siden terminalene var tenkt til utendørs bruk, var det derfor viktig å sikre diverse elektriske komponenter bak vanntett forsegling. Dette ble løst ved hjelp av sveising, pakninger, kabelgjennomføringer og o-ringer. Der terminalen er sveiset og bøyd vil det ikke være mulig for vann å trenge gjennom da dette er rent rustfritt stål.

Terminalens frontpanel og kabinett har tre åpninger for komponenter, to hull til kabler, samt en glippe mellom frontpanelet og kabinettet. Disse områdene var avgjørende å forsegle for å sikre at terminalens indre forblir tørt. Knappen, skjermen og antennen er alle komponenter som stikker ut av terminalen og vil bli direkte utsatt for vær og vind. Komponentene i seg selv er rangert med minimum IP65 og er av den grunn ikke nødvendig å forsegle videre mot støv og vann.

For å hindre fukt fra å trenge inn i terminalen, måtte glippen mellom kabinettet og frontpanelet tettes med en pakningslist. Skjermen ble først presset mot en pakningslist montert på terminalens frontpanel. Frontpanelet ble så presses mot

pakningslisten på terminalens kabinett for å tette glippen mellom kabinett og frontpanel. Hullene til kablene ble montert med en IP68 kabelgjennomføring for å videre forhindre fuktighet fra å trenge inn i terminalen.

Vandalsikring var en viktig faktor når det gjaldt utendørs bruk av terminalen. Gruppen hadde to hovedmål for konstruksjonen med hensyn til vandalsikring. Det første målet var at terminalen skulle være robust og kunne tåle påkjenninger som dytting, slag, skraping og sparking. Det andre målet var å minimere svakheter som kan utnyttes for å begå hærverk. Eksempelvis unngå utstikkende objekter som lett kunne knekkes av.



Figur 47: Terminal med pakning for vannsikring.

I figur 47 illustreres hvordan frontpanelet på terminalen er montert over en pakningslist for å oppnå en tett forsegling. Det ble lagt til rette for at panelet skal kunne monteres av for vedlikehold. Dette gjøres ved hjelp av 16 M3-skruer som sikrer en tett forsegling.

For å sikre at terminalen kunne tåle de påkjenningene de potensielt kunne bli utsatt for i drift, ble konstruksjonen laget av rustfritt stål som er kjent for sin høye hardhet og robusthet. Rustfritt stål har også en høy korrosjonsbestandighet, som gjør at det vil kunne motstå rust og andre skadelige påvirkninger fra vær og vind over tid (Gardner 2005).



Figur 48: Til venstre vises trykk-knapp med o-ring og til høyre vises trykk-knapp montert på frontpanelet

Trykknappen som ble valgt er spesielt designet for å være vanntett og beskyttet mot fuktighet med IP65/67 gradering. Den har en o-ring som forhindrer at vann kan trenge inn i terminalen og skade de elektroniske komponentene. Låsemutteren sikrer at knappen og o-ringen kan klemmes tett inn mot metallet som vist i figur 48.

Når knappen er montert i henhold til databladet (EAO 2016), kan en være trygg på at den er fullstendig vanntett og vil fungere pålitelig selv under krevende forhold. Trykk-knappen er IK10 grader og produsert i rustfritt stål, gjør den godt egnet for å tåle harde påkjenninger.

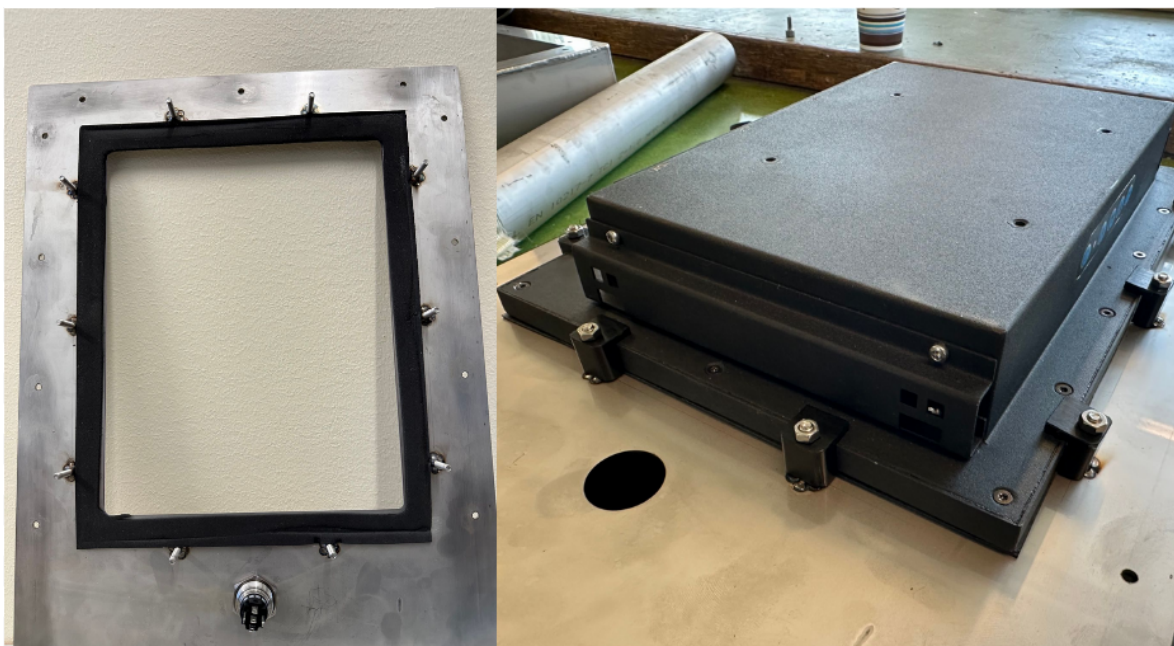


Figur 49: Undersiden av antennen har et gummibelegg som sørger for at vann ikke vil kunne trenge inn i terminalen.

Antennen, ANT-8/9, montert på baksiden av terminalen er i seg selv vurdert til IP67, men for å beskytte innsiden av terminalen fra vann å trenge gjennom hullet til antenne er det en stor gummi-overflate på undersiden av antennen som presses mot utsiden av terminalen, se figur 49. En låsemutter og en låseskive sørger for at antennen og gummiene sitter tett på metallet på baksiden av antennen og at det er en god forsegling mot vann. Så fremt antennen er riktig installert i henhold til datablad, vil den være vanntett inntil IP67 (Linx Technologies 2021).

ANT-8/9 er spesielt designet for å være vandalsikker. Dette betyr at den er konstruert for å tåle grov behandling og skade fra personer som bevisst forsøker å ødelegge den. Antennen er laget av robuste materialer som er motstandsdyktige mot støt og slag, og den er beskyttet av en solid ytre kapsling som hindrer at den blir skadet av ytre påvirkninger (Linx Technologies 2021). I tillegg er antennen montert slik at den er vanskelig å ødelegge uten spesialverktøy eller kunnskap om montering. Antennen er derfor ideell for bruk i offentlige områder, der antenner kan være utsatt for hærverk og skade.

Skjermen som ble brukt i prosjektet var utlånt av oppdragsgiver og er egnet for bruk i marine miljøer, da den har ifølge oppdragsgiver en IP67-klassifisering som gjør den motstandsdyktig mot vann. Skjermen er også veldig robust og beskyttet mot hærverk, ettersom den er laget av solide materialer som er vanskelige å deformere, se seksjon 3.4.1.



Figur 50: Feste for skjerm som sørger for tett forsegling ved rammen rundt skjermen.

Når det gjelder vanntetthet rundt skjermen, er det viktig å sørge for en pålitelig tetning for å unngå at vann eller fuktighet trenger inn i terminalen. Derfor har skjermen en pakning montert mellom skjermen og terminalens frontpanel som vist i figur 50. For å feste skjermen til metallet med tilstrekkelig kraft, brukes stålfester på baksiden av skjermen. Disse festene er konstruert for å presse skjermen tett mot frontpanelet. I kombinasjon med pakningen gir dette en solid og pålitelig forsegling som holder fuktighet ute og beskytter innsiden av terminalen mot fukt.

Kabler som går ut og kommer inn til terminalen fra utsiden, går gjennom en IP68 kabelgjennomføring, se figur 51. Disse forsikrer at fukt ikke trenger inn i terminalen.



Figur 51: Kabelgjennomføring som representerer tilsvarende brukt på kabler som går inn og ut av terminalen for å forhindre fukt inn (RS Pro 2023).

3.4.6 Vanntest på terminal

Det ble i første omgang gjennomført stress-tester for å kontrollere terminalens motstandsdyktighet mot vanninntrengning. I forkant av testen ble skruhull til terminalens pidestall og kabelgjennomgang dekket med tape fra begge sider, for videre å montere terminalens bestillingsknapp til front-panelet og antennen til kabinettet. Gruppen plasserte tørkepapir i terminalen for å lettere kunne se om væske hadde trengt inn. Til slutt ble frontpanelet, med påmontert skjerm, montert med M3-skruer til kabinettet med to ulike typer pakninger, en for hver test.

Frontpanelets M3-skruer ble festet ved hjelp av et kryssmønster, som involverte å feste en skrue i et hjørne før neste skrue ble festet i det diagonalt motsatte hjørnet. Alle skruene ble festet tilsvarende før de ble strammet til i tilsvarende mønster. Dette ble gjort for å sikre en jevn fordeling av trykk rundt hele kabinettet og pakningen.

Testene bestod av å plassere det sammenkoblede frontpanelet og kabinettet, i en dusj av vann over en periode på ti minutter, for å simulere regn. Figur 52 viser oppsett av testene.



Figur 52: Oppsett til testene som ble utført for de ulike typene tetningslister.

I den første testen ble en hard neopren-pakning på 1 mm testet mellom frontpanelet og kabinettet. Etter stress-testen ble terminalen demontert for å undersøke om væske hadde trengt inn. Det viste seg at betydelige mengder vann hadde trengt inn, og det var nødvendig å gjennomføre ytterligere tester med nye løsninger.

Test to ble gjennomført i likhet med den første, men med et mykere materiale som pakningslist. Resultatet fra testen viste at den myke pakningen tetter bedre enn neopren-pakningen. Det ble dermed konkludert med å få anskaffet en mykere pakning på mellom 2 mm og 4 mm til bruk mellom både skjerm og frontpanel, samt frontpanel og kabinett.

Etter anskaffelse av 2 x 3 meter med 19 mm x 3 mm vinylskumgummi-pakningslist, ble det gjennomført en ny test i likhet med de to foregående testene. Listen ble påført kabinettet og frontpanelet, og plassering av paknings-skjøtene ble valgt for å optimalisere beskyttelsen til pakningslistene. Regnvann vil treffe listen i øvre del av kabinettet, og det var derfor strategisk å plassere skjøten i nedre del. I motsetning, vil regnvann treffe nedre del av listen på frontpanelet og skjøten ble av den grunn plassert i øvre del. Figur 53 viser neoprenpakningen til venstre og vinylskumgummi-pakningen til høyre.



Figur 53: Terminalens kabinett montert med neoprenpakning til venstre. Til høyre vises terminalens kabinett montert med vinylskumgummi-pakning.

3.5 Modem på ferge

Oppdragiver foreslo å legge utviklingskortet under dekk på fergen og trekke en koaksialkabel opp til antennen i masten, hvor antennen kunne skrues fast i metallplaten og stikke rett opp. Denne løsningen ga flere fordeler ved at det ville være lettere å tenke på innkapsling ettersom det allerede er vanntett og sikkert under dekk, i tillegg til at det er mer plass og lettere å få tilgang til kortet i ettertid hvis det oppstår behov for å rette feil eller implementere oppgraderinger.

Det måtte derimot undersøkes hvilken effekt det hadde på linkbudsjettet, ettersom det ville oppstå vesentlig større kabeltap på grunn av lenger kabel mellom utviklingskortet og antennen. Kabelen som benyttes er RG-223 og har en demping på 43.5 dB per 100 meter (Multicomp 2020). Det tilsvarer 0.435 dB per meter. Det anslås at det er en maksimal lengde på 5 meter fra kortet under dekk til antennen i masten. Det gir en kabeldemping på omtrent 2.18 dB.



Figur 54: DC-DC omformer brukt til å forsyne SAM R34 med strøm (CUI 2022).

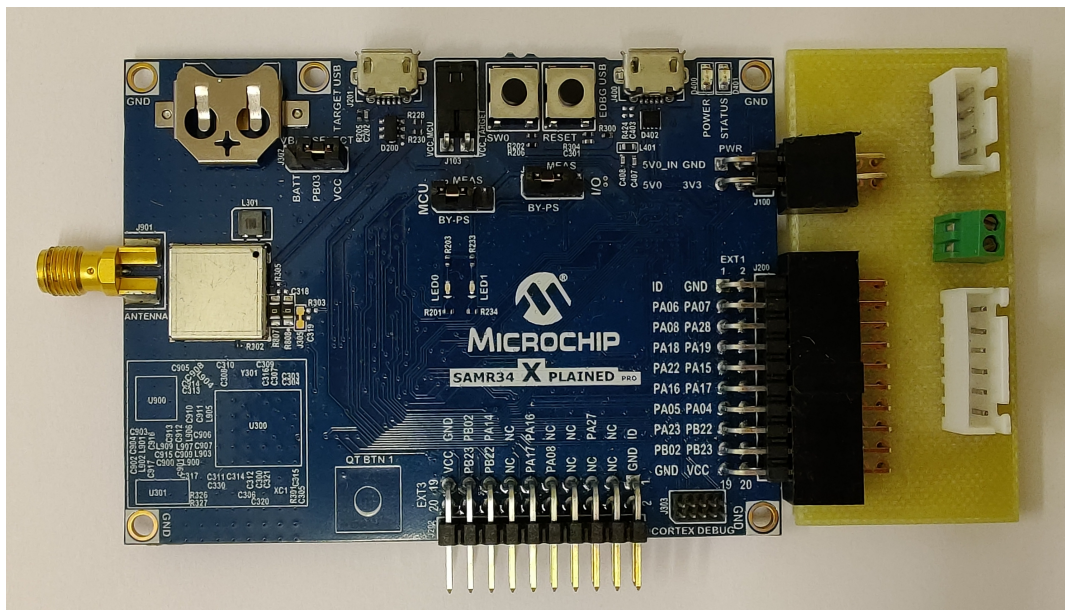
Utviklingskortet SAM R34 krever en inngangsspenning på 5 V og kan variere mellom 4.3 V og 5.5 V, og det anbefales minimum 1.0 A inngangsstrøm (Microchip Technology 2020b). Under dekk på fergen er det kun tilgjengelig 24 V og av den grunn er det behov for en DC-DC omformer for å forsyne mikrokontrolleren med strøm. En DC-DC-omformer (Direct Current to Direct Current - omformer) er en elektrisk enhet med formål å regulere en DC-inngangsspenning til en høyere eller lavere DC-utgangsspenning (Floyd 2012, s. 62).

PQDE6W, som illustreres i figur 54, har utgangsspenning på $5\text{ V} \pm 85\text{ mV}_{P-P}$ og leverer 1.2 A utgangsstrøm. Det er innenfor spesifikasjonene til SAM R34 siden spenningen aldri vil bli mindre enn 4.3 V eller større enn 5.5 V. Det var også ønskelig at omformeren har god effektivitet ettersom den bruker strøm fra batteriet om bord i fergen. Det oppgis at den har en effektivitet på 83%, som ikke er den beste effektiviteten en DC-DC kan tilby, men det er akseptabelt og vil ikke føre til store problemer. En annen fordel med omformeren er at den er innkapslet og dermed isolert fra andre komponenter.

3.6 Utvidelseskort

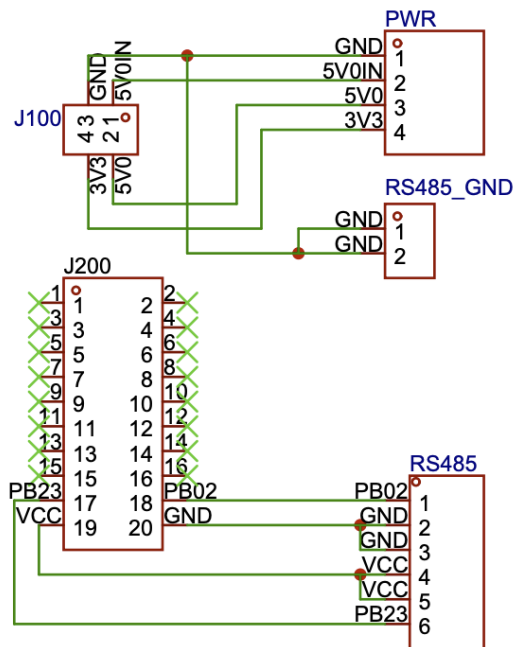
IO-pinnene på SAM R34 var ikke tilrettelagt for pålitelig tilkobling. Det var mulig å koble til ledere, men disse satt veldig løst og det ville vært en stor svakhet dersom utstyr var tilkoblet slik i det ferdige systemet. Av den grunn ble det kartlagt at det var mulig å frese ut enkle kretskort på NTNU. Det ble designet to unike kretskort

i PCB-designverktøyet, EasyEDA (EasyEDA 2022) for begge terminalene og ferge-modem. Figur 55 viser ferge-modemets utvidelseskort tilkoblet SAM R34.

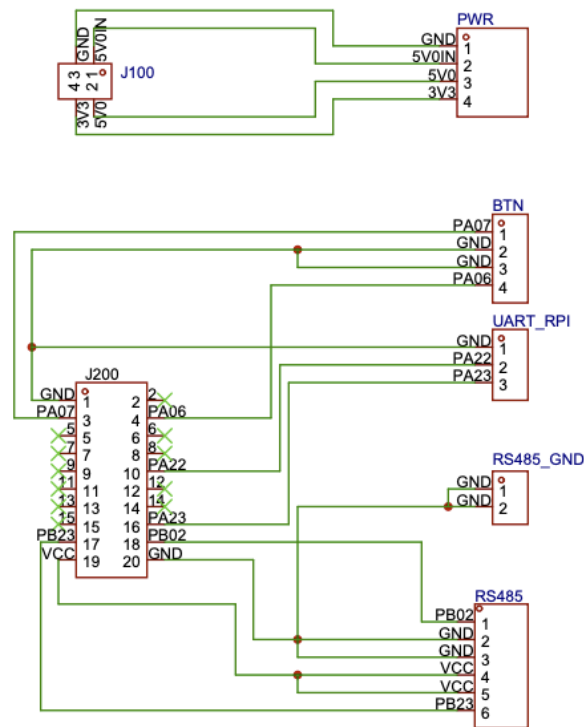


Figur 55: Ferge-modemets utvidelseskort tilkoblet SAM R34.

I figur 56 og 57 presenteres krets-skjemaene for utvidelseskortet til SAM R34 på henholdsvis ferge-modemet og terminalene. Kortene muliggjør en enklere og tryggere tilkobling av flere enheter, inkludert RS485, UART, og digitale I/O-tilkoblinger til bestillingsknapper.



Figur 56: Skjemategning for utvidelseskort til ferge.



Figur 57: Skjemategning for utvidelseskort til terminal.

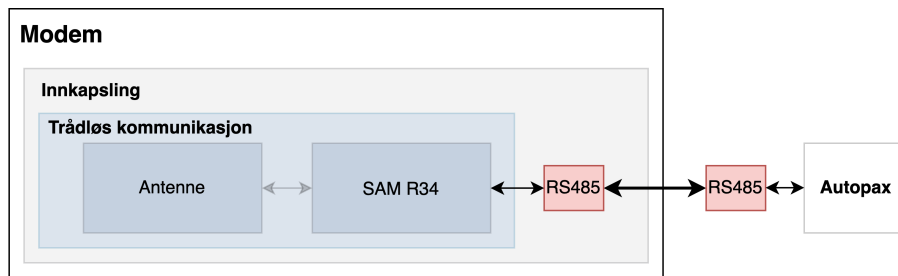


Figur 58: 2x10 Angled Female Pin Header til venstre (*Angled Female Pin Header* 2023), skruterterminal i senter (*Skruterterminal* 2023) og JST kontakt til høyre (*JST konnektor* 2023).

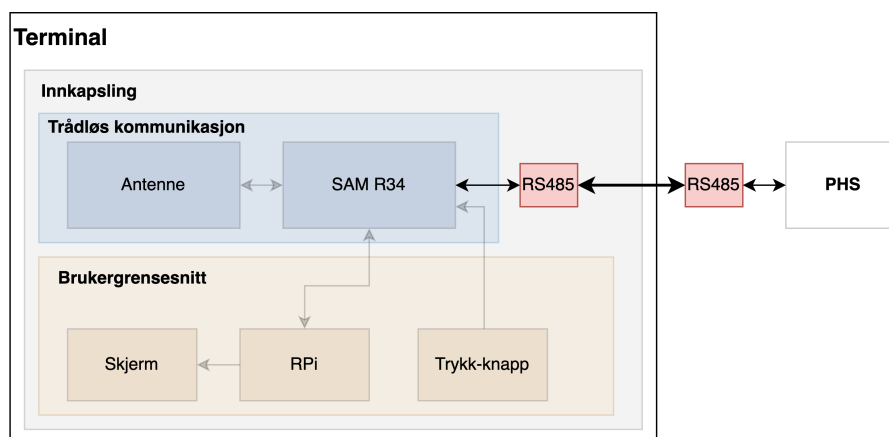
Komponentene J100 og J200 på utvidelseskortene er av typen "Angled Female Pin Headers" med størrelse 2x2 og 2x10. Disse komponentene kobles direkte til J100 og J200 på SAM R34. For koblingspunktet RS485_GND benyttes skruterterminal, mens de øvrige koblingspunktene er av typen JST (JST 2023), se figur 58. Komponentene ble loddet på kretskortene i henhold til risikoanalysens forebyggende tiltak gitt i vedlegg E.

3.7 Oppsett og konfigurering av RS485

Siden terminalene og modem må legge til rette for seriell kommunikasjon over lengre distanser må et fysisk medium tilpasset denne type bruk anvendes. Kommunikasjonen mellom terminalene og PHS, samt modemmet og Autopax-maskinen på fergen, skjer via en UTP-kabel (Unshielded twisted pair-kabel). I hver ende av UTP-kabelen er en RS485 "transceiver" som muliggjør seriell dataoverføring over lengre distanser. En transceiver er en enhet som både kan sende og motta signaler (Frenzel Jr 2023, s. 335). Figur 59 og 60 viser RS485-transceiverne og deres tilknytning til modem og terminal.



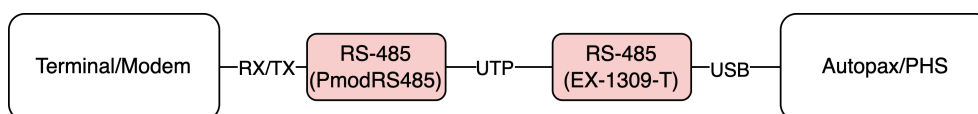
Figur 59: Modemets utvidede arkitektur med RS485.



Figur 60: Terminalens utvidede arkitektur med RS485.



Figur 61: Til venstre vises PmodRS485 (*PmodRS485* 2023) og til høyre EX-1309-T (*EX-1309-T* 2023).



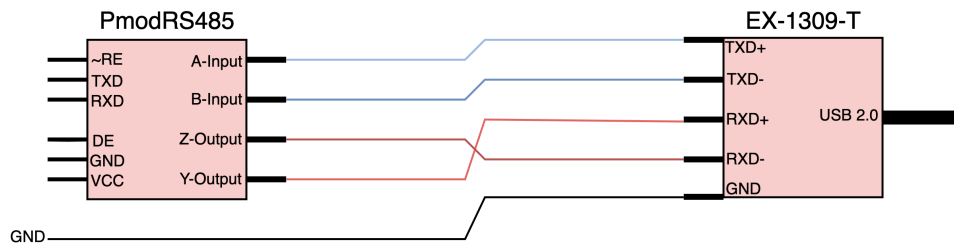
Figur 62: Grensesnitt mellom Terminal/Modem og Autopax/PHS.

I grensesnittet mot ITK-gruppens PHS og Autopax anvendes transceiveren EX-1309-T, og i likhet PmodRS485, i grensesnittet mot terminalen og ferge-modem. Figur 62 illustrerer grensesnittet mellom terminal og PHS med deres RS485-modul samt det fysiske mediet informasjon sendes over.

M0	1	<input type="checkbox"/>	NO
M1	2	<input type="checkbox"/>	
M2	3	<input checked="" type="checkbox"/>	
TERM	4	<input checked="" type="checkbox"/>	
T+	5	<input type="checkbox"/>	
T-	6	<input type="checkbox"/>	
R+	7	<input type="checkbox"/>	
R-	8	<input type="checkbox"/>	

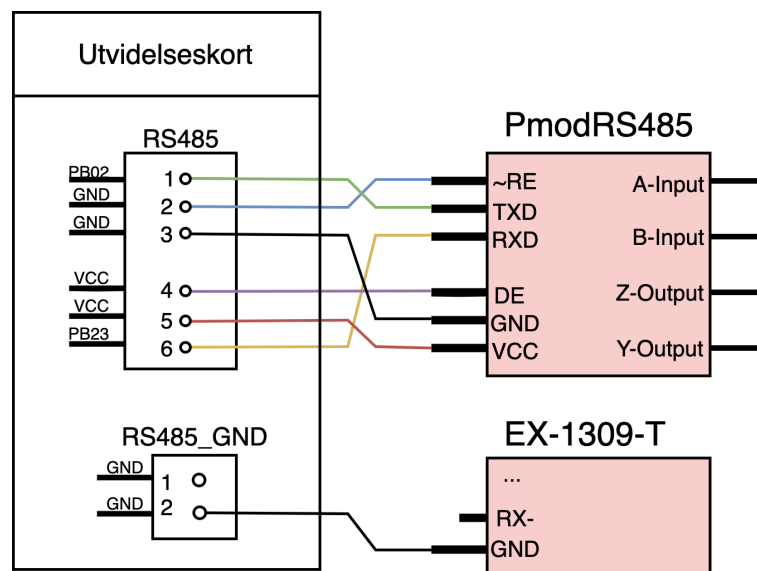
Figur 63: EX-1309-T DIP-switch innstilling for RS485 4-wire.

EX-1309-T er en multifunksjonell seriell transceiver. Den har mulighet for både RS485, RS422 og RS232 (Exsys 2019). Modulen kommer med en DIP-Switch som illustrert i figur 63. Denne anvendes for å sette de ulike innstillingene for dataoverføring. Figuren viser innstillingene for transmisjonsmodus RS485 4-wire som benyttet i kommunikasjonen mellom terminal/modem - PHS/Autopax.



Figur 64: Skjematisk oppkobling av grensesnitt mellom Terminal/Modem og Auto-pax via RS485. Figuren illustrerer ikke anbefalingen om UTP-kabel ved bruk av RS485.

Av databladet til PmodRS485 kommer det frem at B-Input og Z-Output er modulens inverterte terminaler og de kobles av den grunn til henholdsvis TXD- og RXD- på EX-1309-T. Videre oppkobling skjer i samsvar med figur 64.



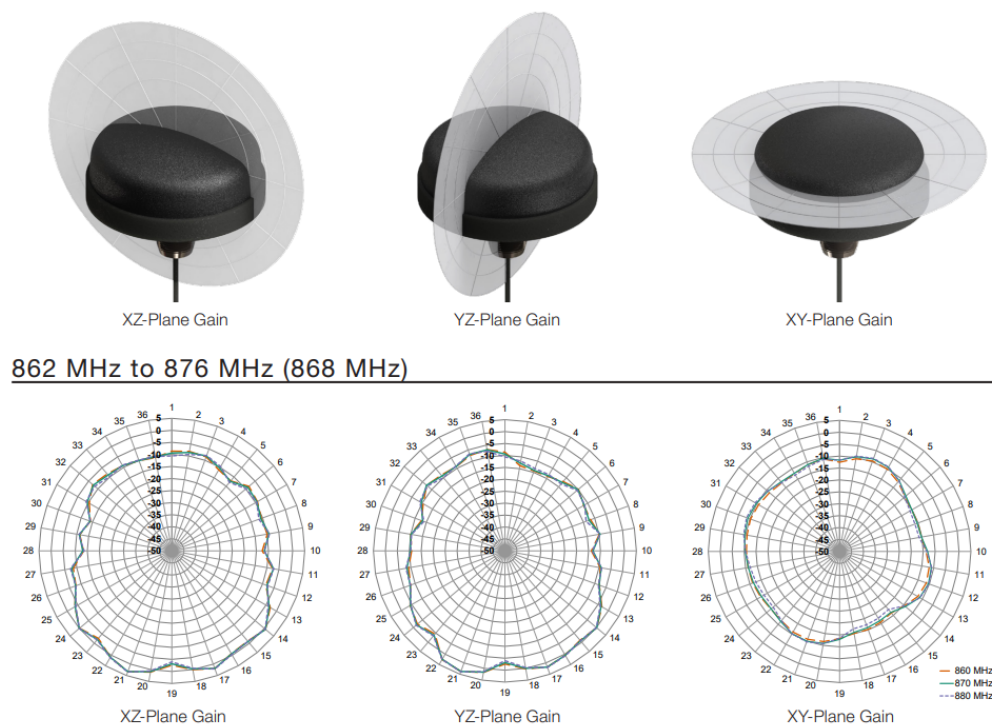
Figur 65: Oppkobling av PmodRS485 og EX-1309-T til utvidelseskortets JST-koblinger, RS485 og RS485_GND.

Figur 65 viser sammenkobling av PmodRS485 til utvidelseskortene, som beskrevet i seksjon 3.6. PmodRS485-modulen kommuniserer med SAM R34, gjennom utvidelseskortet, via UART-protokollen. For å kunne sende data må DE (Driver Enable) pinnen på PmodRS485 drives til en logisk høy spenningstilstand. I motsetning må RE (Receiver Enable) pinnen drives til en logisk lav spenningstilstand for å kunne motta data (Digilent 2021). På denne måten vil PmodRS485 være konfigurert til full-duplex. Tilkoblingen av PmodRS485 til både modem og terminalenes utvidelseskort skjer i henhold til figur 65.

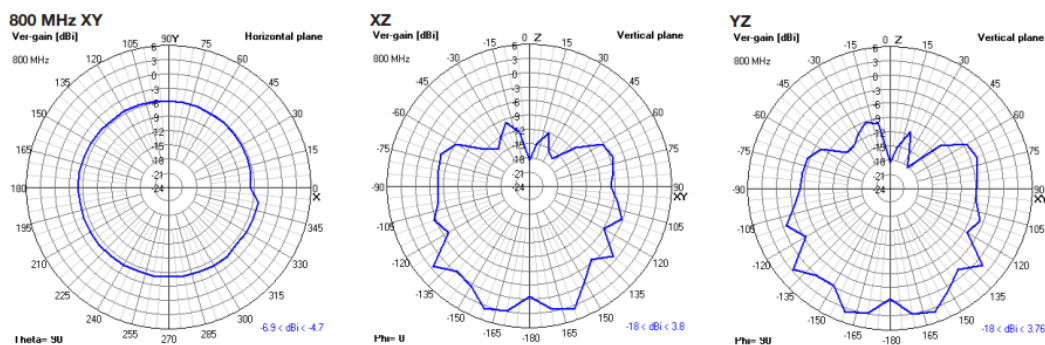
3.8 Signalstyrke og parametere

Linkbudsjett var et viktig verktøy for å kartlegge om gitte parametere er tilstrekkelig for å oppnå pålitelig trådløs kommunikasjon. Beregninger ble utført for å vurdere behovet for eventuelle endringer i parametere for å oppnå tilstrekkelig signalstyrke, og for å sammenligne teoretiske forventninger med praktiske resultater.

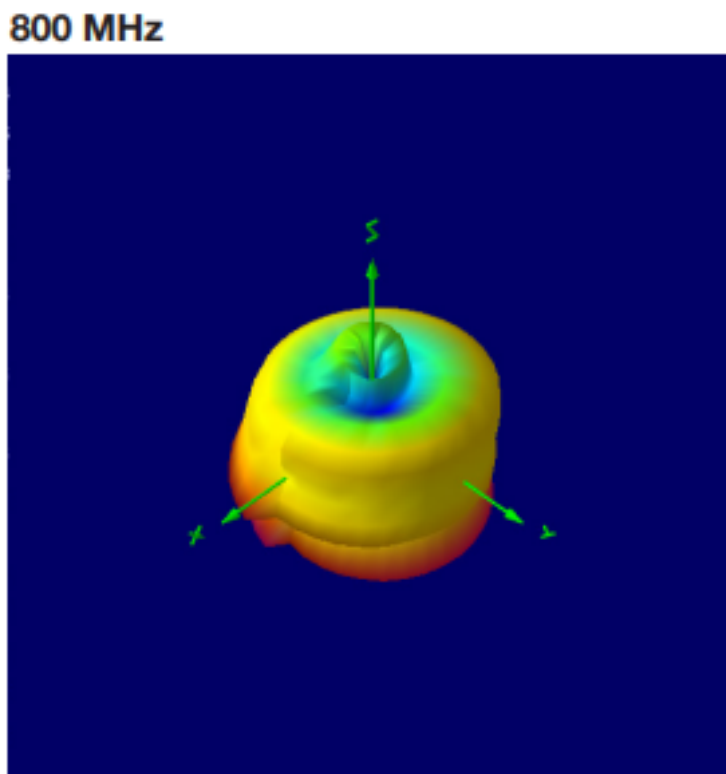
Antennen på terminalen har en tilhørende 1 meter lang RG-174/U koaksialkabel (Linx Technologies 2021). Kabelen oppgis til å ha en demping på 31.0 dB per 100 fot (BELDEN 2022). Det tilsvarer omtrent 31.0 dB per 30 meter (Wight 2018), som igjen er 1.03 dB per meter. Siden kabelen er 1 meter lang, er kabeldempingen for ANT 8/9 lik 1.03 dB.



Figur 66: Strålingsdiagram i xy-, xz- og yz-planet for ANT 8/9 (Linx Technologies 2021).



Figur 67: Strålingsdiagram i xy-, xz- og yz-planet, i 2D, for Delta 5A (Siretta 2023).



Figur 68: Strålingsdiagram i 3D for Delta 5A (Siretta 2023).

ANT har en gjennomsnittlig vinning på -6 dBi (Linx Technologies 2021) og Delta 5A har 3 dBi vinning (Siretta 2023), begge ved 868 MHz. Strålingsdiagrammet til ANT 8/9 viser at den har omtrent lik signalstyrke i alle retninger, men litt mindre i yz-planet som vist i figur 66. Vinklingen av antennen vil ha lite å si, ettersom mottaker vil motta signal pålitelig, uansett hvor den er plassert, i forhold til strålingsdiagrammet.

Delta 5A stråler ulikt avhengig av hvilken retning radiobølgene beveger seg. Figur 67 og 68 viser tydelig at den stråler best i xy-planet, men har kun god stråling mellom 90° og -90° i xz- og yz-planet. Antennen vil stikke opp av masten på fergen, og mottaker-antennen vil omtrent ligge i retning med xy-planet, og det gir tilstrekkelig signalstyrke.

LoRa oppgis til å ha en mottaker-sensitivitet lik -125 dBm. Det gjelder ved båndbredde på 125 kHz og spredningsfaktor 7 (Semtech 2020). Sendereffekten er programmerbar, men kan maksimalt være +14 dBm i henhold til EU-reguleringer (Microchip Technology 2021c). Ved prosjektets start ble sendereffekten satt til 1 dBm for å teste kommunikasjonen med nesten lavest mulige spesifikasjoner, som også fører til lavere effektforbruk.

Alle verdier for å utlede tapet som oppstår når signalet sendes gjennom trådløst medium er dermed kjent. Det gir $FSPL = 74.7$ dB, se formel 3. Utregning av FSPL tillater å sette opp linkbudsjett for hele systemet.

Linkbudsjett		
Sender	Verdi	Enhet
Utgangssignalnivå SAM R34	1	dBm
Kabeldemping ANT	-1.03	dB
Antennevinning ANT-8/9	-6	dB
Trådløst medium		
FSPL	74.7	dB
Mottaker		
Antennevinning Delta 5A	3	dB
Kabeldemping Delta	-2.18	dB
Signalstyrke ved mottaker	-79.91	dBm
Sensitivitet SAM R34	-125	dBm
Link margin	45.09	dBm

Tabell 2: Alle forsterkninger og tap som følge av trådløs kommunikasjon fra terminal til ferge-modem.

Link budsjettet i tabell 2 viser teoretisk at systemet klarer å sende pålitelig data mellom enhetene, og av den grunn er de første testene basert på disse verdiene.

3.9 Utstysrliste

Alt utstyr som benyttes i det totale bestillingssystemet fremgår av tabell 3 og 4.

3.9.1 Ferge

Utstyr	Delnummer	Antall enheter
Antenne-pisk	DELTA5A/X/TNCM/S/S/17	1
DC-DC omformer	PQDE6W-Q24-S5-T	1
Innkapsling (Aluminium)	1455P1601BK	1
Koaksialkabel (5m)	RG223	1
RS485-modul	410-310	1
SAM R34 Xplained Pro	DM320111	1
TNC - adapter	122346	1
USB-seriell omformer	EX-1309-T	1
Utvidelseskort		1

Tabell 3: Komponentene som benyttes i fergen.

3.9.2 Terminal

Utstyr	Delnummer	Antall enheter
Antenne-puck	ANT-8/9-SPS1-1	2
SAM R34 Xplained Pro	DM320111	2
RS485-modul	410-310	2
Skjerm (LCD)	SL133	2
Raspberry Pi 4B - kit*	5011179832	2
Trykk-knapp m/lysring	124-4516	2
USB-seriell omformer	EX-1309-T	2
Utvidelseskort		2

*Kit inneholder: Vifte, innkapsling og strømforsyning 5v

Tabell 4: Komponentene som benyttes i terminalen.

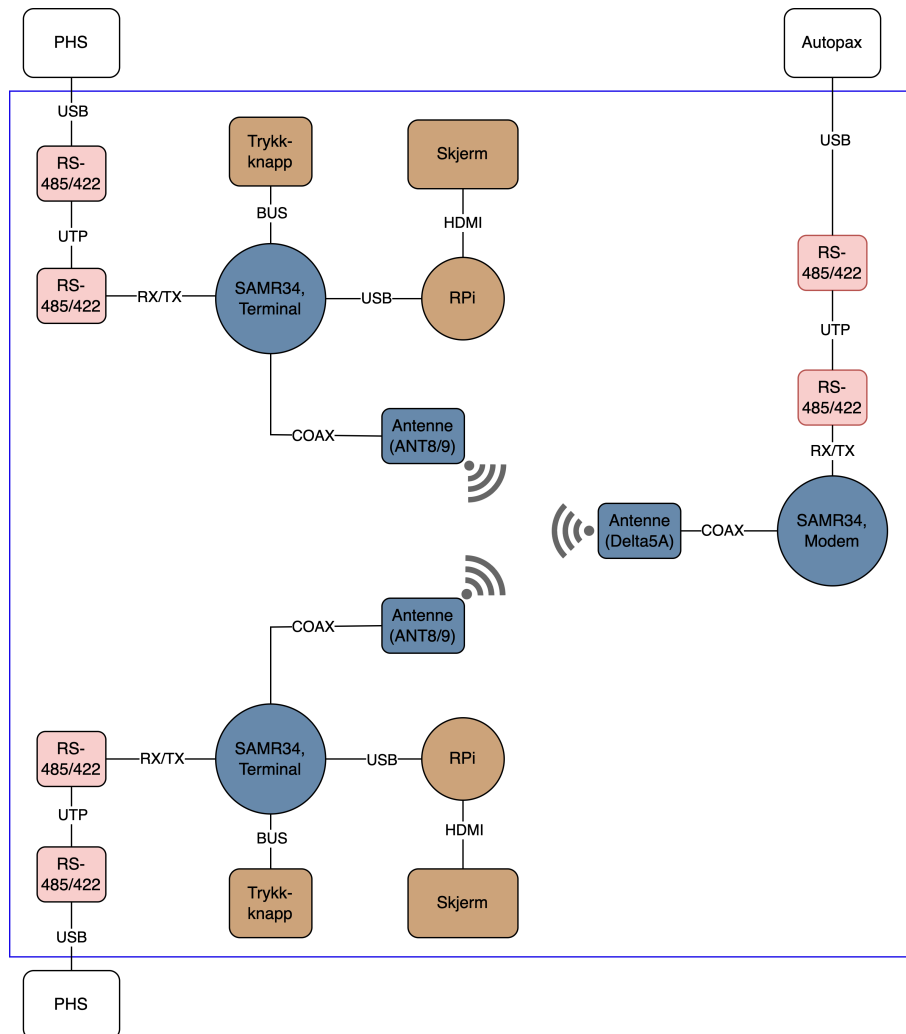
3.9.3 Programvare

Program	Utvikler	Funksjon
Figma	Figma Inc.	Brukergrensesnitt design
Fusion 360	Autodesk	3D-modellering
GanttProject	BarD Software s.r.o. mfl.	Prosjektorganisering
GitHub	GitHub	Versjonskontroll
Microchip Studio	Microchip Technology	IDE
Professional Online Editor	EasyEDA	Kretsdesign

Tabell 5: Programvare som ble benyttet til produktet.

3.10 Oppsett, test og simulering av systemet

Figur 69 viser det komplette systemet med dens komponenter og fysisk medium for kommunikasjon mellom de ulike del-systemene samt kommunikasjonen mellom ITK-gruppens PHS og Autopax-maskin.

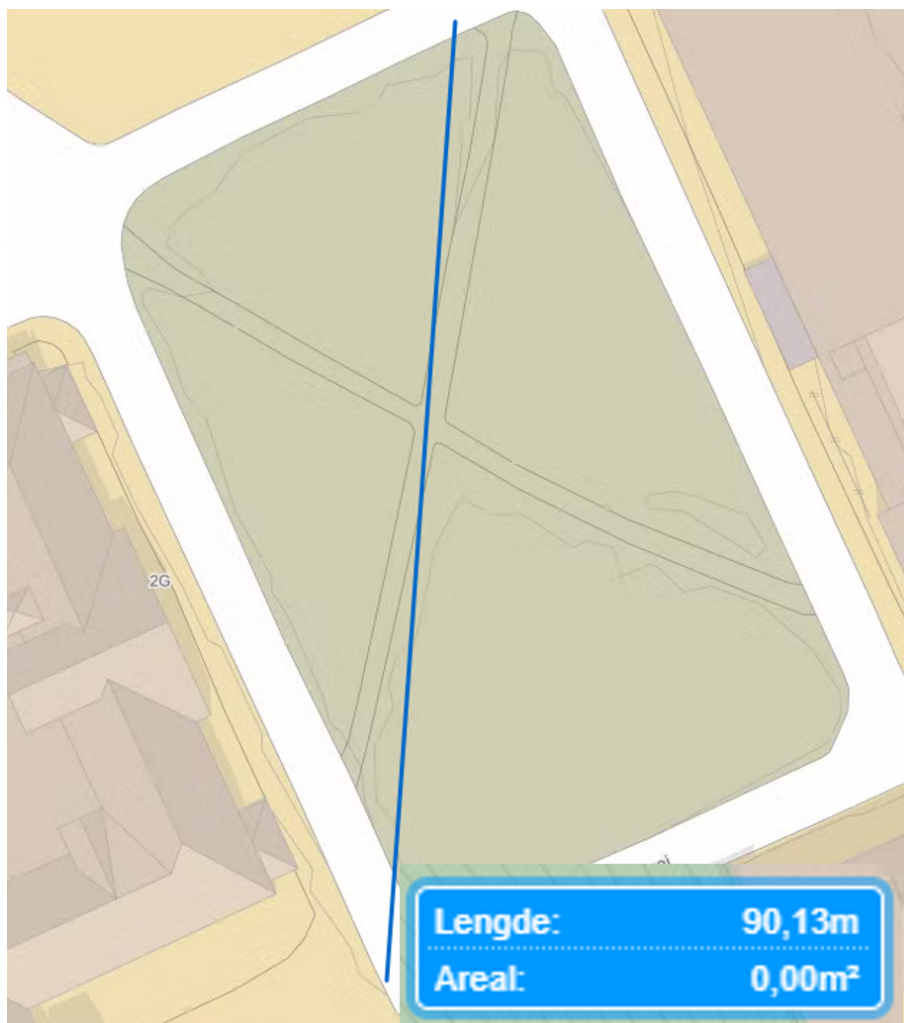


Figur 69: Komplet skjematisk fremstilling av det totale systemet samt fysisk medium for kommunikasjon mellom de ulike delsystemene.

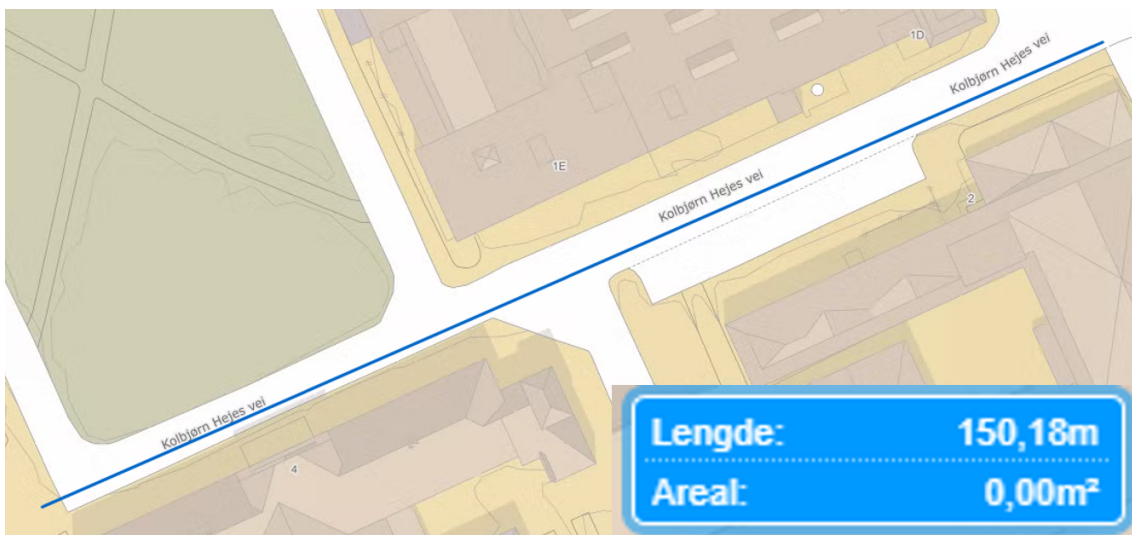
I bestillingen av SAM R34 fulgte det med antenner som ble brukt til testing inntil antennen som skulle benyttes i det ferdige systemet ankom. Det er dårlig dokumentert hvilke antenner som fulgte med. Databladet oppgir et par antenner, hvor begge har en vinning på 2 dBi (Microchip Technology 2020b).

3.10.1 LoRa-rekkevidde

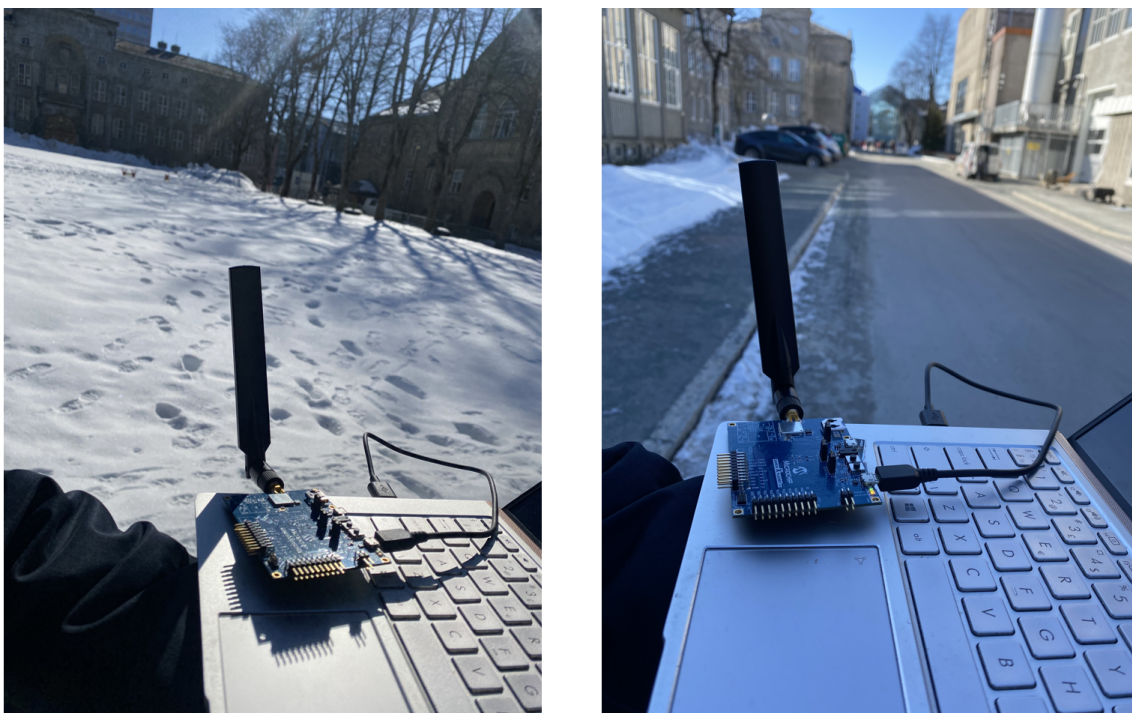
Det ble foretatt tester for å kartlegge hvorvidt utviklingskortene klarte å kommunisere med hverandre over de teoretiske rekkeviddene, se figur 72. Det var spesielt interessant å se på rekkevidden systemet skulle operere i, altså opp mot 150 meter. Første avstandstest ble gjennomført utendørs for å unngå hindringer mellom enhetene. Testene var på campus Gløshaugen, en på tvers av “Frimerket”, gressplen bak hovedbygget, som utgjorde 90 meter, og en rett strekning på 150 meter, se figur 70 og 71.



Figur 70: Testing av LoRa over “Frimerket” på Gløshaugen, med en rekkevidde på 90 meter (*Finn kart 2023*).



Figur 71: Testing av LoRa på Gløshaugen, med en rekkevidde på 150 meter (*Finn kart 2023*).



Figur 72: Rekkeviddetest over “Frimerket” t.v og strekning på Gløshaugen t.h.

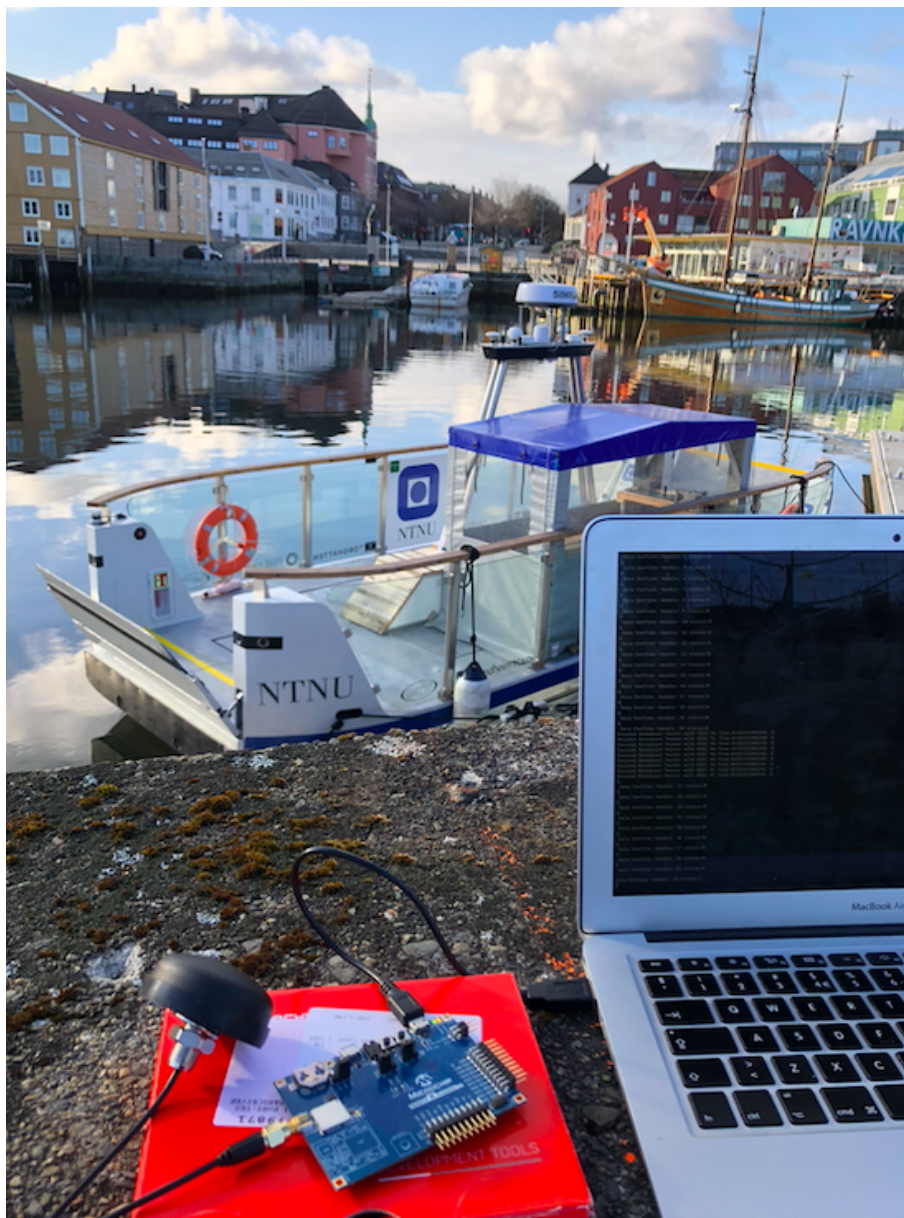
I startfasen av prosjektet ble enhetene programmert til omtrent lavest mulig innstillinger for å kontrollere om resultatene fra linkbudsjettet stemte i praksis. Følgende parametere ble brukt:

$$P_T = 1 \text{ dBm},$$
$$SP = 7,$$
$$BW = 125 \text{ kHz}.$$

Spredningsfaktoren ble satt til 7 for å sende data raskest mulig, i henhold til informasjonen gitt i seksjon 2.1.2. Rekkevidden over kanalen er liten og gruppen så på det som fordelaktig at det er rask overførsel av data og at senderen okkuperer kanalene så lite som mulig for å tillate flere sendinger i henhold til direktiver, se seksjon 3.3.1. Båndbredden er valgt til 125 kHz for å unngå opphopning av data, som er utledet i seksjon 2.1.2. Det gir et mer brukervennlig system, ettersom data overføres raskt og pålitelig.

På strekningen over 150 meter var sender effekten satt til 8 dBm som ga tilstrekkelig signalnivå for å sende og motta data. Det ble også testet et tilfelle der den ene personen plasserte seg slik at kroppen sto mellom nodene og da skygget banen til radiosignalet. Det førte til at enheten ikke lenger var i stand til å motta data.

Gruppen gjennomførte også en avstandstest mellom Ranvkloa og Vestre Kanal-kai, se figur 73, for å kartlegge kommunikasjon over strekningen der systemet skal implementeres. Signalnivået fra senderne var ved første forsøk satt til 1 dBm, og senere oppjustert til 8 og 10 dBm, for å undersøke styrken på signalet ved de ulike utgangs-signalnivåene. Det ga ulike signalstyrker ved mottaker, og la grunnlag for hvilke innstillinger enhetene skulle programmeres med i sluttproduktet. Gruppen benyttet Delta 5A og en ANT 8/9 antenne under testen.



Figur 73: Rekkevidde- og signalstyrketest mellom Ravnkloa og Vestre Kanalkai.

De mottatte signalstyrkene ble representert av programvaren som heksadesimale verdier. For å gjøre om de heksadesimale verdiene til RSSI i dBm, finner man først den binære verdien og videre toerkomplementet av den for deretter å konvertere toerkomplementet til desimal verdi, se beskrivelse og eksempel i vedlegg F.

3.10.2 Trådløs og seriell signalflyt

Det ble gjennomført en rekke tester som tok for seg signalflyten i systemet. Det vil si hele flyten fra SAM R34 mottar et signal, sender det trådløst til en annen enhet som videresender signalet serielt til en enhet hvor signalet skrives ut på skjerm.

De første testene gikk ut på å sende et trådløst signal mellom enhetene ved bruk av

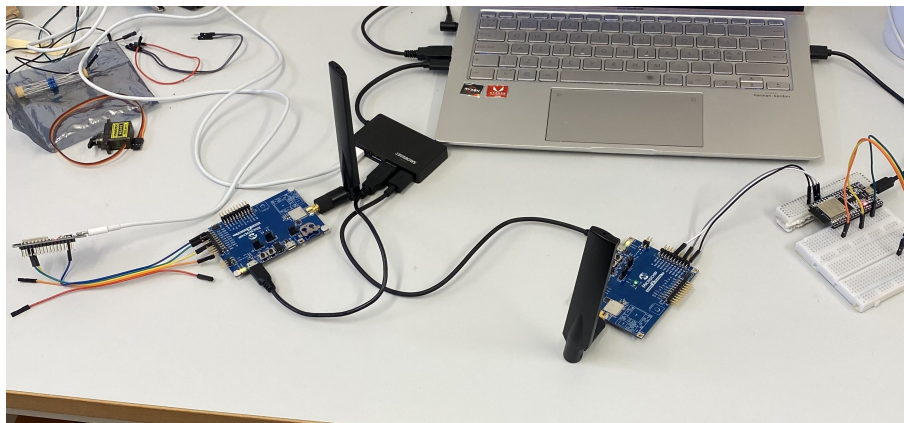
eksempelkoden som forelå, se seksjon 3.3.1. Videre utviklet gruppen egen funksjonalitet for å minimere programmet til det som faktisk var nødvendig for systemet, og la til funksjonaliteten steg for steg. For hvert steg ble trådløs kommunikasjonen testet på nytt slik at det hele tiden var klart at programmet fungerte etter sin hensikt.

Gruppen begynte senere å implementere LoRa og UART sammen. Programmeringen besto av at SAM R34 skulle ta inn et UART-signal fra en ekstern mikrokontroller, ESP32, og sende signalet trådløst til den andre enheten, som igjen skulle videregjøre LoRa-signalet med UART til en annen ESP32, se figur 74. Oppsettet illustreres i figur 75.



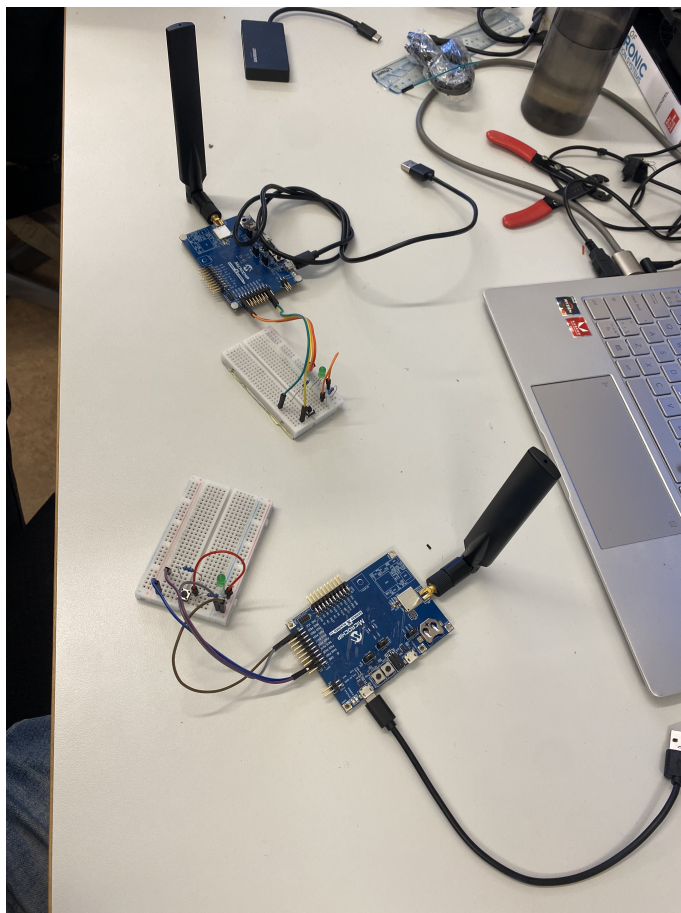
Figur 74: Signalflyt i testen.

Den siste ESP32-en visualiserte det mottatte signalet i seriell monitor, slik at det var mulig å se om det mottatte signalet var det samme som ble sendt. ESP32 er en liten mikrokontroller og fungerte som et alternativ til Autopax-maskinen i testene, ettersom de har mulighet til å motta og sende data med UART, på tilsvarende linje som Autopax-maskinen (Espressif 2023).



Figur 75: Signalflyt test fra ESP32 til ESP32 via SAM R34 og trådløst medium.

Videre testing gikk ut på å koble opp en ekstern trykk-knapp for å simulere en bestilling, som illustreres i figur 76. Programmet ble satt opp til å sende et LoRa-signal når kortet detekterte et knappetrykk, og på samme måte som tidligere, visualisere signalet i seriell monitor ved mottaker. I tillegg koblet gruppen opp en LED som skulle lyse og slukkes i henhold til beskrivelsen gitt i seksjon 3.3.4.



Figur 76: Oppsett av test med ekstern trykk-knapp og LED.

Testene beskrevet over gikk kun mellom to enheter ettersom det tok tid før gruppen mottok det tredje utviklingskortet. Da leveranse av det tredje kortet forelå, ble det koblet opp et nettverk bestående av 3 enheter. De samme testene som nevnt tidligere ble gjennomført for å kartlegge at alle enhetene i nettverket mottok, sendte og behandlet signalene riktig. Flere knapper ble tilkoblet en ESP32 med den hensikt å simulere ulike signaler fra Autopax-maskinen. Disse signalene ble deretter kommunisert til modemmet via UART, som igjen hadde som oppgave å formidle de videre til terminalen. Denne tilnærmingen ble valgt på grunnlag av modemmet sitt behov for å motta og videreformidle flere signaler fra Autopax-maskinen. Testene innbar også bruk av de bestilte antennene. Enheten som representerte modemmet på fergen var tilkoblet Delta 5A, hvor de to andre enhetene, altså terminalene, benyttet ANT 8/9.

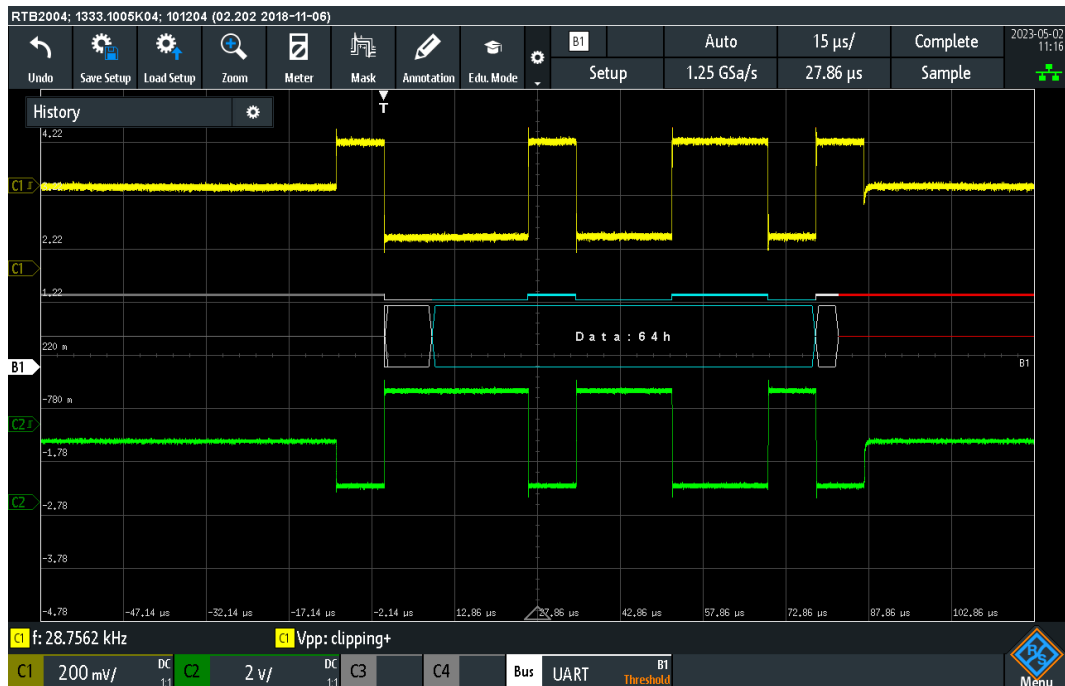
3.10.3 Signalflyt til/fra Autopax/PHS

For å undersøke signalflyten mellom terminalene og PHS, samt ferge-modemet og Autopax, ble det utført målinger av signalene. Testen ble gjennomført ved hjelp av et oscilloskop for å måle signalene og analysere bitstrømmen.

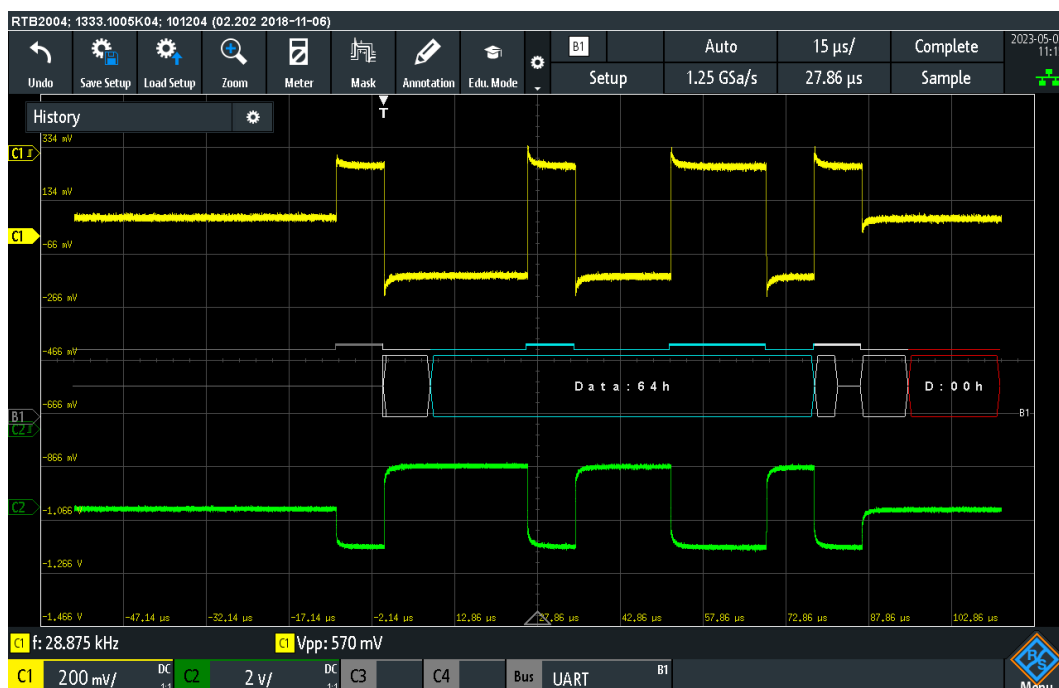
Testene omfattet både sending og mottak av data mellom nodene, der ulike typer data ble overført. I tillegg gjennomførte gruppen tester der signalflyten ble målt

under forskjellige forhold, som for eksempel med og uten felles jordingskabel mellom systemene.

Resultatene fra testene ble analysert for å vurdere kvaliteten på signalflyten mellom nodene, og for å identifisere eventuelle problemer eller begrensninger i nettverket. Basert på resultatene foretok gruppen endringer på utvidelseskortene ved å legge til en skruterterminal for å enklere koble en jordingskabel mellom PHS og terminalene. Dette på bakgrunn av at målingene ga et tydeligere signal med felles jord, se ulikhetene i figur 77 og 78.



Figur 77: UART-signal som er overført via RS485 mellom et simulert PHS og terminal med felles jord. Signalet består av en UART-ramme, representert av den blå og røde boksen, og er vist i både det originale signalet i gult og det inverterte signalet i grønt.

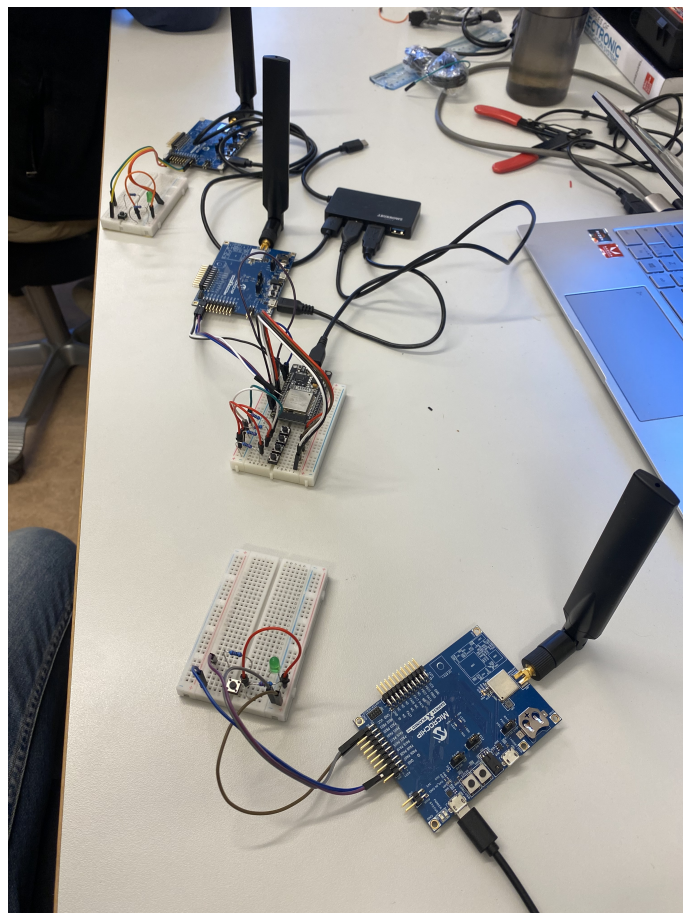


Figur 78: UART-signal som er overført via RS485 mellom et simulert PHS og terminal uten felles jord. Signalet består av en UART-ramme, representert av den blå og røde boksen, og er vist i både det originale signalet i gult og det inverterte signalet i grønt.

3.10.4 Programflyt ved bestilling av ferge

Etter at hele programflyten var programmert ble den testet for å simulere et hendelsesforløp for bestilling av fergen. Dette ble gjort med et nettverk bestående av alle tre utviklingskortene. Terminalene hadde en trykk-knapp hver, samt en LED, hvor modemmet var tilkoblet en ESP32 via UART for å simulere statussignaler fra Autopax-maskinen. Oppsettet av testen fremgår av figur 79.

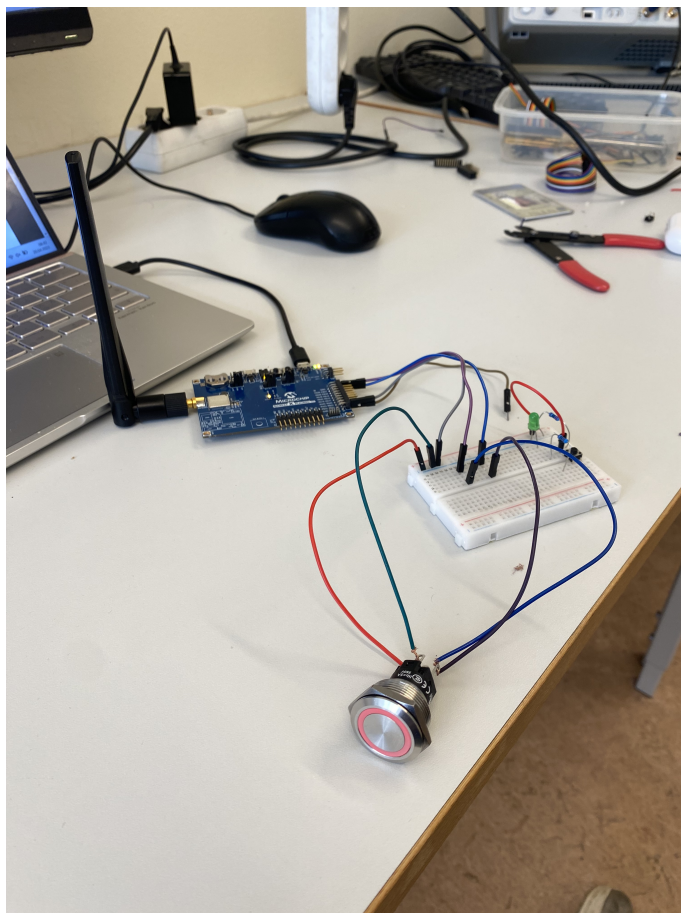
ESP32-en hadde tilkoblet fire trykk-knapper som representerte signalene "Open Gate at Terminal A", "Open Gate at Terminal B", "Close Gate at Terminal A", "Close Gate at Terminal B". Dette er signaler som er relevante for bestillingssystemet ettersom de indikerer oppdatering av fergens status og når passasjerer får beskjed om å kunne gå om bord i fergen eller må vente. I tillegg er dette kommandoer som videresendes til PHS som videre kontrollerer portene på land. Programmet skrev de sendte og mottatte signalene til skjerm for å feilsøke og sikre at programflyten stemte.



Figur 79: Oppsett av test for å simulere programflyt ved bestilling av ferge.

Bestillinger og signaler fra Autopax-maskinen ble sendt både i takt med et reelt scenario og helt tilfeldig. Det reelle scenariet var viktig for å simulere et tenkt hendelsesforløp ved bestilling. Samtidig var det nyttig å sende signaler og bestillinger tilfeldig for å kontrollere hvor robust systemet er for uforutsette påvirkninger og eventuelle signalfeil.

Den bestilte trykk-knappene ble senere implementert for å sikre at den fungerte med det oppsatte systemet, se figur 80.



Figur 80: Test av bestillingsknapp.

3.11 Simulering av bestilling av ferge

Etter at alle komponentene var mottatt, bestemte gruppen seg for å sette opp en simulering og rollespill av bestillingssystemet. Det inkluderte ferdige terminaler hvor fergen kunne bestilles og det ble gitt tilbakemeldinger på skjermen. Elektronikken som var nødvendig for å drive terminalene ble plassert inne i kabinetet, se seksjon 4.3.3. Terminalene ble forsynt med strøm ved å legge inn skjøteledning for å drive elektronikken, samtidig som det var mulig å plassere på topplokket.

På denne måten ble terminalene fullstendig operative og det var mulig å simulere bruken av systemet og kartlegge potensielle svakheter. I tillegg inviterte gruppen eksterne personer til å teste systemet og fylle ut et skjema for hvordan de opplevde bestillingen av milliAmpere2, se vedlegg G.

Terminalene ble plassert med litt avstand mellom hverandre, mens utviklingskortet som representerte modemmet på fergen var omtrent midt mellom. Modemet var koblet til en ESP32 for å simulere signaler til og fra Autopax-maskinen. Det var henholdsvis signalene som var nødvendig for bestillingssystemet som ble simulert, det vil si signalene som oppdaterer status og informasjon på skjermene. Signalflyten fulgte figur 34 og 35. En datamaskinen var koblet til den ene enden av UTP-kabelen for å simulere PHS og visualisere de mottatte signalene, men det ble ikke sendt signaler tilbake ettersom det ikke var relevant for bestillingssystemet.

Gruppen ga minimalt med beskjeder til brukerne gjennom simuleringen, da terminalene skulle gi den nødvendige informasjonen, beskrevet i seksjon 3.3.7. Ved endt simulering fikk personene beskjed om å fylle ut skjema for brukeropplevelse, se vedlegg G.

4 Resultater

4.1 Trådløs kommunikasjon mellom Ravnkloa og Vestre Kanalkai

Basert på testene utført for trådløs kommunikasjon mellom Ravnkloa og Vestre Kanalkai, som beskrevet i seksjon 3.10.1, ble det målt signalstyrker som vist i figur 81 og 82. I testprosessen ble signalet 0x51 sendt, som representert i figurene med bokstaven Q.

```
Data Confirm: Handle: 12 status:0
Secured Broadcast Packet with RSSI 93 from 042519181e599: Q
Secured Broadcast Packet with RSSI 93 from 042519181e599: Q
Secured Broadcast Packet with RSSI 94 from 042519181e599: Q
Secured Broadcast Packet with RSSI 94 from 042519181e599: Q
Secured Broadcast Packet with RSSI 94 from 042519181e599: Q
Secured Broadcast Packet with RSSI 96 from 042519181e599: Q
Secured Broadcast Packet with RSSI 94 from 042519181e599: Q
Secured Broadcast Packet with RSSI 94 from 042519181e599: Q
```

Figur 81: RSSI ved 1 dBm sendereffekt over kanalen.

```
Data Confirm: Handle: 29 status:1
Secured Broadcast Packet with RSSI 9d from 042519238b240: Q
Secured Broadcast Packet with RSSI 9e from 042519238b240: Q
Secured Broadcast Packet with RSSI 9c from 042519238b240: Q
Secured Broadcast Packet with RSSI 9c from 042519238b240: Q
Q
Data Confirm: Handle: 30 status:0
Q
Data Confirm: Handle: 31 status:0
Q
Data Confirm: Handle: 32 status:0
Q
Data Confirm: Handle: 33 status:0
Q
Data Confirm: Handle: 34 status:0
Q
Data Confirm: Handle: 35 status:0
Secured Broadcast Packet with RSSI 9f from 042519238b240: Q
Secured Broadcast Packet with RSSI a2 from 042519238b240: Q
Secured Broadcast Packet with RSSI a1 from 042519238b240: Q
Secured Broadcast Packet with RSSI 97 from 042519238b240: Q
Secured Broadcast Packet with RSSI 9c from 042519238b240: Q
Secured Broadcast Packet with RSSI a0 from 042519238b240: Q
Secured Broadcast Packet with RSSI a1 from 042519238b240: Q
Secured Broadcast Packet with RSSI a0 from 042519238b240: Q
Secured Broadcast Packet with RSSI a0 from 042519238b240: Q
Secured Broadcast Packet with RSSI a0 from 042519238b240: Q
Secured Broadcast Packet with RSSI a1 from 042519238b240: Q
Secured Broadcast Packet with RSSI a2 from 042519238b240: Q
Secured Broadcast Packet with RSSI a2 from 042519238b240: Q
Q
```

Figur 82: RSSI ved 8 og 10 dBm sendereffekt over Kanalen. De fire øverste verdiene er ved 8 dBm og de nederste er for 10 dBm.

RSSI-verdiene oppgis heksadesimalt og har tilsvarende desimal- og decibel-verdier vist i tabell 6. Utleddingen fra heksadesimalt til decibel er gjort i henhold til seksjon 3.10.1.

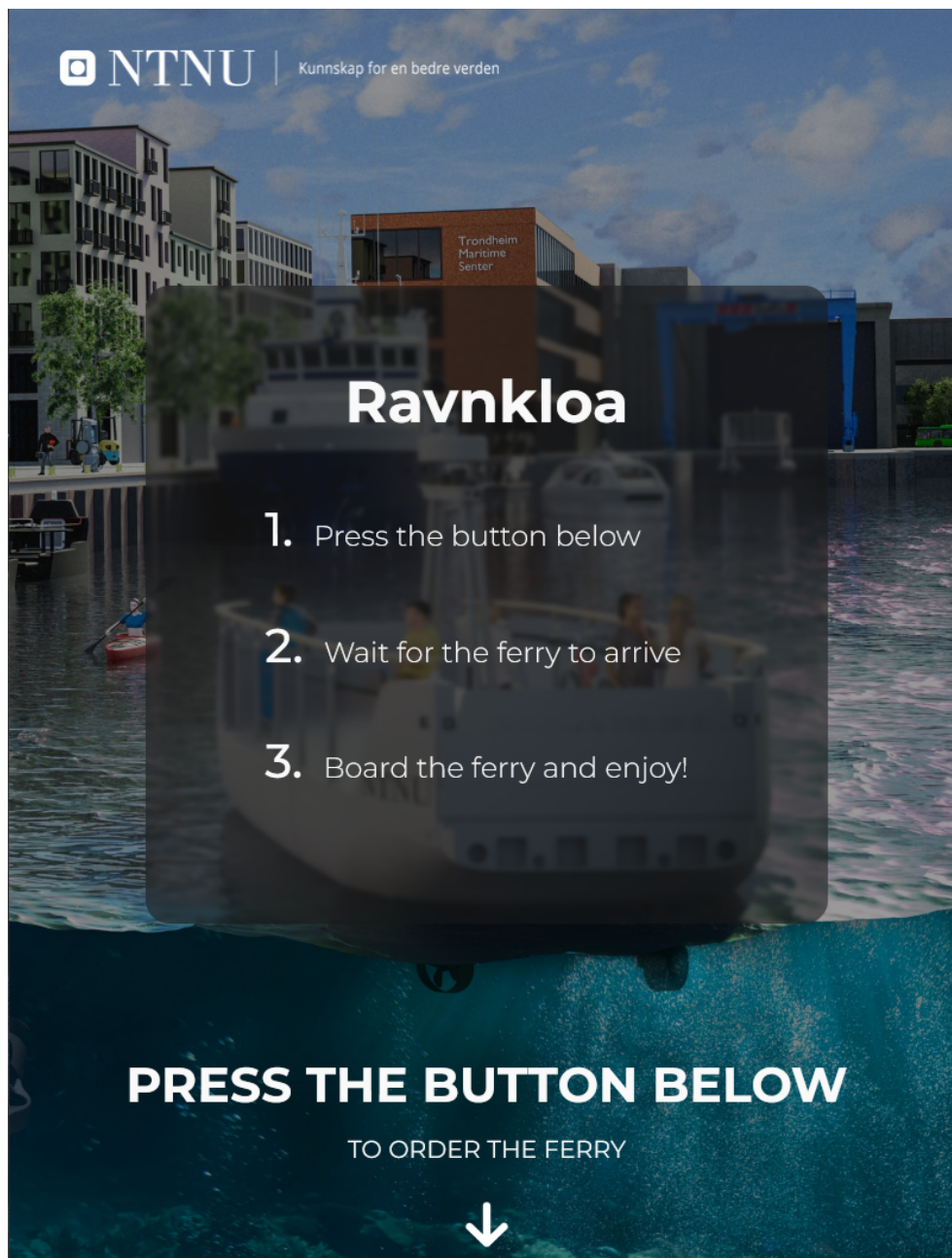
Sendereffekt [dBm]	RSSI heksadesimal	RSSI desimal	RSSI [dBm]
1	94	148	-108
8	9d	157	-99
10	a0	160	-96

Tabell 6: Mottatte signalstyrker ved 1, 8 og 10 dBm, med tilhørende heksadesimal-, desimal- og decibelverdi.

Resultatene i tabell 6 er utgangspunkt for hvilke spesifikasjoner produktet leveres med. Ettersom det oppnås tilstrekkelig signalstyrke ved 1 dBm vil det være sendernivået programmert inn og levert til oppdragsgiver, slik at nodene bruker lite effekt på å sende trådløse signaler.

4.2 Brukergrensnitt

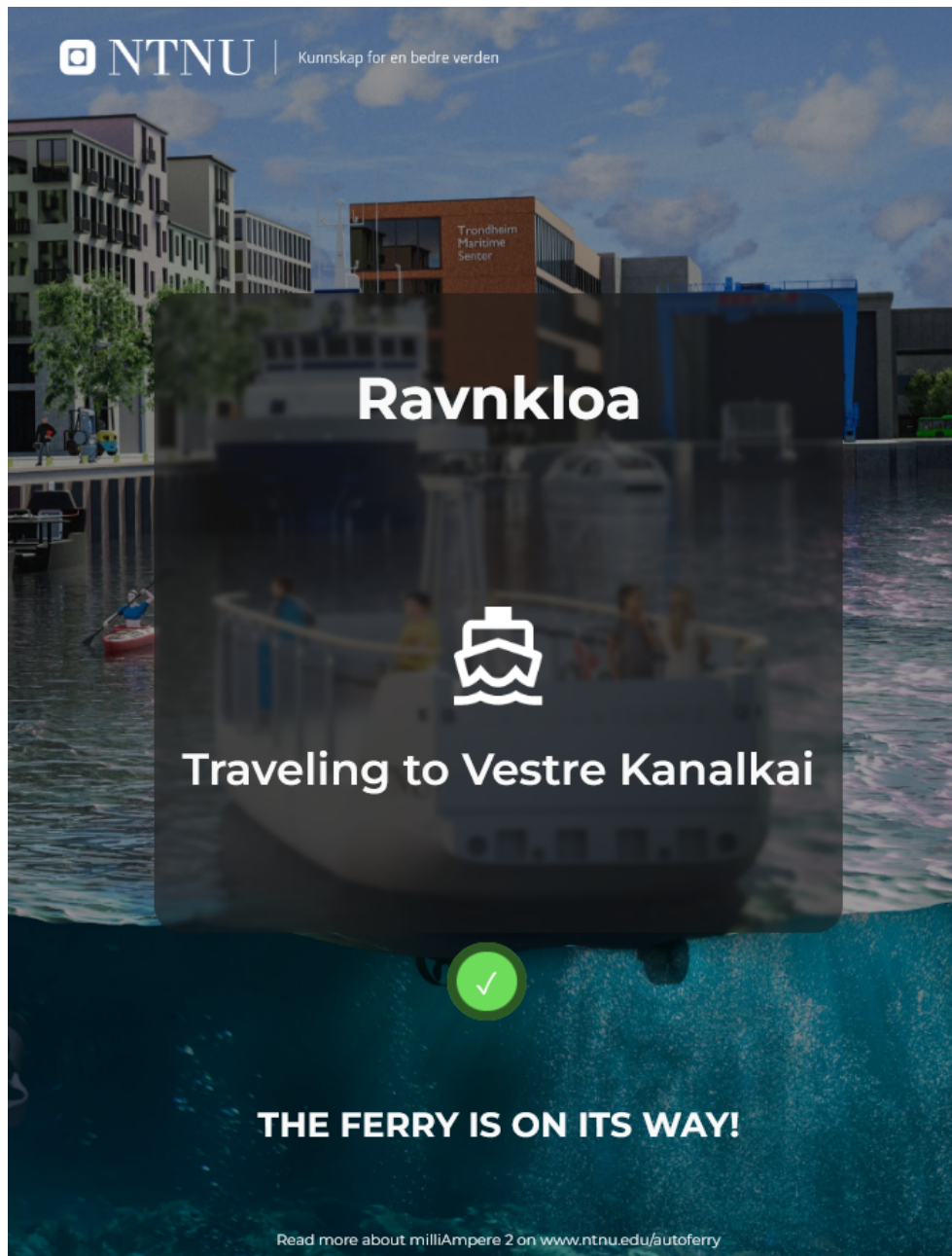
Brukergrensesnittet på terminalene består av tre deler; en bestillingsknapp, en for-sideskjerm som viser stegvis informasjon for gjennomføring av bestilling, som vist i figur 83 og en bestillingsskjerm som bekrefter bestillingen, vist i figur 84. Knappens LED-ring vil lyse når den blir trykket inn.



Figur 83: Skjerm bilde av det GUI-et som vises på skjermen før knappen er trykket.

Forsiden inneholder en kortfattet beskrivelse av hvordan terminalen brukes. På bunnen av skjermen står det “PRESS THE BUTTON BELOW”, se figur 83. Her er teksten mer fremtredende, med større skrift og tykkere bokstaver. Når bestillingsknappen trykkes vil forsidebildet på skjermen rulle oppover, og bestillingsskjermen

vil komme frem. Bestillingsskjermen viser status på fergen og en bekreftelse på at fergen er på vei, under et grønt pulserende avkrysnings-ikon for å understreke mot-tatt bestilling.



Figur 84: Skjerm bilde av det GUI-et som vises på skjermen etter at knappen er trykket.

4.3 Terminal

I henhold til seksjon 3.4 består den endelige utførelsen av terminalenes skjelett av tre hoveddeler; et frontpanel, et kabinett og en pidestall.

4.3.1 Vannsikring av terminal



Figur 85: Tørr indre del av terminalens kabinett etter gjennomført vanntest med vinylskumgummi-lister. Papir ble plassert i terminalens kabinett for å gi tydelig indikasjon dersom vann trengte inn i kabinettet.

Resultatet fra vanntesten viser at terminalen med vinylskumgummi-lister beskytter terminalen mot simulert regn i minimum ti minutter. Figur 85 viser kabinettet med tørkepapir etter gjennomført test, som beskrevet i seksjon 3.4.6.

4.3.2 Vandalsikring av terminal



Figur 86: Festeordning for montering av terminal til bakken.

I figur 86 vises foten til terminalen. Foten er konstruert slik at terminalen skal kunne stå støtt, ved hjelp av seks M8-bolter som kan skrus fast i bakken. I tillegg har foten støttevinkler sveiset fast til benet av terminalen, slik at den er mer robust for slag fra siden.



Figur 87: Festeanordning for kabinett til pedestall.

Festanordningen til kabinettet, som vist i figur 87, er designet slik at kabinettet står støtt på pedestallen og ikke lett kan rives av. Festestykket er sveiset fast i pedestallen og festet til kabinettet med fire M5-bolter.



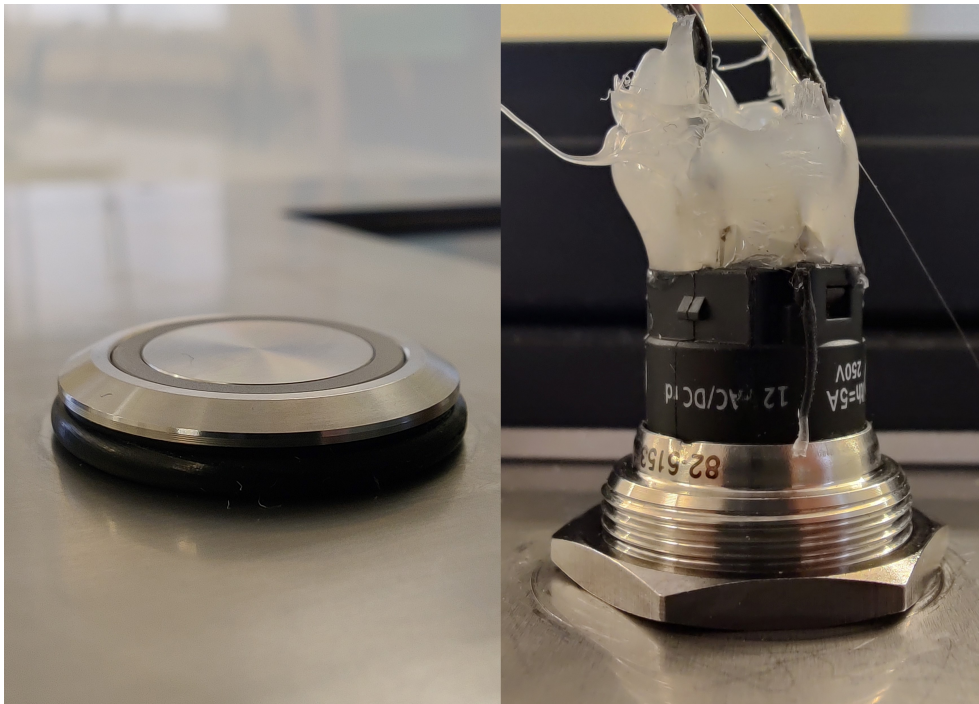
Figur 88: Festeordning for frontpanel til kabinett.

I figur 88 er det vist hvordan frontpanelet er festet. Frontpanelet er montert på kabinettet med 16 M3-skruer, slik at det ikke er enkelt å løsne frontpanelet fra kabinettet uten nødvendig verktøy.



Figur 89: Skjerm festet til frontpanel med 10 plastikk-fester.

Skjermen er festet med 10 plastikk-fester på baksiden av frontpanelet. Disse festene bidrar til at skjermen er mer robust fra støt som kan dytte skjermen inn i kabinettet. Skjermen i seg selv er sertifisert som vandalsikker, som beskrevet i seksjon 3.4.1. Figur 89 viser hvordan skjermen er festet på frontpanelet med M7-skruer.



Figur 90: Trykk-knapp festet i frontpanel på over- og undersiden.

Figur 90 viser hvordan knappen er montert på frontpanelet, i henhold til beskrivelser gitt i seksjon 3.4.5. Lim er påført knappen for å isolere og avlaste koblingspunktene.

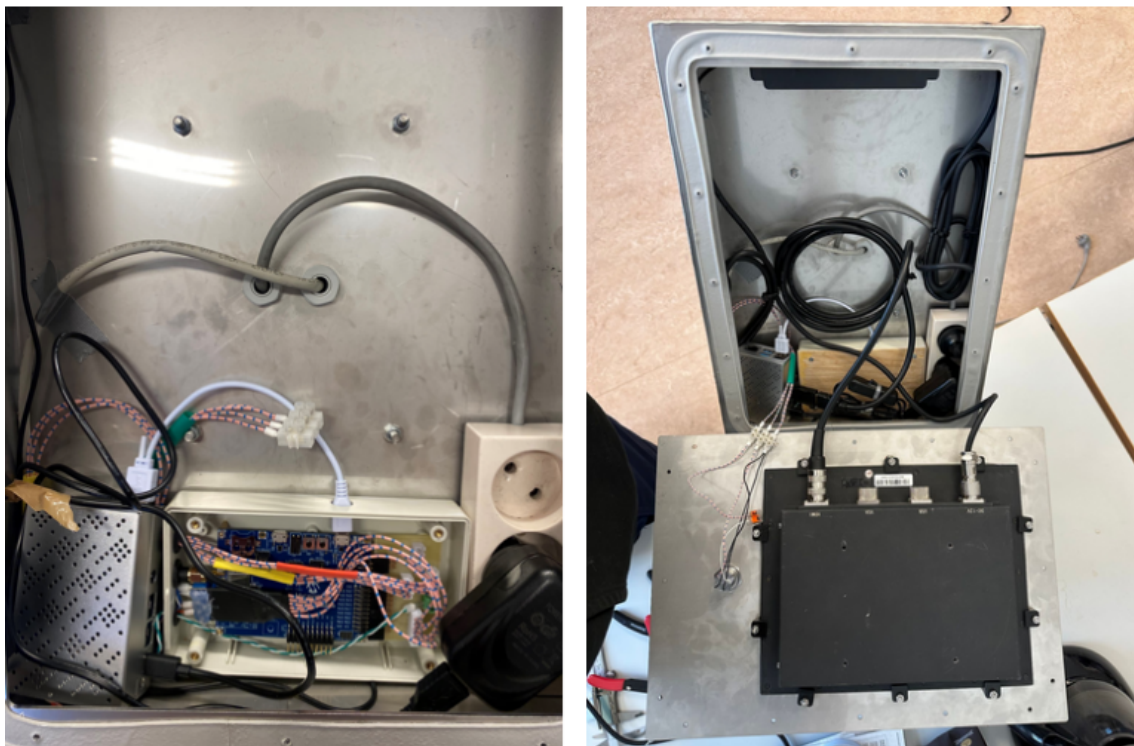


Figur 91: Antennen montert til kabinettet, vist på over- og undersiden.

I figur 91 er det vist hvordan antennen er montert. Antennen er vandalsikker og montert som beskrevet i seksjon 3.4.5.

4.3.3 Organisering av elektronikk i terminalene

Figur 92 viser hvordan elektronikken er plassert og lagt i terminalene. I terminalene forsynes SAM R34 med strøm via en USB-kabel fra RPi-en.



Figur 92: Løsning for elektronikk i terminalene. Til venstre er boksen med utviklingskortet uten lokk. Til høyre vises full oppkobling med skjerm og trykk-knapp.

4.3.4 Ferdigstilling av terminal

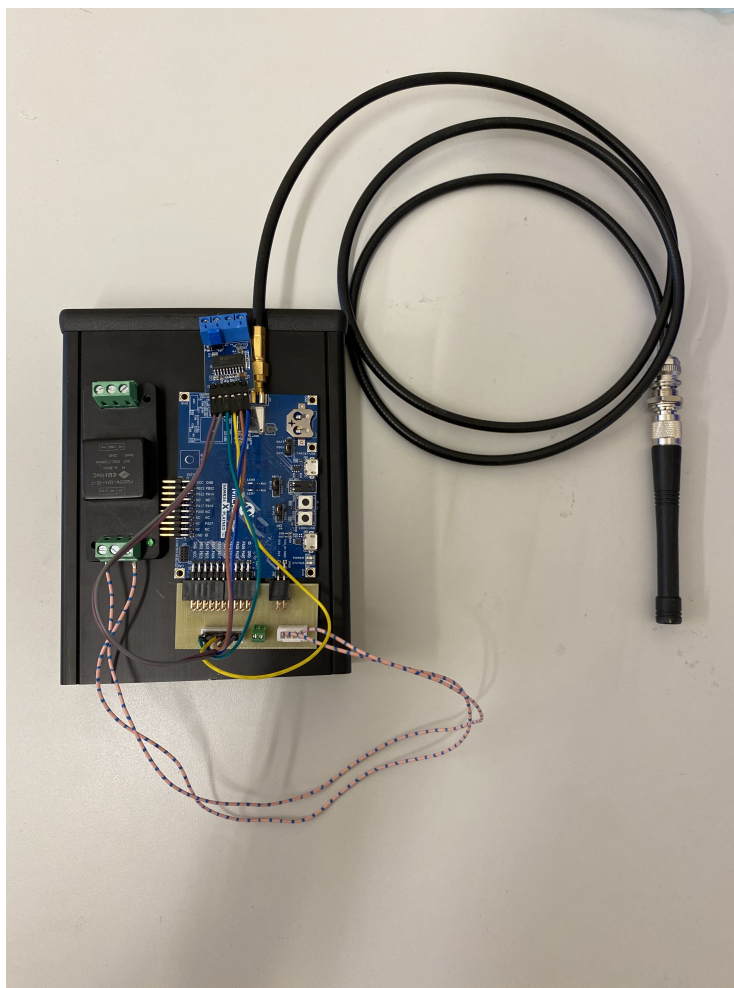
Figur 93 viser terminalen i operativ tilstand, med skjermen montert til frontpanelet, frontpanelet festet til kabinettet, og kabinettet festet til pedestallen.



Figur 93: Terminalen slik den vil fremstå for passasjerer som ankommer Ravnkloa.

4.4 Innkapsling av ferge-modem

Figur 94 viser hvordan elektronikk-løsningene er foreslått for modemet i fergen.



Figur 94: Forslag til organisering av elektronikk i innkapslingen til ferge-modemet.

4.5 Fysisk simulering og tilbakemeldinger på bestillingssystemet

Bestillingssystemet gjennomfører signalbehandling og foretar handlinger basert på kommandoene i meldingene. Bestillingsterminalene registrerer knappetrykk, samtidig som de ikke sender nye bestillinger dersom det allerede foreligger en bestilling. LED-en på knappen skruer seg på og av ut fra status på bestillingen ved den respektive terminalen.

Simuleringen av bestillingssystemet resulterte i at systemet sendte riktige signaler basert på handlinger, og oppdaterte informasjonen på skjermen slik at test-deltagere fikk oversikt over statusen på fergen. Signalene ble i tillegg sendt til PHS-systemet via UTP-kabelen og mottatt riktig.

Bildene under oppsummerer tilfredsheten til test-deltagere som var med på rollespillet av bestillingssystemet til milliAmpere2. Figur 95 og 96 forteller hvor fornøyd de er med brukeropplevelsen og tilbakemeldingene de fikk fra skjermen gjennom prosessen.

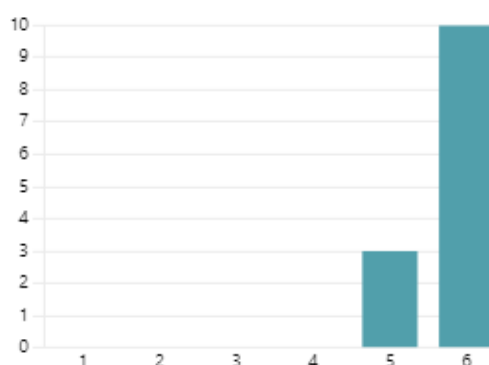
Figur 97 er en samling av tilbakemeldinger brukerne fikk skrive inn og gir en helhetlig opplevelse av bestillingen. Tilbakemeldingene gir et godt inntrykk av hvordan brukere opplevde systemet, og gir innsikt i hvilke aspekter som har potensiale for videre utvikling.

1. Hvor fornøyd er du med brukeropplevelsen? (1 = veldig lite fornøyd, 6 = veldig fornøyd)

[Flere detaljer](#)

[Innblikk](#)

5.77
Gjennomsnittlig vurdering



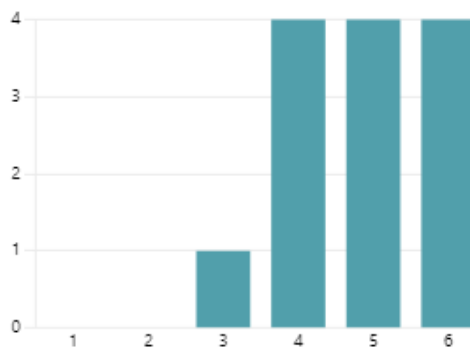
Figur 95: Statistikk som viser hvor fornøyd test-deltagere var med brukeropplevelsen under simuleringen av milliAmpere2 på en skala fra 1 til 6.

2. Hvor fornøyd er du med tilbakemeldingene du fikk på terminalene ved bestilling? (1 = veldig lite fornøyd, 6 = veldig fornøyd)

[Flere detaljer](#)

[Innblikk](#)

4.85
Gjennomsnittlig vurdering



Figur 96: Statistikk som viser hvor fornøyd test-deltagere var med tilbakemeldingene de fikk på terminalenes skjermer under simuleringen av milliAmpere2.

3. Forklar hvordan du opplevde bestillingen av milliAmpere2

13 Svar

ID ↑	Navn	Svar
1	anonymous	Superbra! Enkel å forstå og å bruke.
2	anonymous	Enkel å forstå og bruke. Veldig bra!
3	anonymous	Den var bra. Terminalen var veldig stille og bra laget. Brukergrensesnittet var fint, responsivt og hadde fine animasjoner
4	anonymous	Bra brukergrensesnitt med en litt rotete informasjonsutveksling. Kunne ha vært tydeligere beskjeder med mindre informasjon.
5	anonymous	Visste ikke ved hvilken terminal jeg var ved, så da det sto boarding visste jeg ikke om det gjaldt for meg, eller personen som hadde bestilt på andre siden.
6	anonymous	Det var tydeleg kor ferja var i forhold til meg, men det mangla informasjon om kor ferja går før du trykker på knappen.
7	anonymous	Konsis og enkel
8	anonymous	Veldig koselig, litt bugs, men ikke mer enn man burde forvente fra offentlig transport;))
9	anonymous	Helt ok;))
10	anonymous	Helt nydelig! Syns dette var bra! Står både at fergen er på vei, og at man kan boarde samtidig. Men det ga fortsatt mening.
11	anonymous	Veldig fin, og rund stang. Rart at det står boarding mens den er på vei, ellers veldig god stemning på turen;))
12	anonymous	Opplevelse det helt ok, men forsto ikke hvor fergen var til en hver tid
13	anonymous	Var forvirrende med to meldinger samtidig (at ferga er på vei og at det var boarding). Veldig fin tur med ferga!

Figur 97: Kommentarer test-deltagere hadde etter gjennomført simulering av milliAmpere2.

5 Drøfting

5.1 Vurdering av resultater

5.1.1 Trådløs kommunikasjon og signalstyrke

Det kommer tydelig frem av tester og resultater at det etableres trådløs kommunikasjon mellom ferge-modem og terminalene. Nodene i nettverket sender de forhåndsdefinerte signalene og behandler dem på riktig måte, ved å sende de til RPi og PHS/Autopax.

Tester for å kartlegge avstand og tilhørende signalstyrke viser at teori og praksis skiller seg fra hverandre, hvor signalstyrken ofte er svakere ved mottaker enn det linkbudsjettet tilsier. Det vil alltid foreligge ulikheter mellom teori og praksis. Linkbudsjettet gir fremdeles en god indikasjon på hvor pålitelig den trådløse kommunikasjonene er, men man må ta høyde for faktorer i den virkelige verden som påvirker signalstyrken, som er vanskelig å sette tall på.

Av linkbudsjettet i tabell 2 skulle det teoretisk være tilstrekkelig signalstyrke ved mottaker for å oppnå pålitelig kommunikasjon over 150 meter. Rekkeviddetesten over “Frimerket” viste derimot at kortene slet med å sende signaler når avstanden økte over 90 meter. Årsaken var mest sannsynlig at det oppsto hindringer mellom kortene, for eksempel trær og bygninger, som blokkerte signalet. Dette kalles diffraksjon og fører til at en enhet ikke mottar radiosignalet (Frenzel Jr 2023, s. 562).

Under testen på Gløshaugen på 150 meter var det fritt rom mellom nodene og de kommuniserte pålitelig med 8 dBm utgangsnivå. Som nevnt i seksjon 3.10.1, ser man hvordan kommunikasjonen brytes når det oppstår diffraksjon, ettersom radiosignalet blokkeres av personen som står mellom.

Testen gjennomført for å måle signalstyrke over kanalen viser tydelig at det foreligger pålitelig kommunikasjon ved alle forsøkene med forskjellig utgangs-signalnivå, se tabell 6.

Et interessant aspekt er RSSI-en sammenlignet med det som ble mottatt ved 150 meter over Gløshaugen. Der mottok gruppen rundt 0xa4, som tilsvarer -92 dBm RSSI, ved 1 dBm utgangsnivå, og er betydelig høyere enn alle tilfellene over kanalen. Under testene på Gløshaugen ble antennene som fulgte med utviklingskortene, som har 2 dBi vinning, brukt. Over kanalen benyttet gruppen Delta 5A og ANT 8/9, med henholdsvis 3 dBi og -6 dBi vinning. Ettersom vinningen er betydelig lavere for ANT 8/9 i forhold til standardantennene, vil det være en logisk grunn til at RSSI-en er svakere når den benyttes.

Selv om RSSI-en er svakere ved å benytte de innkjøpte antennene har det ikke utslagsgivende effekt på systemet, hvor det fremdeles er tilstrekkelig signalstyrke for å opprettholde pålitelig trådløs kommunikasjon. Det kommer av at mottatt RSSI fremdeles er høyere enn sensitiviteten til SAM R34.

Bruk av antennene som fulgte med utviklingskortene medfører derfor en usikkerhet i testene hvor de er benyttet, ettersom de ikke vil gi samme signalstyrke som de bestilte antennene. Testene er av den grunn ikke en nøyaktig representasjon av hvordan kommunikasjonen i systemet foregår, men gir en indikasjon på hvor pålitelig LoRa-protokollen er.

Det ble ikke gjennomført tester under ulike miljøforhold som regn, snø og tåke. Både regn og tåke fører til demping i radiosignalet, hvor dempingen øker med mer regn og skyer (Blaunstein og Christodoulou 2007, s. 193 og 210). Det er derfor ikke sikkert at den trådløse kommunikasjonen er robust nok til å håndtere slike situasjoner. Parametere som sendereffekt kan potensielt justeres for å opprettholde pålitelig kommunikasjon i ulike værforhold. Det vil uansett kreve videre testing for å kartlegge ulike værforhold sin påvirkning.

Det foreligger ikke resultater omhandlende signalstyrker på tvers av kanalen når hele systemet, inklusiv terminaler, er koblet opp. Det vil av den grunn være nødvendig å kartlegge signalstyrken ved fullt operativ system for å se om det skiller seg ut, og om parametere må endres for å opprettholde tilstrekkelig signalstyrke.

Det er ikke foretatt tester som prøver kommunikasjon ved større avstander, eksempelvis over 1000 meter. Det følger av at det ikke er relevant for oppgaven, ettersom kommunikasjonen ikke vil foregå over 150 meter.

5.1.2 Vanntest

Resultatet fra vanntesten, som vist i seksjon 4.3.1, viser at indre del av terminalen forble tørr i testperioden på ti minutter. Testen ble gjennomført uten at de 16 M3-skruene ble påført gjengetetning, og resultatet var bedre enn forventet. Det var i utgangspunktet, som forklart i seksjon 3.4.6, forventet at vann ville trenge inn i terminalen uten gjengetetningen.

Årsaken til at gjengetetning ikke ble påført i testen var på bakgrunn av at dette ville bli en midlertidig permanent løsning og i oppløpet mot innlevering av prosjekt var det behov for å gjøre mer arbeid knyttet til organisering av komponenter i terminalene.

Fra resultatet i seksjon 4.3.1 kan det sees små dråper med vann liggende på ytre del av vinylskumgummi-listen og kan være et produkt av uforsiktig demontering av frontpanel fra kabinett eller ujevn stramming av festeanordningen. Det ble ikke benyttet verktøy for å kontrollere kraften som ble påført skruene og det er derfor uvisst hvor mye hver skrue ble strammet. Det vil være hensiktsmessig ved videre testing å ha oversikt over hvor mye skruene bør strammes for å oppnå best mulig sikring mot fukt. Det er mindre sannsynlig at vandrdråpene som lå på listen kommer fra hullene til M3-skruene da det hverken var fukt rundt hullene eller i kabinettet.

Som beskrevet i seksjon 5.4, oppsto det forsinkelser i leveringen av terminalens skjelett. Dette førte til begrenset med tid til å gjennomføre tester for å videre forsøke å gi terminalene en tilsvarende test som ville blitt gitt ved en offisiell IP-test. Det

anbefales derfor å utføre ytterligere tester for å verifisere terminalens beskyttelse mot væske for å sikre at elektroniske komponenter i terminalen ikke påføres skade av fukt før den tas i bruk.

5.1.3 Oppkobling av elektronikk i terminalene

Det viste seg ved oppkoblingen av terminalen at det ble lite plass innvendig til alt av nødvendig utstyr. Det var i hovedsak problemer med å få plass til strømforsyningen til skjermen og tilhørende kabel, og HDMI-kabelen. Det ble akkurat nok plass til at gruppen fikk satt på og skrudd fast topplokket. Et element som mest sannsynlig vil spare plass er en annen form for skjøteledning. Løsningen som foreligger er midlertidig og tar unødvendig plass. Det vil av den grunn være hensiktsmessig å benytte seg av en elektriker eller en annen forgreiner for å oppnå en mer permanent og plassbesparende løsning.

5.1.4 Tilbakemeldinger fra brukertesten

Resultatene i seksjon 4.5 viser at brukerne i stor grad var fornøyd med bestillingssystemet, samtidig som det gir indikasjon på hvilke aspekter som kan forbedres.

Figur 97 viser tilbakemeldingene fra test-deltakerne som ble samlet inn etter endt test. Det er nødvendig å ta forbehold om at test-deltakerne fikk minimalt med informasjon ved testens start, og at testen var preget av at den ikke ble gjennomført på Ravnkloa og Vestre Kanalkai, med fergen til stede.

Et forbedringspunkt som gruppen ønsker å trekke frem er hvordan fergens rute ikke kommer tydelig frem. Selv om det for mange vil være innlysende, vil det for brukere som ikke er kjent i Trondheim være vanskelig å identifisere fergens mål. Trolig vil en bruker være skeptisk til å bruke fergen dersom en ikke vet hvor en kan gå i land igjen. Ved å fremheve fergens rute ville det også blitt lettere for brukeren å identifisere hvor en befinner seg før bestillingen, som er et annet punkt som fremkommer av tilbakemeldingene.

En subjektiv svakhet ved GUI-et er at ved bestilling, dersom fergen ankommer en terminal og videre åpner portene, endres status-teksten til boarding. Samtidig vil skjermen på dette tidspunktet vise teksten "The ferry is on its way". Enkelte test-deltakere opplevde denne informasjonen som noe motsigende, og kan oppfattes som en bug. På en annen side vil fergen i et reelt tilfelle på dette tidspunktet nettopp ha åpnet portene sine like ved terminalen, som sender et klart signal om at en kan gå om bord.

Flere av tilbakemeldingene uttrykker en positiv opplevelse og viser til at designet er tydelig å forstå og enkelt å bruke. En ting som ble positivt mottatt av test-deltakerene var terminalens tydelige og raske respons ved knappetrykk.

5.2 Produkt levert til oppdragsgiver

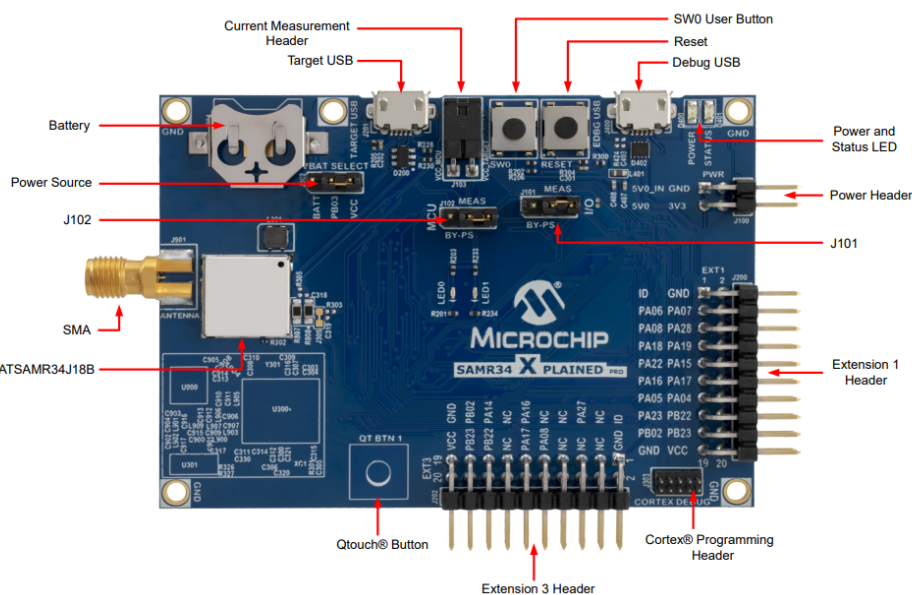
5.2.1 Backupløsning tilknyttet strømforsyning

I oppgavebeskrivelsen og forprosjektrapporten er det omtalt å tilrettelegge for bruk av reservebatteri, slik at systemet fremdeles fungerer hvis det oppstår problemer med systemets strømforsyning. En løsning som ble vurdert var en UPS (Uninterruptable Power Supply). Det er en enhet som forsyner strøm til et system i en tidsperiode, dersom hovedstrømmen kuttes (McFarlane 2022).

Hvis det skulle oppstå problemer med strømmen til terminalene vil en UPS kunne drifte systemet i en periode. Det er derimot ikke foretatt en bestilling eller implementering av UPS i produktet som leveres til oppdragsgiver. Gjennom prosjektfasen har det lenge vært et usikkerhetsmoment knyttet til hvordan terminalen og tilhørende elektronikk skal forsynes med strøm. Dette ble ikke klart før sent i prosjektfasen, og av den grunn visste ikke gruppen nøyaktige spesifikasjoner UPS-en behøvde for å fungere i produktet.

Det er i tillegg vurdert som mer hensiktsmessig å koble en UPS til PHS, og drifte terminalene ved å trekke strøm fra PHS. Det vil være av begrenset nytte å kun utstyre terminalene med UPS, ettersom et eventuelt strømbrydd vil påvirke hele systemet. Dersom PHS mister strømforsyningen, vil heller ikke terminalene og bestillingssystemet fungere som tiltenkt.

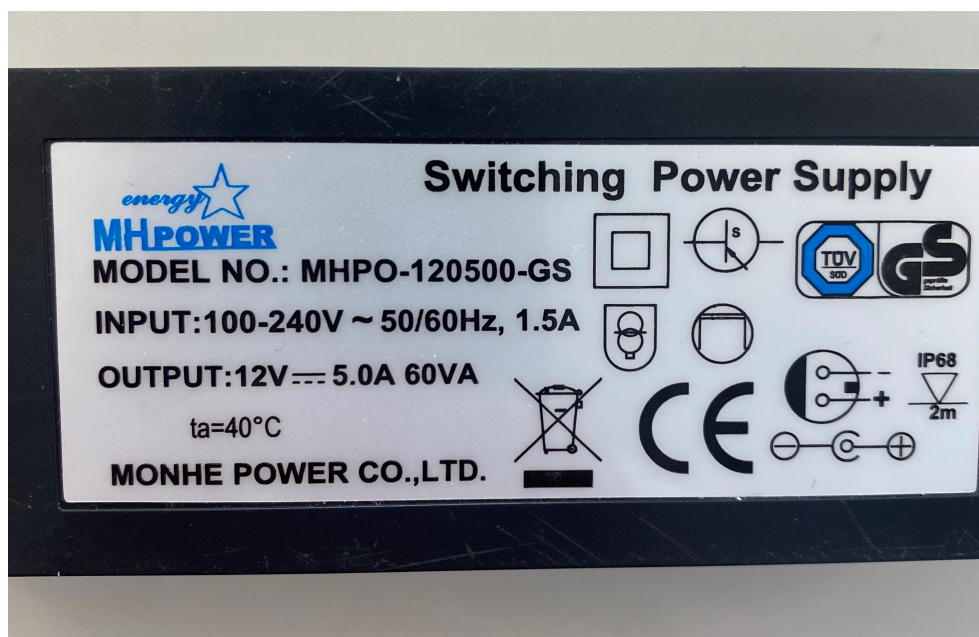
Gruppen har imidlertid utarbeidet en midlertidig backup-batteriløsning. SAM R34-modulen er utstyrt med en CR1220-batteriholder som gir 35 mAh til modulens SRAM (Static Random Access Memory) for sikkerhetskopiering (Microchip Technology 2020b), se figur 98. Det er dermed kun SRAM som mottar strøm. Microchip sitt datablad gir ingen direkte informasjon om hvilke data som lagres ved hjelp av batteristrømmen. Kortet kan derfor ikke fungere utelukkende ved hjelp av backup-batteriet. CR1220-batterier er likevel anskaffet for å sikre strømforsyning til minnet når det er nødvendig.



Figur 98: SAM R34 med batteriholder øverst i venstre hjørne (Microchip Technology 2020b).

Dette er en midlertidig løsning som ikke tilfører omfattende strømtilførsel for å drifte kortet eller hele systemet, men vil lagre minneinformasjon. Det foreslås derfor at oppdragsgiver implementerer et kraftigere backupbatteri i fremtiden, som både holder port- og bestillingssystemet i live.

Et alternativ oppdragsgiver kan gå for er en Socomec UPS (Socomec 2021). Den kan velges til forskjellige spesifikasjoner utfra hvor mye effekt systemene trekker. En 1000 W UPS kan ved 75% last, altså utstyr som trekker rundt 750 W (N1C Technologies 2018), drive systemet i 12 minutter (Socomec 2021). Det er uklart nøyaktig hvor mye bestillingssystemet trekker, men det er rimelig å anta at det ikke overstrider 70 W. Det kommer av at SAM R34 opererer på 3.3 V (Microchip Technology 2019) og trekker 16.8 mA, se figur 30, som gir effekttrekk på $16.8 \text{ mA} \cdot 3.3 \text{ V} \approx 50 \text{ mW}$. Strømforsyningen til skjermen trekker maksimalt 60 W som vist i figur 99, og gruppen går ut fra at forbruket til RPi-en ikke vil føre til en overskridelse av antagelsen på 70 W.



Figur 99: Spesifikasjoner for strømforsyningen som benyttes til skjermene på terminalene.

UPS-en vil kunne drifte bestillingssystemet en periode, men det vil foreligge begrensninger dersom UPS-en benyttes sammen med PHS. Det må av den grunn kartlegges hvor mye effekt som er nødvendig for å drifte både PHS og bestillingssystemet, og skalere UPS-en deretter.

5.2.2 Installasjon av terminaler

Når terminalene installeres og monteres der de skal driftes anbefales det å påføre gjengelås på alle skruer som går inn til terminalen for å forebygge mot at fukt trekker inn i terminalen.

5.2.3 Installasjon av modem

Årsaken til at det kun foreligger et forslag til hvordan komponentene i modemmet skal innkapsles, har bakgrunn i leveringsforsinkelser hvor den bestilte innkapslingen til modemmet tok tre uker fremfor antatte tre dager. Innkapslingen er nøye utvalgt for å kunne enkelt installeres i et "rack-kabinett" under dekk på fergen, som ble foreslått av oppdragsgiver.

Komponentene tilhørende ferge-modemet leveres av den grunn til oppdragsgiver i enkeltdele. Gruppen foreslår en løsning som innebærer modifikasjon av innkapslingen slik at en strømkabel kan gå inn til DC-DC omformeren. I tillegg må Autopax kobles via UTP og PmodRS485 inn på JST-konnektorene på utvidelseskortet. Dette krever å følge skjematene gitt i seksjon 3.7.

5.2.4 Vedlikehold av maskinvare

Gruppens forslag til vedlikehold går ut på å gjøre innkjøp av reservedeler til terminalen. Frontpanelet fungerer som vedlikeholdsluke og kan avmonteres for å bytte ut deler og gjøre endringer om nødvendig. Det anbefales også å bestille opp et par utvidelseskort for å ha i reserve, da dette kan ta lengre tid å produsere og er ikke et standardisert produkt. Tetningslister og o-ringer byttes ved behov.

5.2.5 Oppgradering av programvare

Det vil i fremtiden være lett å reprogrammere utviklingskortet i terminalen, ettersom det kun er nødvendig å koble seg inn på samme USB-port som brukes for å forsyne SAM R34 med strøm, se figur 92.

5.3 Optimering og fremtidig arbeid

Prosjektet og bestillingssystemet fører med seg flere muligheter for utvidelse, forbedringer og oppgraderinger.

5.3.1 Effektivisering av programvare

Programvaren på SAM R34 bygger på en relativt stor “switch-case” som sjekker hver melding mot de forhåndsdefinerte signalene. Dette er gjort for å kontrollisikre signalene og er beskrevet i seksjon 3.3.4.

Programvaren ble veldig lang ettersom det kom betydelig flere signaler enn det gruppen hadde sett for seg etter endringen ITK-gruppen gjorde, som først beskrevet i seksjon 5.3.4. Det vil av den grunn være mulig å videreutvikle programvaren slik at den kontrollerer signalene på en mer effektiv måte. Mulige forbedringer av programvaren kan være å implementere en datastruktur som eksempelvis en hashtabell eller et tre, som raskt kan hente ut signalinformasjonen basert på en nøkkelverdi. En annen tilnærming kan være å gruppere signalene i kategorier og bruke funksjoner til å håndtere disse kategoriene, i stedet for å sjekke hvert enkelt signal individuelt.

Det bør også gjøres en vurdering hvorvidt det er relevant at systemet skal be om å sende meldinger på nytt. Under testing førte det ofte til problemer hvor nodene konstant sendte meldinger frem og tilbake. Det var for eksempel ved en koblingsfeil eller dårlig kontakt som utløste ugjenkjenkelige meldinger. Dette kan for eksempel løses ved å legge inn funksjonalitet for meldingsbekreftelse, som diskuteres i seksjon 5.3.3, hvor noden vil sende meldingen på nytt dersom den ikke mottar meldingsbekreftelse. Reduksjon av hvilke kriterier som kreves for å sende en melding på nytt er også et alternativ, for å minimere sannsynligheten for at det oppstår konstant sending av uønskede signaler.

5.3.2 Tilfelle med mulighet for unødvendige seilas

Mot innlevering av oppgaven oppdaget gruppen et scenario som det ikke er tatt hensyn til i programvaren. Dersom fergen ankommer en terminal, men bestillingen til terminalen ligger som andre element i køen, vil ikke modemmet fjerne bestillingen. Fergen vil først fjerne bestillingen neste gang den ankommer den respektive terminalen, altså den må seile frem og tilbake før køen tømmes. Her vil det være mer hensiktsmessig at fergen fungerer på lik linje som en heis, hvor en bestilling til en holdeplass fjernes når den ankommer, uavhengig av posisjonen i køen.

Valg av FIFO viser seg derfor å ikke være hensiktsmessig for bestillingssystemet, da det ikke går an å fjerne et element som er lenger bak i køen. Det burde vurderes en struktur hvor du kan sjekke om et element er i køen og fjerne det dersom det foreligger.

5.3.3 Meldingsbekreftelse ved mottatt trådløst signal

I den nåværende trådløse systemarkitekturen og API-et mangler det en bekreftelsesfunksjonalitet som kan gi avsenderen en forsikring om at meldingen har blitt mottatt. Det kan oppstå en situasjon der to noder i nettverket sender en melding på nøyaktig samme tidspunkt som kan føre til at ingen av meldingene vil nå tenkt mottaker. Selv om sannsynligheten for at to meldinger sendes samtidig er relativt lav, vil det være fornuftig å forebygge slike situasjoner ved fremtidig arbeid. En programvareløsning som gir beskjed til avsenderen av LoRa-signalet om at meldingen er mottatt, vil derfor være hensiktsmessig. Dette vil bidra til å gjøre kommunikasjonen i systemet mer robust.

5.3.4 API

I det ferdige systemet utveksles informasjon ved bruk av et egendefinert API. API-et anvendes både for de trådløse signalene i nettverket og via UART mellom terminalene og ITK-gruppens PHS, samt mellom modemmet og Autopax. Generelt har hver melding en mottaker-ID og en avsender-ID for å avgjøre hvem meldingen er ment til og hvor meldingen kom fra. I grensesnittet ferge-modem - Autopax og terminal - PHS vil det være overflødig å ha denne type informasjon. Tilleggsinfo som mottaker-ID og avsender-ID kan derfor ved videre arbeid inkluderes/filtreres i meldingen lokalt hos terminal og ferge-modem som sender/mottar UART-rammen. Dette gjør at antall bits i rammen reduseres, som i sin tur potensielt kan redusere antall mulige bit-feil som kan oppstå ved kommunikasjon. Videre vil ekskluderingen av både mottaker- og avsender-ID gi rom for flere kommandoer. Dette vil ved mulig utvidelse av systemet gi en mer oversiktlig kommando-struktur som i sin tur kan føre til enklere og mer effektiv signalbehandling i programvaren.

En konsekvens av å øke antall kommandoer ved skalering kan by på utfordringer knyttet til blant annet trafikk i nettverket. Dersom det tidligere adresserte proble-

met knyttet til noder som sender melding samtidig ikke løses, vil sannsynligheten for at meldinger sendes samtidig fra flere noder i nettverket øke ytterligere når det implementeres flere kommandoer og flere signaler utveksles. Det samme gjelder dersom det blir aktuelt å utvide API-et til å inkludere flere bytes per melding. Det vil gi utslag på tiden senderen bruker på å sende meldingen ut på kanalen, og vil redusere antall sendinger som kan gjennomføres i løpet av en time, ut fra EU-direktivene beskrevet i seksjon 3.3.1.

Opprinnelig ble API-et utviklet som en intern protokoll innenfor LoRa-nettverket. API-et måtte omstruktureres i løpet av prosjektperioden da ITK-gruppen innså at de trengte en landbasert kontrollstasjon for personhåndtering, noe som opprinnelig var tenkt å håndteres om bord på fergen. Dette resulterte i en omstrukturering av API-et og et nært samarbeid med ITK-gruppen for å inkludere deres nødvendige signaler som kunne sendes ut på LoRa-nettverket. Mot slutten av prosjektperioden kom det forespørsler fra ITK-gruppen om å legge til flere signaler i API-et. Dette ble gjort på en måte som krevde minimal tidsbruk og endring av programvare, da tid på dette tidspunktet var en begrenset ressurs. Det er derfor viktig å understreke at API-et i sin nåværende form ikke er optimalisert, og at det finnes flere kommandoer som kunne vært kombinert.

5.3.5 Vann- og vandalsikring

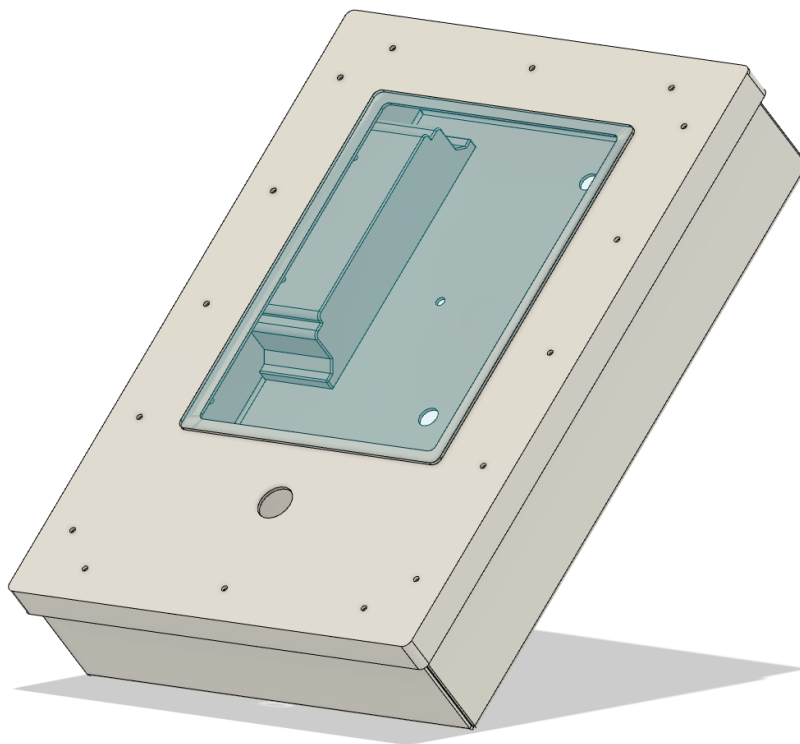
Til tross for at skjermen på terminalen er kategorisert som vandalsikkert med en IK-rangering på IK10, er den fortsatt utsatt for skade fra skarpe gjenstander og grafitti. Det kan derfor være hensiktsmessig å forbedre skjermens beskyttelsesnivå ved å legge til et ekstra lag med glass som er spesielt utviklet for å motstå slike typer hærverk.

For å videre forebygge terminalene mot vandalisme kan det være hensiktsmessig å påføre et lag med overflatebehandling mot graffiti, på eksponert stål. Dette gjør det enklere å fjerne maling og permanent tusj som ofte påføres på offentlig eiendom av enkelte grupper i samfunnet. Et eksempel på et slikt produkt er: (Sika Norge AS 2017).



Figur 100: Området på terminalene som er utsatt for vannansamling og potensiell vanninfiltrasjon.

I den nåværende utformingen av terminalene er skjermen montert på undersiden av frontpanelet. Dette gir opphav til en kant mellom skjermen og frontpanelet som kan føre til vannansamling i det aktuelle området. Selv om det er en pakning mellom skjermen og topp-platen for å hindre fuktinntrengning, er det likevel en risiko for at fukt kan trenge inn over tid dersom det samler seg vann i dette området. En mulig løsning på dette kan være å implementere løsningen beskrevet tidligere med et ekstra lag med glass som vil gå kant i kant med frontpanelet. Figur 100 viser området hvor det har oppstått vannansamling etter vanntesten, som beskrevet i seksjon 3.4.6.



Figur 101: Forslag til forbedret terminal.

For å demonstrere de ønskede forbedringene er det utviklet en 3D-modell av kabinettet, som vist i figur 101. Modellen inkluderer beskyttelsesglass samt en hette som omslutter frontpanelet. Hovedfunksjonen til hetten er å beskytte pakningen mellom frontpanelet og kabinettet mot både skarpe objekter samt direktekontakt med regn.

5.3.6 Universell utforming

I oppgavefasen ble det drøftet muligheter for implementering av universell utforming for personer med nedsatt funksjonsevne. Imidlertid ble dette ikke videreført på grunn av tidsbegrensninger.

Selv om tiltakene ikke ble gjennomført, har gruppen foreslått tiltak som kan implementeres senere. Blant de mulige funksjonene som kan innføres i senere utviklingsfaser er inkludering av et høyttaler-system for å gi hørbar tilbakemelding ved bruk av terminalen. En tydelig utfordring i denne sammenhengen er vannsikringen.

En annen mulighet som kan vurderes er inkludering av blindeskrift på frontpanelet. Dette kan forbedre opplevelsen av terminalen for blinde. En fordel med denne tilnærmingen er at det kan implementeres enkelt ved hjelp av klistremerker (Gravic Group 2023). Utfordringen kan være kompleksiteten av informasjonen som skal viderefremidles.

Det kan også være hensiktsmessig å vurdere tilgjengeligheten av terminalen for personer med fysiske funksjonshemninger, for eksempel rullestolbrukere. Det viktigste

tiltaket her er å plassere terminalen på en måte som gjør den lett tilgjengelig for alle brukere. Dette kan innebære tilpasning av høyden og avstanden til terminalen.

5.4 Prosjektgjennomføringen

Prosjektet bygger på en konsis arbeidsmengde gjennom 4 måneder, hvor det kontinuerlig er gjort vurderinger og valg slik at det foreligger et ferdig produkt ved innlevering.

Det ble aldri gjennomført tester med ITK-gruppens system på bakgrunn av at prosjektet deres endte i et konsept. Gitt forutsetningen om at ITK-gruppen hadde et forventet ferdig produkt, kreves det ytterligere arbeid for å integrere terminalene og modemmet på milliAmpere2 i fravær av ITK-gruppens system.

Gruppen opplevde forsinkelser i konstruksjonen av terminalen på grunn av sykdom på verkstedet hvor terminalen ble sveiset. Dette medførte en to ukers forsinkelse i etableringen av terminalens fundament, som igjen førte til forsinkelser i fullskala testing og produksjon av resultater knyttet til rapporten. I etterkant vurderer gruppen at det ville vært hensiktsmessig å forutse slike hendelser og lagt inn større marginer for å ta hensyn til slikt.

I integrasjon av komponenter til det ferdige systemet oppsto det tekniske utfordringer hvor det ble gjennomført intensive feilsøkningsprosesser for å kartlegge og vurdere løsninger til problemene. Dette var nødvendig for å levere et fungerende produkt innen tidsfristen. Et eksempel var at SAM R34 låste seg på en bestilling, og nye knappetrykk medførte ingen handling, selv om systemet åpnet for nye bestillinger. Forklaringen var at SAM R34 ble forsynt med strøm fra 5 V-pinnen på RPi-en. Nøyaktig forklaring er gruppen usikker på, men hypotesen er at SAM R34 ikke fikk levert tilstrekkelig strøm. Løsningen ble å koble en USB-kabel mellom RPi og SAM R34 for å forsyne utviklingskortet med strøm, som vist i figur 92, etter det registrerte SAM R34 alle nye bestillinger.

6 Konklusjon

Formålet med denne oppgaven var å lage en infrastruktur for bestilling av den autonome fergen milliAmpere2. Dette innebar å konstruere og utvikle to brukervennlige bestillingsterminaler som kunne oppnå trådløs kommunikasjon med et modem som gruppen skulle utvikle til fergen. Videre var det nødvendig at terminalene kunne samhandle med hvert sitt passasjer-håndteringssystem for styring av sikkerhetsporter på land, og modemet med en maskin om bord på fergen. Det var også nødvendig å utvikle løsninger tilpasset det operative miljøet til systemet, ved å forebygge mot vandalisme og beskytte mot miljøforhold, dvs. vann og regn.

Utformingen av terminalene tilrettelegger for en intuitiv måte å bestille fergen på, med et brukergrensesnitt bestående av en bestillingsknapp og skjerm som viser informasjon om bruk. En mindre brukertest ble gjennomført og det ble gitt mange positive tilbakemeldinger. Resultatene viste også at det grafiske brukergrensesnittet kunne forbedres ved å justere informasjonen på skjermen til å bli mer presis.

Valget av den trådløse protokollen ga pålitelig kommunikasjon over kanal-kryssingen og er videre en protokoll som kan anvendes ved skalering av systemet. Terminalene er laget med forbehold om utendørsbruk og er beskyttet mot vann og vandalisme. Det har kommet frem i gjennomførte tester at med foreslåtte justeringer ved terminalens design, ville konstruksjonen vært bedre tilpasset omgivelsene den skal operere i, spesielt med tanke på å forebygge fuktproblemer. Systemet er ikke testet sammen med passasjer-håndteringssystemet og maskinen om bord på fergen da de systemene er en del av en annen bacheloroppgave som endte med et konsept.

Det er alt i alt utviklet en bestillingsløsning som tilrettelegger for bestilling av milliAmpere2. Det er utviklet et trådløst kommunikasjonssystem mellom ferge-modem og terminalene, som i sin tur kan samhandle med passasjer-håndteringssystemet og en maskin om bord i fergen. Det er utviklet et brukervennlig og intuitivt brukergrensesnitt. Terminalenes innkapsling gir en robust konstruksjon med potensielle forbedringer knyttet til fuktsikring.

LoRa som trådløs kommunikasjonsprotokoll ga tilfredsstillende resultater og anbefales til videre bruk ved eventuell skalering av systemet. Tilbakemeldingene fra brukertesten ga nyttig informasjon knyttet til utforming av det grafiske brukergrensesnittet og det er videre å anbefale å justere den visuelle tilbakemeldingen på skjermen til å bli mer presis. Alt i alt kan det utviklede systemet bidra med verdifull data og informasjon knyttet til brukerinteraksjon, ved autonom fergetransport.

Referanser

- Alex Voxel (2023). *Color Changing Screen - Mood Led Lights [10 Hours]*. URL: <https://www.youtube.com/watch?v=JUodmkXP41s> (sjekket 14. mai 2023).
- Analog Devices (2008). *Application note: RS-485/RS-422 Circuit Implementation Guide*. URL: <https://www.analog.com/media/en/technical-documentation/application-notes/an-960.pdf> (sjekket 17. apr. 2023).
- Angled Female Pin Header* (2023). URL: <https://www.technobotsonline.com/female-pcb-header-2x10-right-angle.html>.
- Atmel (2012). *Atmel AVR4030: Atmel Software Framework - Reference Manual*. URL: <http://ww1.microchip.com/downloads/en/AppNotes/doc8432.pdf>.
- Bachelorgruppe E2318 (2023). *Wireless network ordering milliAmpere2 programvare*. URL: https://github.com/Ocheid/Wireless_network_ordering_milliAmpere2.
- BELDEN (2022). *50 Ohm Wireless Transmission Coax, RG-174*. URL: https://no.mouser.com/datasheet/2/46/8216_techdata-2324139.pdf (sjekket 29. apr. 2023).
- Berg, E. mfl. (2023). «Passasjerhåndteringssystem for den autonome urbane passasjerfergen milliAmpere2». Bachelor's Thesis. Norwegian University of Science og Technology.
- Bhargava, M. (2017). *IOT Projects with Bluetooth Low Energy*. Packt Publishing Ltd., s. 123.
- Blaunstein, N. og C.G. Christodoulou (2007). *Radio Propagation and Adaptive Antennas for Wireless Communication Links*. WILEY.
- Brooks, D. R. (2007). *An Introduction to HTML and JavaScript*. Springer, s. 3.
- BUD Industries Inc. (2023). *NEMA vs. IP Enclosure Protection Ratings*. URL: <https://www.budind.com/nema-vs-ip-protection-ratings/> (sjekket 13. mai 2023).
- Bugl, D. (2019). *Learn React Hooks*. Packt Publishing Ltd., s. 8.
- Cormen, T. H. mfl. (2009). *Introduction to algorithms*. 3. utg. Mit Press, s. 232–235.
- CUI (2022). *PQDE6W-T DC-DC datasheet*. URL: <https://www.cui.com/product/resource/pqde6w-t.pdf> (sjekket 8. mar. 2023).
- Dictionary Data Structure* (2023). URL: javascript.plainenglish.io/how-to-clone-an-app-from-github-446541a0302d (sjekket 10. mai 2023).
- Digilent (2021). *PmodRS485 Reference Manual*. URL: https://digilent.com/reference/_media/reference/pmod/pmodrs485/pmodrs485-rm.pdf (sjekket 29. apr. 2023).
- EAO (2016). *Illuminated pushbutton 22mm stainless-steel, datasheet*. URL: <https://docs.rs-online.com/63d5/0900766b81559356.pdf> (sjekket 29. apr. 2023).
- EasyEDA (2022). *EasyEDA Online PDC Design Tool*. URL: <https://easyeda.com/>.
- Elprocus (2023). *Different Microcontroller Boards and their Applications*. URL: <https://www.elprocus.com/different-types-of-microcontroller-boards/> (sjekket 24. apr. 2022).
- Espressif (2023). *ESP32 Documentation*. URL: <https://www.espressif.com/en/products/socs/esp32> (sjekket 6. mai 2023).

- EX-1309-T* (2023). URL: <https://www.distrelec.biz/en/usb-serial-converter-rs232-rs422-rs485-terminal-block-exsys-ex-1309/p/30082283>.
- Exsys (2019). *EX-1309-T RS485 to USB adapter manual*. URL: https://cdn-reichert.de/documents/datenblatt/E910/EXSYS_EX1309T_MN_ENG.pdf.
- Faculty of Information Technology and Electrical Engineering NTNU's photos (2021). *milliAmpere2 bilde flickr*. URL: <https://www.flickr.com/photos/tags/milliAmpere2/>.
- Fang, Yi-yuan og Xue-jun Chen (2011). «Design and Simulation of UART Serial Communication Module Based on VHDL». I: s. 1–4. DOI: [10.1109/ISA.2011.5873448](https://doi.org/10.1109/ISA.2011.5873448).
- Finn kart* (2023). URL: <https://kart.finn.no/>.
- Floyd, T.L. (2012). *Electronic devices*. Ninth edition. New Jersey: Prentive Hall.
- Frenzel Jr, L.E. (2023). *Electronic communication systems*. Fifth edition. New York: McGraw Hill LLC.
- G. R. Kanagachidambaresan Kolla Bhanu Prakash, V. Mahima (2021). «Programming Tensor Flow with Single Board Computers». I: DOI: https://doi.org/10.1007/978-3-030-57077-4_12.
- Gardner, Leroy (2005). «The use of stainless steel in structures». I: *Progress in Structural Engineering and Materials* 7.2, s. 45–55. DOI: [10.1002/pse.190](https://doi.org/10.1002/pse.190).
- Gravic Group (2023). *Braille Labels*. URL: <https://gravicgroup.com/en/printed-and-cut-parts/adhesive-labels/braille-labels/> (sjekket 19. mai 2023).
- Gulotta, F. (2022). *About SerialPort*. URL: <https://serialport.io/docs/> (sjekket 8. mai 2023).
- Hareide Design Norway* (2023). URL: <https://hareidedesign.com/> (sjekket 9. mai 2023).
- Haugan, I. (2022). *Verdens første prøvedrift av selvkjørende passasjerferge i by starter opp i Trondheim*. URL: <https://forskning.no/data-miljoteknologi-ntnu/verdens-forste-provedrift-av-selvkjorende-passasjerferge-i-by-starter-opp-i-trondheim/2081327> (sjekket 28. apr. 2023).
- Hoperf (2006). *RFM95/96/97/98(W) - Low Power Long Range Transceiver Module*. URL: https://www.rfsolutions.co.uk/downloads/1463993415RFM95_96_97_98W.pdf (sjekket 30. jan. 2023).
- IBM (2020). *What is an API?* URL: <https://www.ibm.com/topics/api> (sjekket 29. apr. 2023).
- IEC (2022). *IP ratings*. URL: <https://www.iec.ch/ip-ratings> (sjekket 3. apr. 2022).
- InfiNet Wireless (2023). *Wireless Networking Fundamentals*. URL: <https://academy.infinetwireless.com/en/online-education/wireless-networking-fundamentals/8> (sjekket 11. mai 2023).
- Interelectronix (2022). *IK-slagfasthet dokumentasjon*. URL: <https://www.interelectronix.com/no/en-62262-ik-stotmotstandsgrad.html> (sjekket 3. apr. 2022).
- JST (2023). *JST XH Connector manual*. URL: <https://docs.rs-online.com/f25e/0900766b81357f2f.pdf> (sjekket 5. mai 2023).

- JST konnektor* (2023). URL: <https://www.tinytronics.nl/shop/en/cables-and-connectors/cables-and-adapters/jst-compatible/jst-xh-6p-compatible-connector-with-cable-and-pcb-connector>.
- Kjell & Company (2023). *Raspberry Pi 4 Model B Ettkortsdatamaskin 4 GB RAM*. URL: <https://www.kjell.com/no/produkter/data/ettkortsdatamaskin/raspberry-pi-4-model-b-ettkortsdatamaskin-4-gb-ram-p88181> (sjekket 11. mai 2023).
- Kurose, James F. og Keith W. Ross (2013). *Protocol Layers and Their Service Models*. Pearson, s. 47–55.
- Linx Technologies (2021). *ANT-8/9-SPS1 Series datasheet*. URL: <https://docs.rs-online.com/ddd6/A700000008249447.pdf> (sjekket 28. mar. 2023).
- Mazidi, M. A., J. G. Mazidi og R. D. McKinlay (2006). *The 8051 microcontroller and embedded systems*. Pearson.
- McFarlane, R. (2022). *uninterruptible power supply (UPS)*. URL: <https://www.techtarget.com/searchdatacenter/definition/uninterruptible-power-supply> (sjekket 1. mai 2023).
- Meaney, F. (2023). *LORA AND LORAWAN TIMING*. URL: <https://ecstxal.com/lora-lorawan-timing/> (sjekket 25. apr. 2023).
- Microchip Technology (2019). *SAM R34/R35 Low Power LoRa® Sub-GHz SiP Datasheet*. URL: <https://ww1.microchip.com/downloads/aemDocuments/documents/OTH/ProductDocuments/DataSheets/SAM-R34-R35-Low-Power-LoRa-Sub-GHz-SiP-Data-Sheet-DS70005356C.pdf> (sjekket 27. jan. 2023).
- (2020a). *SAM L21 Family Data Sheet*. URL: https://ww1.microchip.com/downloads/aemDocuments/documents/MCU32/ProductDocuments/DataSheets/SAM_L21_Family_DataSheet_DS60001477C.pdf.
- (2020b). *SAM R34 Xplained Pro User Guide*. URL: <https://ww1.microchip.com/downloads/en/DeviceDoc/SAM-R34-Xplained-Pro-User-Guide-DS50002803D.pdf> (sjekket 27. jan. 2023).
- (2021a). *ATSAMR34 LONG RANGE P2P manual*. URL: https://github.com/MicrochipTech/atsamr34_long_range_p2p (sjekket 27. jan. 2023).
- (2021b). *SAM R34 Learning Phase manual*. URL: https://github.com/MicrochipTech/atsamr34_long_range_p2p/tree/master/01_learning_phase (sjekket 27. jan. 2023).
- (2021c). *SAM R34 Real-life Use Case Examples manual*. URL: https://github.com/MicrochipTech/atsamr34_long_range_p2p/tree/master/03_customize (sjekket 27. jan. 2023).
- (2022). *Microchip Studio for AVR® and SAM Devices*. URL: <https://www.microchip.com/en-us/tools-resources/develop/microchip-studio> (sjekket 25. apr. 2022).
- Multicomp (2020). *RG223 Coaxial Cable*. URL: <https://www.farnell.com/datasheets/2921248.pdf> (sjekket 2. feb. 2023).
- My Digital Portofolio. By - Anders Kristensen* (2023). URL: <https://anderszk.github.io/> (sjekket 10. mai 2023).

- N1C Technologies (2018). *3 Practical Tips to Figure Out the Size Requirements of UPS System*. URL: <https://lithium-ion-ups.com/sizing-ups-tips/> (sjekket 8. mai 2023).
- Norwegian Schitech New (2023). *NTNU trials world's first urban autonomous passenger ferry*. URL: <https://norwegianscitechnews.com/2022/09/ntnu-trials-worlds-first-urban-autonomous-passenger-ferry/> (sjekket 20. mai 2023).
- Perez, R. (1998). *Isotropic Antenna*. URL: <https://www.sciencedirect.com/topics/engineering/isotropic-antenna> (sjekket 28. mar. 2023).
- PmodRS485* (2023). URL: <https://no.rs-online.com/web/p/communication-wireless-development-tools/1346474>.
- Radio Link Budget: details & formula* (2023). URL: <https://www.electronics-notes.com/articles/antennas-propagation/propagation-overview/radio-link-budget-formula-calculator.php> (sjekket 27. mar. 2023).
- Raspberry Pi (2023). *Frequency Management and Thermal Control*. URL: <https://www.raspberrypi.com/documentation/computers/raspberry-pi.html#frequency-management-and-thermal-control> (sjekket 14. mai 2023).
- React Community* (2023). URL: <https://react.dev/community> (sjekket 9. mai 2023).
- RS Pro (2023). *RS PRO Cable Gland Datasheet*. URL: <https://docs.rs-online.com/5149/A700000007815523.pdf> (sjekket 9. mai 2023).
- Semtech (2020). *SX1276 - 137 MHz to 1020 MHz Low Power Long Range Transceiver datasheet*. URL: https://semtech.my.salesforce.com/sfc/p/#E0000000JelG/a/2R0000001Rbr/6EfVZUorrpoKFvvaF_Fkpgp5kzjiNyiAbqcpqh9qSjE.
- (2023a). *LoRa (PHY)*. URL: <https://www.semtech.com/lora/what-is-lora> (sjekket 12. mai 2023).
- (2023b). *What are LoRa® and LoRaWAN®?* URL: <https://lora-developers.semtech.com/documentation/tech-papers-and-guides/lora-and-lorawan/> (sjekket 27. mar. 2023).
- Shah, D. (2018). *Introduction to JavaScript Object Notation*. BPB Publications.
- Sihovison (2023). *Datablad: Aluminum alloy marine displays 1000 nits ip67 133 inch lcd monitor*. URL: https://www.alibaba.com/product-detail/Aluminum-alloy-marine-displays-1000-nits_1600101712893.html?spm=a2700.details.0.0.5f5c7ca2jHf13h (sjekket 14. mai 2023).
- Sika Norge AS (2017). *Antigrafittmiddel for stål og betong*. URL: https://nor.sika.com/dms/getdocument.get/5a6738a3-cfa6-36ed-9f04-87c2a7e5d018/PDB_Antigrafitti_15.pdf.
- Siretta (2023). *Delta 5A antenna specifications*. URL: <https://www.siretta.com/products/antennas/delta-5a/> (sjekket 27. apr. 2023).
- Skruterminal* (2023). URL: <https://nor.grandado.com/products/20-stk-mye-kf350-3-5-2p-kf350-3p-terminal-300v-10a-skru-2-pin-3-pin-3-5mm-rett-pin-pcb-skru-terminal-terminal-24-18-awg>.

- SMART Modular Technologies (2022). *Peer-to-peer communication in LoRa®*. URL: <https://smartmodular.com.br/en/blog/peer-to-peer-communication-in-lora/> (sjekket 27. mar. 2023).
- Socomec (2021). *Reliable and versatile power protection*. URL: <https://docs.rs-online.com/ab06/A700000008283405.pdf>.
- Sponås, J.G. (2023). *Things You Should Know About Bluetooth Range*. URL: <https://blog.nordicsemi.com/getconnected/things-you-should-know-about-bluetooth-range> (sjekket 3. apr. 2023).
- Start a New React Project* (2023). URL: <https://react.dev/learn/start-a-new-react-project> (sjekket 9. mai 2023).
- T., Kugelstad (2021). *The RS-485 Design Guide - Texas Instruments*. URL: https://www.ti.com/lit/an/slla272d/slla272d.pdf?ts=1682750002935&ref_url=https%5C%253A%5C%252F%5C%252Fwww.google.com%5C%252F (sjekket 29. apr. 2023).
- TAOGLAS (2023). *Roadmarker Antenna for 868 MHz*. URL: <https://www.taoglas.com/product/road-marker-antenna-868mhz-adhesive/> (sjekket 1. feb. 2023).
- Tata Communications (2022). *A Guide to Cellular IoT*. URL: <https://www.tatacommunications.com/solutions/mobility-iot/cellular-iot-enablement/> (sjekket 3. apr. 2022).
- Texas Instruments (2016). *Full Duplex RS-485 Over Two Wires Reference Design*. URL: https://www.ti.com/lit/ug/tidubh8/tidubh8.pdf?ts=1681722542985&ref_url=https%5C%253A%5C%252F%5C%252Fwww.google.com%5C%252F (sjekket 17. apr. 2023).
- The Things Network (2023a). *Spreading Factors*. URL: <https://www.thethingsnetwork.org/docs/lorawan/spreading-factors/> (sjekket 27. mar. 2023).
- (2023b). *What are LoRa and LoRaWAN?* URL: <https://www.thethingsnetwork.org/docs/lorawan/what-is-lorawan/> (sjekket 27. mar. 2023).
- Understanding Antenna Gain, Beamwidth, And Directivity* (2023). URL: <https://www.ahsystems.com/articles/Understanding-antenna-gain-beamwidth-directivity.php> (sjekket 11. mai 2023).
- Using the Effect Hook* (2023). URL: <https://legacy.reactjs.org/docs/hooks-effect.html> (sjekket 4. mai 2023).
- Using the State Hook* (2023). URL: <https://legacy.reactjs.org/docs/hooks-state.html> (sjekket 4. mai 2023).
- What Socket.IO is* (2023). URL: <https://socket.io/docs/v4/> (sjekket 8. mai 2023).
- Wight (2018). *Metrix Conversions*. URL: <https://www.metric-conversions.org/length/feet-to-meters.htm> (sjekket 29. apr. 2023).
- Zeabuz (2023). *Zeabuz: Who we are*. URL: <https://www.zeabuz.com/> (sjekket 14. mai 2023).
- Zeam (2023). *Electric and self-driven ferry in Stockholm*. URL: <https://zeam.se/> (sjekket 14. mai 2023).

A AFT - API

Milliampere2 AutonomFergeTerminal - API (AFT - 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.

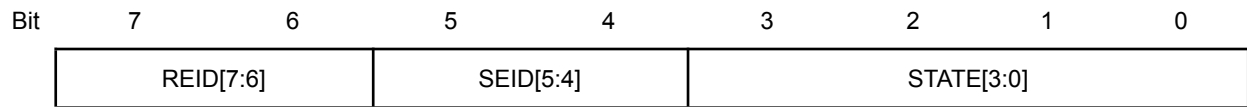
Sammendrag av kommando-identifikatorer

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:]	
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	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
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	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
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	5	4	3	2	1	0
	REID[7:6]		SEID[5:4]		MANUALSTP[3:0]			

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	5	4	3	2	1	0
	REID[7:6]		SEID[5:4]		CMD[3:0]			

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 5 4 3 2 1 0

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

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

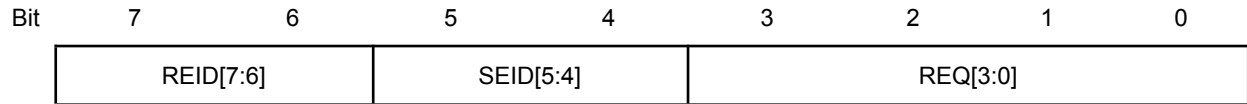
Dette bit-feltet definerer en bestilling av fergen.

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

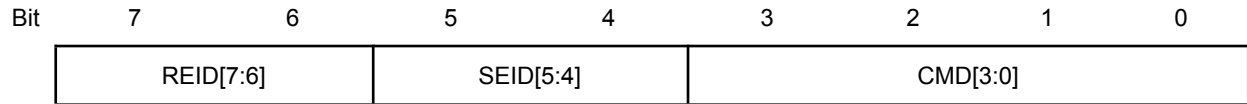
Dette bit-feltet definerer en bestilling av fergen.

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	5	4	3	2	1	0
	REID[7:6]		SEID[5:4]		CONFIRM[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 - 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	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] 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	5	4	3	2	1	0
	REID[7:6]		SEID[5:4]		PROC[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.

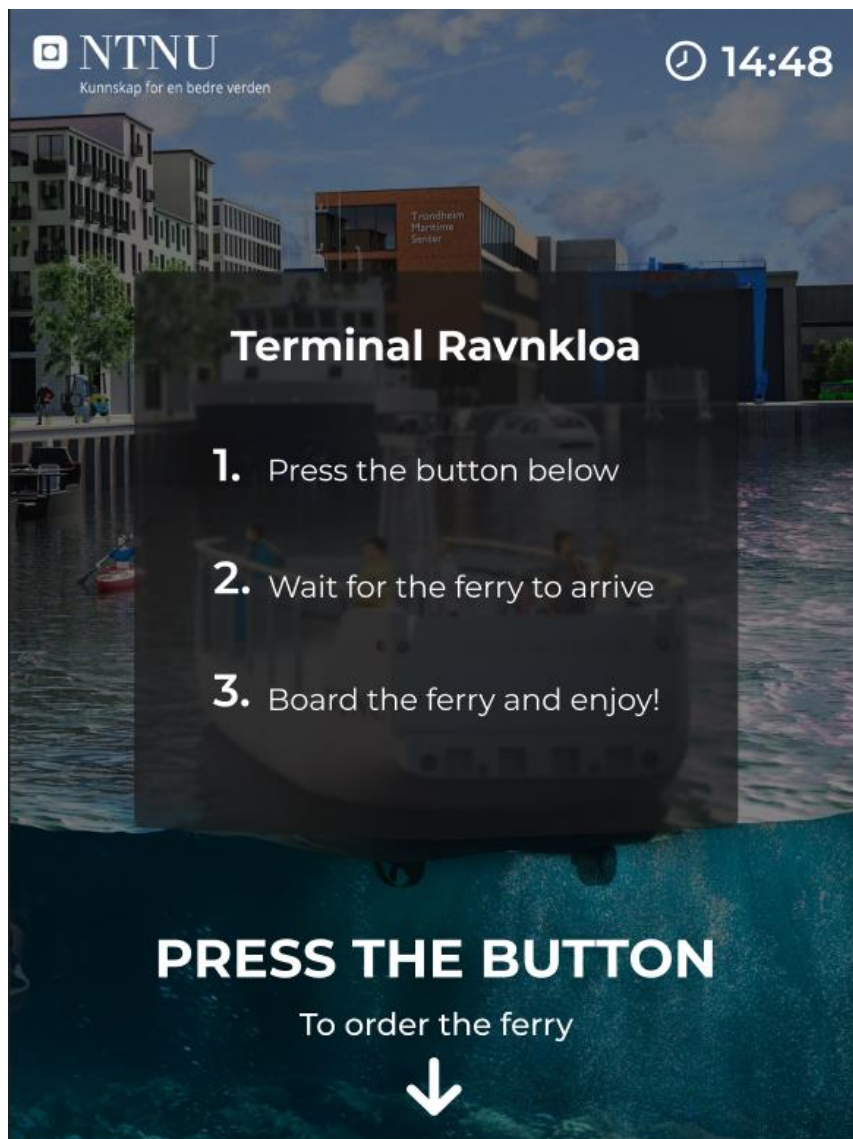
Verdi	Navn	Beskrivelse
0x0	None	None

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



Guide for oppsett av Raspberry Pi 4 med grafisk-brukergrensesnitt

E2318 – TRÅDLØS TERMINAL FOR “ON-DEMAND” BESTILLING AV AUTONOM FERGE

MILLIAMPERE2

1. Aktivere maskinvare UART:

- a. Start Raspberry Pi konfigurerings program:

```
Pi4@Terminal:~ $ sudo raspi-config
```

- b. Velg alternativ 3:

```
Raspberry Pi Software Configuration Tool (raspi-config)

1 System Options          Configure system settings
2 Display Options        Configure display settings
3 Interface Options       Configure connections to peripherals
4 Performance Options    Configure performance settings
5 Localisation Options   Configure language and regional settings
6 Advanced Options       Configure advanced settings
8 Update                 Update this tool to the latest version
9 About raspi-config     Information about this configuration tool

<Select>                                <Finish>
```

- c. Velg alternativ I6:

```
Raspberry Pi Software Configuration Tool (raspi-config)

I1 Legacy Camera Enable/disable legacy camera support
I2 SSH           Enable/disable remote command line access using SSH
I3 VNC           Enable/disable graphical remote access using RealVNC
I4 SPI           Enable/disable automatic loading of SPI kernel module
I5 I2C           Enable/disable automatic loading of I2C kernel module
I6 Serial Port  Enable/disable shell messages on the serial connection
I7 1-Wire       Enable/disable one-wire interface
I8 Remote GPIO  Enable/disable remote access to GPIO pins

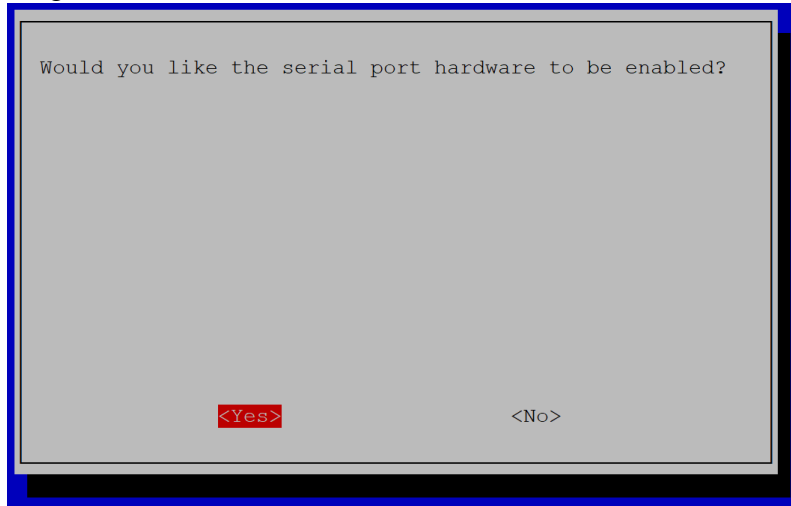
<Select>                                <Back>
```

- d. Velg <No>

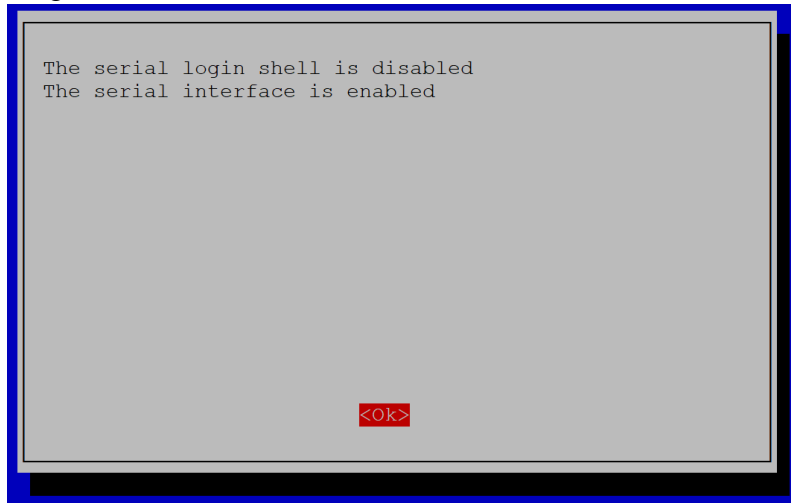
```
Would you like a login shell to be accessible over
serial?

<Yes>                                <No>
```

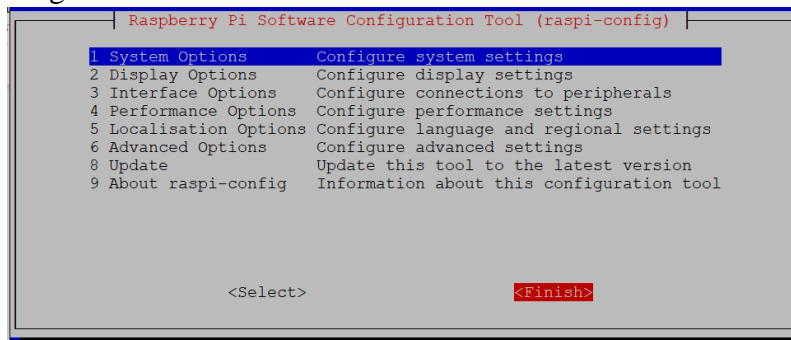
e. Velg <Yes>:



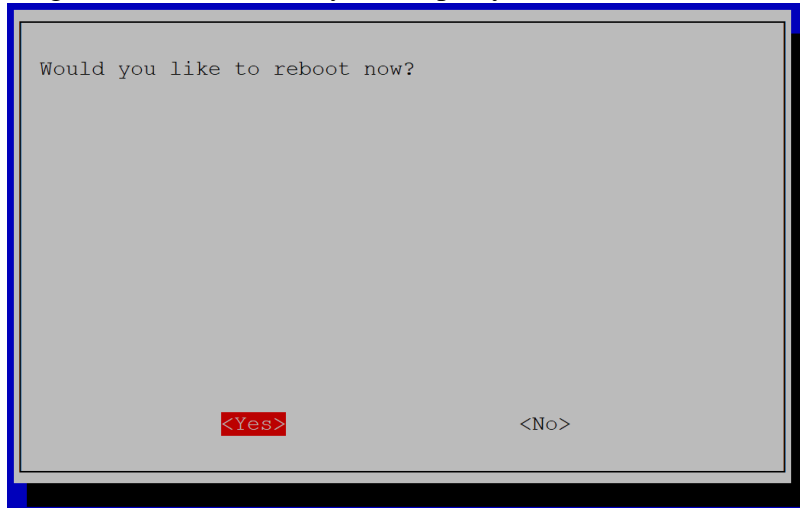
f. Velg <Ok>:



g. Velg <Finish>:



- h. Velg <Yes>, for å starte systemet på nytt:



2. Skru av bluetooth-modul for å bruke AMA0 (GPIO 14 og GPIO 15):

- a. Åpne /boot/config.txt filen:

```
Pi4@Terminal:~ $ sudo nano /boot/config.txt
```

- b. Bla nederst og skriv inn:

```
[all]
enable_uart=1
dtoverlay=disable-bt
```

- c. Ctrl+X, Y, Enter, for å lukke og lagre filen.

3. Deaktiver service filer som forstyrrer UART:

```
Pi4@Terminal:~ $ sudo systemctl disable hciuart.service
Pi4@Terminal:~ $ sudo systemctl stop serial-getty@ttyAMA0.service
Pi4@Terminal:~ $ sudo systemctl disable serial-getty@ttyAMA0.service
```

4. Start systemet på nytt for å la endringene for UART ta effekt:

```
Pi4@Terminal:~ $ sudo reboot
```

5. Test UART for å sjekke om det er rett satt opp:

- a. Koble opp UART på GPIO 14(Tx) og 15(Rx)

- b. Installer minicom eller lignende:

```
Terminal@Ravnkloa:~ $ sudo apt install minicom
```

- c. Kjør minicom for å se om UART er rett satt opp:

```
Pi4@Terminal:~ $ minicom -b 115200 -o -D /dev/ttyAMA0
```


- a. Opprett en .service fil for start av server på oppstart i /etc/systemd/system/:

```
Pi4@Terminal:~ $ sudo nano /etc/systemd/system/E2318-Terminal_server.service
```

- b. Skriv inn følgende i filen*:

```
GNU nano 5.4 /etc/systemd/system/E2318-Terminal_server.service
[Unit]
Description = Service for starting react app server

[Service]
WorkingDirectory=/home/Pi4/Desktop/E2318-Terminal/server
ExecStart=/usr/bin/npm start
Restart=Always
User=Pi4

[Install]
WantedBy=multi-user.target
```

*Merk at WorkingDirectory må være satt til hvor server-mappen i React-appen er på Raspberry-en.

- c. Opprett en .service fil for start av server på oppstart i /etc/systemd/system/:

```
Pi4@Terminal:~ $ sudo nano /etc/systemd/system/E2318-Terminal_client.service
```

- d. Skriv inn følgende i filen*:

```
GNU nano 5.4 /etc/systemd/system/E2318-Terminal_client.service
[Unit]
Description = Service for starting react app client

[Service]
WorkingDirectory=/home/Pi4/Desktop/E2318-Terminal/client
ExecStart=/usr/bin/serve -s build
Restart=Always
User=Pi4

[Install]
WantedBy=multi-user.target
```

*Merk at WorkingDirectory må være satt til hvor client-mappen i React-appen er på Raspberry-en.

- e. «enable» begge .service filene, for å la de starte på oppstart av systemet:

```
Pi4@Terminal:~ $ sudo systemctl enable E2318-Terminal_server.service
```

```
Pi4@Terminal:~ $ sudo systemctl enable E2318-Terminal_client.service
```

9. Oppsett av skjerm (Portrett modus(venstre)):

- a. Opprett en .desktop fil i /etc/xdg/autostart/:

```
Pi4@Terminal:~ $ sudo nano /etc/xdg/autostart/ownDispSetup.desktop
```

- b. Skriv inn følgende i filen:

```
GNU nano 5.4 /etc/xdg/autostart/ownDispSetup.desktop
[Desktop Entry]
name=Raspberry Pi Screen Setup
Exec=/bin/bash -c "sleep 2 && xrandr --output HDMI-1 --rotate left"
Type=Application
```

- c. Ctrl+X, Y, Enter, for å lukke og lagre filen.

10. Sett opp Chrome til å starte på oppstart i kiosk-modus:

a. Åpne filen autostart i /etc/xdg/lxsession/LXDE-pi/autostart:

```
Pi4@Terminal:~ $ sudo nano /etc/xdg/lxsession/LXDE-pi/autostart
```

b. Skriv inn følgende nederst i filen:

```
@xset s off
@xset -dpms
@xset s noblank
@chromium-browser --kiosk http://localhost:3000
unclutter -idle 0
█
```

c. Ctrl+X, Y, Enter, for å lukke og lagre filen.

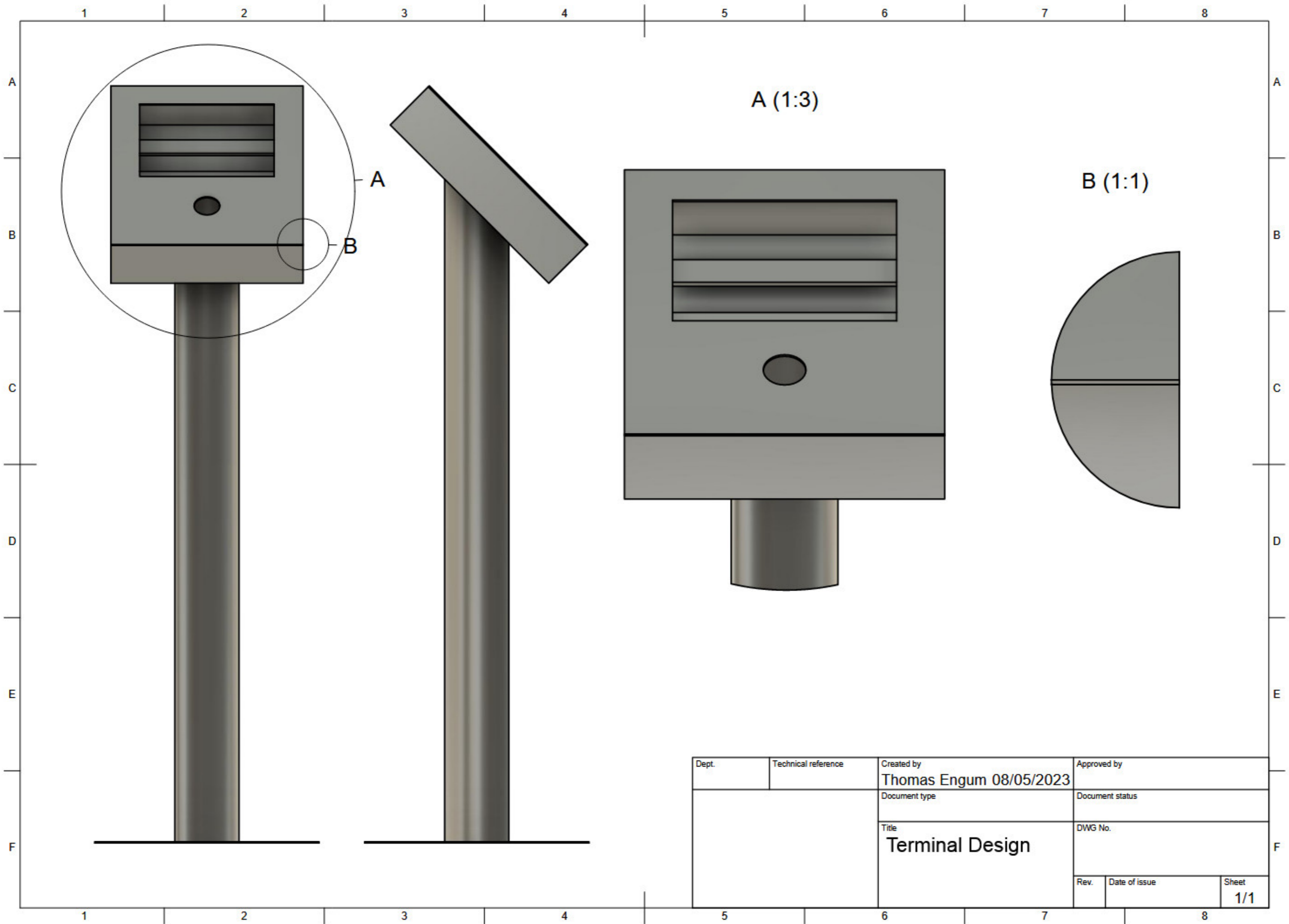
d. Installer unclutter for å fjerne musepeker:

```
Pi4@Terminal:~ $ sudo apt install unclutter
```

e. Start systemet på nytt for å la endringene ta effekt:

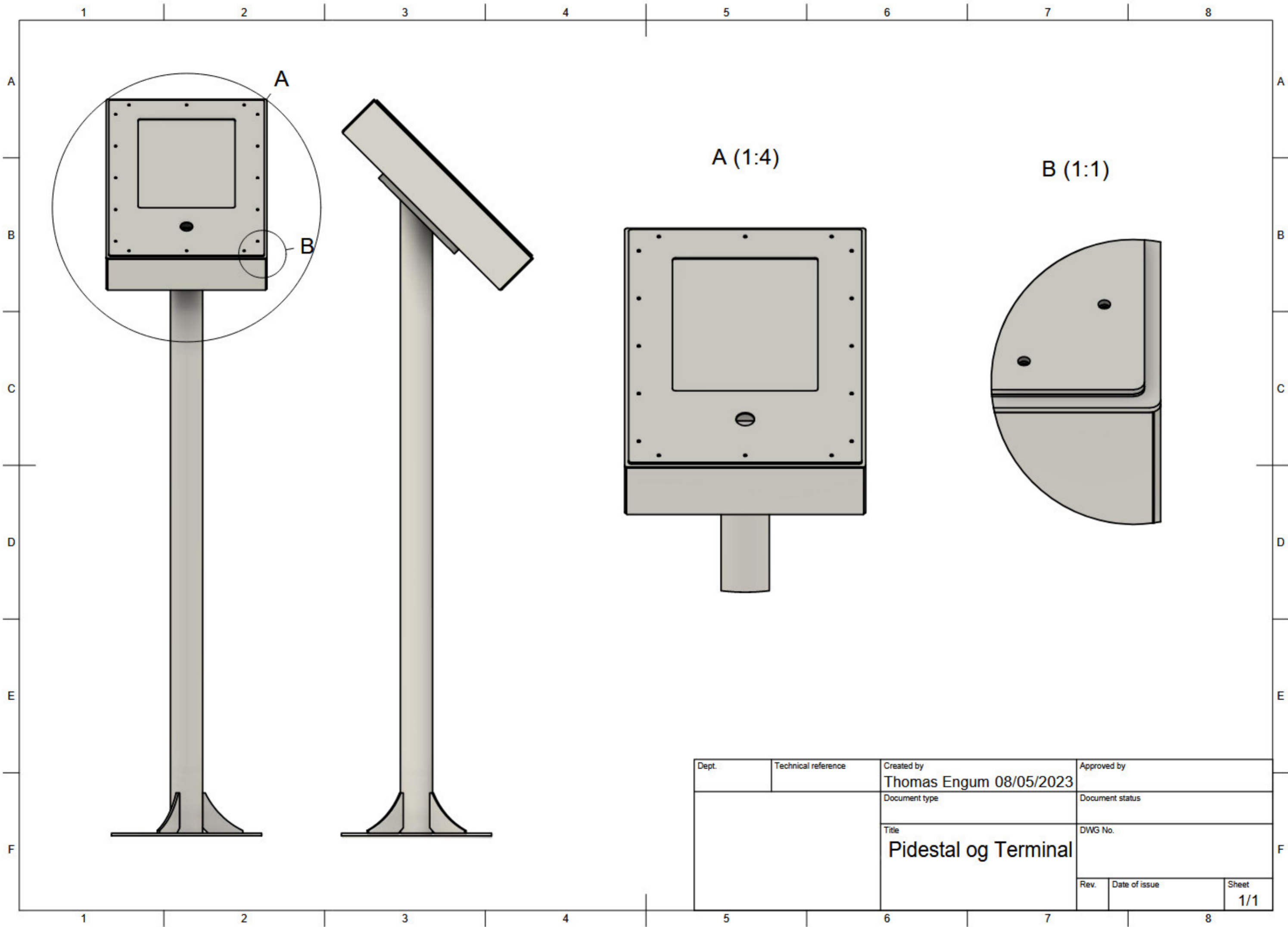
```
Pi4@Terminal:~ $ sudo reboot
```


C Design av terminal



Dept.	Technical reference	Created by Thomas Engum 08/05/2023	Approved by
		Document type	Document status
		Title Terminal Design	DWG No.
		Rev.	Date of issue
			Sheet 1/1

D Pidestill og terminal



Dept.	Technical reference	Created by Thomas Engum 08/05/2023	Approved by
		Document type	Document status
		Title Pideстал og Terminal	DWG No.
		Rev.	Date of issue
			Sheet 1/1

E Risikoanalyse

RISIKOANALYSE

Enhet/institutt	NTNU - Institutt for elektroniske systemer	Dato opprettet:	26.01.2023
Ansvarlig (navn):	Egil Eide	Sist revidert:	09.02.2023
Ansvarlig for aktiviteten som risikovurderes (navn):	Idar Evenstad		
Deltakere (navn):	Idar Evenstad, Petter Wandsvik, Magnus Dag Oseid og Thomas Engum		

Beskrivelse av den aktuelle aktiviteten:
Gruppen skal konstruere, designe og teste en bestillingsterminal for autonomfergen "Milliampere2". Terminalen skal være vandalsikker og sertifisert for IP67 el. tilsvarende.

Aktivitet/arbeidsoppgave:	Mulig uønsket hendelser	Eksisterende risikoreducerende tiltak	Vurdering av sannsynlighet(S) (1-5)	Vurdering av konsekvens (K)				Risikoverdi (S x K)	Forslag til forebyggende og/eller korrigerende tiltak:	Restrisiko (S x K)
				Menneske (1-5)	Øk/materiell (1-5)	Omdømme (1-5)	Ytre miljø (1-5)			
Integrering av system med 230v	Strømgjennomgang	Forsiktighet når man er i nærheten av steder med høyere spenniger. Lese brukermanualer.	3	5	3			15	Skru av spenning/slå ut sikringen. Gode rutiner rundt bruk av høyere spenninger	8 (2 x 4)
Test av system på vann	Fare for å falle på vannet	Redningsvest	3	3				9	Være 2 eller flere personer på båten/bryggen når systemet tester. Livredningsbøye tilgjengelig.	4 (2 x 2)
Kobling av maskinvare på kretskort.	Elektrostatisk utlanding	Antistatisk pose rundt kretskortene	2		4			8	Bruk av jordingsarmbånd og antistatisk matte når man skal holde på med selve PCB-en.	3 (1 x 3)
Simulering av brukeropplevelsen (VR-lab)	Kan gå på ting som man ikke ser på grunn av virtuell simulering	Ha det ryddig rundt modellen av ferga.	4	2				8	Være to personer til enhver tid, slik at en kan passe på hvor den som bruker VR-brillene er i den ekte verden	2 (1 x 2)
Programmering av programvare og bruker grensesnitt	Programmering av LoRa er nytt og blir mer komplisert enn først antatt.	Generelle kunnskaper om programmering i C og noen forskjellige mikrokontrollere	3		3			9	Starte tidlig med å lese på manual til mikrokontroller og perefierenheter.	4 (2 x 2)
Research av teori og fagstoff	Vondt i rygg og nakke	Pauser, strekke ut med jevne mellomrom	4	1				4	Innføring av gode kontorstoler der man jobber og bytte sittestilling ofte.	3 (3 x 1)
Generelt arbeid	Sykdom/skade	Godt smittevern	3	2				6	Godt smittevern og gjennomføring av tiltak etter risikoplan, for å begrense sannsynlighet for menneskelige skader	4 (2 x 2)
Lodding av komponenter	Brannskade, innånding av giftige gasser	Loddekurs	4	3				12	Gode rutiner rundt lodding. Bruk av avtrekksvifte med kullfilter. Drikk og spis et annet sted en hvor loddingen finner sted. Vaske hender etter lodding og før mat og drikke inntak	6 (3 x 2)
Bestilling av deler	Delene blir forsinket	Bestille i god tid	3		4			12	Sjekk lagerstatus fra leverandør. Bestille mest mulig fra Norge/EU, hvor frakttiden kan være kortere.	6 (2 x 3)
	Underdimensjonering av deler og innkapsling	Måle og kalkulering av nødvendige parametre og mål	3		2			6	Bruk av utstyr på NTNU for å måle de ulike parametrene. Måle på ferdig krets før man bestiller innkapsling for å være sikker på at det passer.	4 (3 x 2)
Testing av vanntetthet	Ikke vanntett, lekker vann inn til kretskortet	Research på IP67	4		2			8	Gjennomfør ulike tester og les på teroi rundt å få innkapslinger vanntett	6 (3 x 2)
Stresstest av komplett system	Det ferdige produktet fungerer ikke som det skal til enhver tid og er ikke pålitelig	Kjøre ulike tester underveis og gjøre research på de ulike modulene	3		2	4		12	Test for ulike scenarier og sørg for at det fungerer alltid uansett scenario	6 (2 x 3)

Risikoverdi = Sannsynlighet x Konsekvens. Risikoverdi er beregnet først og fremst for menneske. Dernest den verdien som er mest signifikant.

ref: <https://www.ntnu.no/hms/retningslinjer/HMSRV2603B.pdf>

Sannsynlighet vurderes etter følgende kriterier:

Svært liten 1	Liten 2	Middels 3	Stor 4	Svært stor 5
1 gang pr 50 år eller sjeldnere	1 gang pr 10 år eller sjeldnere	1 gang pr år eller sjeldnere	1 gang pr måned eller sjeldnere	Skjer ukentlig

Konsekvens vurderes etter følgende kriterier:

Gradering	Menneske	Ytre miljø	Øk/materiell	Omdømme
5 Svært Alvorlig	Død	Svært langvarig og ikke reversibel skade	Drifts- eller aktivitetsstans >1 år	Troverdighet og respekt betydelig og varig svekket
4 Alvorlig	Alvorlig personskade. Mulig uførhet.	Langvarig skade. Lang restitusjonstid	Driftsstans > ½ år Aktivitetsstans i opp til 1 år	Troverdighet og respekt betydelig svekket
3 Moderat	Alvorlig personskade.	Mindre skade og lang restitusjonstid	Drifts- eller aktivitetsstans < 1 mnd	Troverdighet og respekt svekket
2 Liten	Skade som krever medisinsk behandling	Mindre skade og kort restitusjonstid	Drifts- eller aktivitetsstans < 1uke	Negativ påvirkning på troverdighet og respekt
1 Svært Liten	Skade som krever førstehjelp	Ubetydelig skade og kort restitusjonstid	Drifts- eller aktivitetsstans < 1dag	Liten påvirkning på troverdighet og respekt

Fargene angir grad av risiko:

KONSEKVENNS	5 - Svært alvorlig	5	10	15	20	25
	4 - Alvorlig	4	8	12	16	20
	3 - Moderat	3	6	9	12	15
	2 - Liten	2	4	6	8	10
	1- Svært liten	1	2	3	4	5
		1 - Svært liten	2 - Liten	3 - Moderat	4 - Stor	5 - Svært Stor
SANNSYNLIGHET						

F Heksadesimalt til dBm



Magnus Oseid <magnusoseid@gmail.com>

Case# 01260182 - New Comment - Convert RSSI Hex Value - [ref: _00Do0KAKK._500311Y61ew:ref]

1 message

Microchip Support <replytotechsupport@microchip.com>
To: "magnusoseid@gmail.com" <magnusoseid@gmail.com>

Tue, May 9, 2023 at 1:49 PM

--- DO NOT EDIT ANY OF THE TEXT THAT FOLLOWS ---

Hello Magnus Dag,

Microchip Technical Support has added a comment on Case# **01260182**.

Case Link: <https://microchipsupport.force.com/s/case/5003100001Y61ew>

Latest Comment:

Hi Magnus Dag,

Please try the method below to convert dBm value to hexadecimal or vice versa.

Assume that you measured the RSSI level at -59 dBm and you need to convert that to the proper signed value for the tx power:

1. Write the positive value of the RSSI in binary format: (59) decimal= (0011 1011) binary
2. Reverse all bits: (0011 1011) binary => (1100 0100) binaryreversed
3. Take 1's complement: (1100 0100) binary reversed => (11000101)1's complement
4. Convert to hexadecimal: (1100 0101)1's complement = (c5) hex

For converting the hex value to dBm,

1. Convert Hexadecimal value to binary number.
2. Calculate 2s compliment for this:
 - i) Reverse all bits in binary.
 - ii) Add 1 to the binary number.
 - iii) Convert this to decimal with – as prefix.

Hope this answers your query.

G Brukeropplevelse undersøkelse

Brukeropplevelse ved bestilling av milliAmpere2

Etter gjennomført simulering av bestilling ber vi om å fylle ut skjemaet for å kartlegge hvordan opplevelsen er ved bestilling av milliAmpere2

* Obligatorisk

1. Hvor fornøyd er du med brukeropplevelsen? (1 = veldig lite fornøyd, 6 = veldig fornøyd) *

1	2	3	4	5	6
---	---	---	---	---	---

2. Hvor fornøyd er du med tilbakemeldingene du fikk på terminalene ved bestilling? (1 = veldig lite fornøyd, 6 = veldig fornøyd) *

1	2	3	4	5	6
---	---	---	---	---	---

3. Forklar hvordan du opplevde bestillingen av milliAmpere2 *

H Mail ang. datablad for skjerm

Thomas Jøntvedt Engum

From: Felix-Marcel Petermann
Sent: 27 February 2023 15:44
To: Idar Evenstad
Cc: Thomas Jøntvedt Engum; Magnus Dag Oseid; Petter Wandsvik
Subject: Re: Veien fra skisse til vandalsikker, IP-rangert bestillingsterminal

Hei,

So we have two waterproof marine touchscreen displays, 1000 nits lamination, in the size around 12,1". 4:3 format. It is 210mm high and 160.5mm wide, it has a margin of 15 mm on all sides, installation depth is 55 mm, on the sides you probably need more space because the connectors are water pressure-tight and quite bulky for that reason.

Compare with following, that's not the exact one but close: https://www.alibaba.com/product-detail/Aluminum-alloy-marine-displays-1000-nits_1600101712893.html?spm=a2700.details.0.0.5f5c7ca2jHfi3h

This you could borrow for your project.

Alternatively I can offer a 10.1 " LCD screen 1024x600, that one is not water- and vandal proof.
https://www.amazon.de/gp/product/B01CQUCJDM/ref=ppx_yo_dt_b_asin_title_o00_s02?ie=UTF8&psc=1

I guess its best if you take a look if they fit your use case.

Kind Regards / Med vennlig hilsen

Felix-Marcel Petermann
PhD Student

Department of Design
Faculty of Architecture and Design
Norwegian University of Science and Technology

From: Idar Evenstad <idar.evenstad@ntnu.no>
Date: Monday, 27 February 2023 at 15:19
To: Felix-Marcel Petermann <felix.m.petermann@ntnu.no>
Cc: Thomas Jøntvedt Engum <thomas.j.engum@ntnu.no>, Magnus Dag Oseid <magnus.d.oseid@ntnu.no>, Petter Wandsvik <petter.wandsvik@ntnu.no>
Subject: RE: Veien fra skisse til vandalsikker, IP-rangert bestillingsterminal

Hi Felix,
Could you give any update on this mail?

Kindly regards
Idar Evenstad

From: Ole Andreas Alsos <oleanda@ntnu.no>
Sent: onsdag 22. februar 2023 14:24
To: Idar Evenstad <idar.evenstad@ntnu.no>; Petter Wandsvik <petter.wandsvik@ntnu.no>; Leander Spyridon Pantelatos <leander.s.pantelatos@ntnu.no>; Felix-Marcel Petermann <felix.m.petermann@ntnu.no>
Cc: Thomas Jøntvedt Engum <thomas.j.engum@ntnu.no>; Magnus Dag Oseid <magnus.d.oseid@ntnu.no>
Subject: Re: Veien fra skisse til vandalsikker, IP-rangert bestillingsterminal

I Fullstendig deleliste

Utstyr	Antall
2-veis forgreiner m/kabel (x meter)	x2
Angled Female Pin Header (2x2)	x3
Angled Female Pin Header (2x10)	x3
Antenne-puck	x2
Antenne-pisk	x1
Female pin header (6x1)	x2
Female pin header (4x1)	x2
Gummigjennomføringer (1/4" x1/2")	x6
HDMI til DIN(19-Pin)	x2
Innkapsling - Ferge	x1
Innkapsling - Terminal (frontpanel)	x2
Innkapsling - Terminal (kabinett)	x2
Innkapsling - Terminal (pidestal)	x2
Innkapsling - SAM R34	x2
Innkapsling - Raspberry Pi	x2
JST male (1x3)	x2
JST male (1x4)	x5
JST male (1x6)	x3
JST female (1x3)	x2
JST female (1x4)	x5
JST female (1x6)	x3
Kabelgjennomføring	x4
Koaksialkabel 5 meter	x1
Maskinskrue M3x8mm unbrako	x16
Mikro-HDMI til HDMI	x2
Mikro-USB strømledning	x4
O-ring 22.5mmx2.5mm	x2
RS485-modul	x3
SAM R34 Xplained pro	x3
Spacer m/skruer - PCB (plastikk)	x8
Signalkabel (5.5 meter)	x1
Skjerm	x2
Skjermdriver SBC	x2
Skruterminal (1x2)	x2
Strømforsyning 5v RPi 4B	x2
Strømforsyning 12v Skjerm	x2
Sukkerbit	x6
Tetningslist vinyl gr.3mm	x2
Trykknapp m/lysring	x2
TNC - adapter	x1
UTP - Ethernet CAT6 (x meter)	x2
USB-seriell omformer	x3
Utvidelseskort - Ferge	x1
Utvidelseskort - Terminal	x2
Vifte - RPi 4B	x2
Wago 221 (1x2)	x6

J Budsjett for utstyrsinnkjøp

Utstyr	Delnummer	Produsent	Antall enheter	Sats (kr)	Kostnad (kr)
Antenne Pisk	DELTA5A/X/TNCM/S/S/17	SIRETTA	1	101.83	101.83
Antenna Puck	ANT-8/9-SPS1-1	Linx	4	317.20	1268.8
Coaxial Adapter	122346	AMPHENOL RF	1	105.71	105.71
DC-DC converter	PQDE6W-Q24-S5-T	CUI Inc.	1	210.00	210.00
Innkapsling(Aluminium)	1455P1601BK	Hammond Manufacturing	1	354.76	354.76
Innkapsling - Terminal (alle deler)	Ingen	Verksted ITK/Smith stål	2	4,000.00*	4,000.00*
Knapp med lysring	82-6153.1113	EAO	2	408.00	816.00
Litiumbatteri CR1220	32230	Kjell & Company	3	29.90	89.70
Mikro-HDMI til HDMI adapter	98790	Kjell & Company	2	129.90	259.8
Raspberry Pi 4B - kit**	5011179832	One Nine Design	2	1,995.00	3,990.00
RS485-modul	410-310	Digilent	3	406.76	1,220.28
SAM R34 Xplained Pro	DM320111	Microchip Technologies	2	1,125.00	3,375.00
Skjerm	Ukjent	Ukjent	2	3,431.00	6,862.00
Sukkerbit	Ukjent	Ukjent	6	2.00	12.00
Tetningslist vinyl gr.3mm	NA16000-4-3	PSP marine tapes	2	142.00	284.00
USB-seriell omformer	EX-1309-T	Exsys	3	654.00	1,962.00
Total:					24,911.88

* 4,000.00 kr var totale prisen for materialet til to terminaler

** Kit inneholder: Vifte, innkapsling og strømforsyning 5v

K Populærvitenskapelig artikkel

Bestillingssystem for milliAmpere2 for å redusere byens omveier

I dag må man gå omveier til nærmeste bro for å krysse en kanal eller elv. milliAmpere2 er en autonom ferge som kan benyttes over kanalstrekninger og dermed senker avstanden til nærmeste vannkryssing.

Fergen skal settes inn i trafikk over kanalen i Trondheim, mellom Ravnkloa og Vestre Kanalkai. Det skal ikke være skytteltrafikk og bruker må bestille fergen fra siden man ønsker å reise fra. Løsningen innebærer to produserte landterminaler på hver sin side som kommuniserer trådløst med fergen og gir beskjed når personer ønsker å benytte den. milliAmpere2 registrerer bestillingen og tar avgjørelse basert på foreløpig bestillings-kø og hvor den befinner seg i kanalen.

På landsidene av kanalen vil det foreligge porter som et sikkerhetsaspekt for menneskelig bruk. Portene åpnes når det er klart for mennesker å gå av eller ombord i fergen. Terminalene fungerer som et mellomledd som videregir beskjeder fra fergen til en datamaskin som styrer portene.

Ideen om en autonom ferge vil i Trondheim ha større potensial enn kun over kanalen og enda flere omveier kan kuttes. De utviklede landterminalene er tilrettelagt for operasjon i større skala, da den trådløse kommunikasjonen er kapabel til å dekke et omfattende område.

Kampen mellom menneske og teknologi

Teknologi og menneske kjemper en evig kamp mot hverandre. Teknologien er ofte komplisert og man må ha gode forkunnskaper for å forstå hvordan den faktisk fungerer. Teknologene må av den grunn ikke glemme hvem som er bruker av teknologien som utvikles, og må gjøre den så forståelig som overhode mulig.

milliAmpere2 medfører store spørsmål og problemstillinger tilknyttet interaksjon mellom menneske og teknologi. Teknologien skal ta menneske med på en seilas, uten at det oppstår fare for sikkerheten til passasjerene. I tillegg skal systemet være intuitivt for menneske å benytte. På terminalene følger med en skjerm, hvor funksjonen er å gi brukeren lett og oversiktlig informasjon om hvordan man benytter seg av milliAmpere2.

Teknologien må på den annen side beskyttes mot menneske. Oppførselen til menneske er vanskelig å forutse, og mennesker kan handle på en slik måte som skader teknologien. Det vil ikke være mulig å bestille fergen flere ganger etter hverandre, hvor en foreliggende bestilling vil hindre et nytt knappetrykk i å foreta handlinger.



Det er aktivt jobbet gjennom prosjektet for å beskytte landterminalene mot uønsket menneskelig oppførsel, samtidig som de tilbyr et brukervennlig bestillingssystem. Det er derimot vanskelig å forutse alt av situasjoner som kan oppstå, og implementasjon i praksis vil kartlegge flere scenarier hvor menneske og teknologi må tilpasses hverandre.

Trondheim er et uoversiktlig værkaos

Elektrisitet og vann er ikke alltid de to beste vennene. Terminaler og teknologi som skal operere i et utendørs klima må tåle været Trondheim utsetter sine innbyggere for. Det å sikre terminalene og de elektriske enhetene mot vann og andre elementer som kan komme til skade, har vært nødvendig for at bestillingssystemet til milliAmpere2 fungerer uansett værforhold.

Terminalene er utviklet og produsert med særlig tanke på at det ikke skal være mulig for vann å trekke inn og skade komponentene. Bestillingsknappen er vannsikker, og det vil være en pakningslist mellom topplaten og boksen for å hindre at vann trekker inn gjennom sprekkene.



I tillegg vil værforholdene ha innvirkning på den trådløse kommunikasjonen, hvor regn fører til svekkelse i radiosignalet. Det er undersøkt signalstyrker på tvers av kanalen for kartlegge hvilket signalnivå som kreves for å opprettholde en pålitelig kommunikasjon, slik at det til enhver tid vil være mulig å benytte seg av milliAmpere2.

Ikke bare et fremkomstmiddel, men en opplevelse

milliAmpere2 er et godt tilbud for å korte ned gåavstanden mellom Ravnkloa og Vestre Kanalkai, og med pålitelig bestilling og passasjerhåndtering vil det være en flott brukeropplevelse. Teknologi og menneske er tilpasset hverandre gjennom hele seilasen, fra bestilling til man stiger av på andre siden. Menneske skal fremdeles ikke undervurderes, og det kan lett oppstå situasjoner det er vanskelig å planlegge for.

Fergen gir mennesket og brukeren en god innsikt i et potensielt fremtidig samfunn bestående av autonome fremkomstmidler, samt innblikk i den teknologiske utviklingen. Det vil være en opplevelse å benytte seg av en ferge som styrer seg selv, og man blir ofte forbløffet over teknologiske fremskritt og muligheter.

Landterminalene og bestillingen av milliAmpere2 danner et grunnlag for videre forskning på løsninger som inkluderer teknologiske og menneskelige aspekter. Begge må tilpasses hverandre, i tillegg må det tas forbehold om at vær og vind vil påvirke vurderinger og løsninger.

Prosjektet tilbyr mennesket en ny mulighet for å krysse kanalen, som vil være mer innholdsrik og spennende enn å gå over en bro.

