

NORGES TEKNISK-NATURVITENSKAPELIGE UNIVERSITET

ELEKTRONISKE SYSTEM: AVANSERT DESIGNPROSJEKT

TFE4205

Trebeint robot

Omega Verkstedes arbeidsgruppe:

Per Anton Øverseth Olsen

Petter Diderik Breedveld

Einar Henriksen

Johannes Wågen

Mai 2018



Veileder: Geir Mathisen

Forord

Denne rapporten spesifiserer de tekniske avgjørelsene og motivasjonen for design, utvikling og testing av en trebeint plattformrobot bygget i.l.a våren 2018. Prosjektet ble gjennomført i forbindelse med faget TFE4205 etter oppdrag fra Omega Verksted. Grappa som gjennomførte prosjektet besto av fire aktive styremedlemmer i Omega Verksted. Tre av disse studerer teknisk kybernetikk/kybernetikk og robotikk, fjerde og femte året, og en studerer elektronisk systemdesign og innovasjon, fjerde året. Motivasjonen for å lage en trebeint robot som skal kunne gå, er å øke den generelle interessen og kompetansen innenfor robotikk på Omega Verksted, samt utvikle moduler som kan være nyttige å bruke i andre Omega Verksted-prosjekter. Veileder og fagkoordinator godkjente oppgaven, og Omega Verksteds styre satte interne krav til gjennomføring og resultat. Denne rapporten er skrevet med hensyn til lesere med noe teknisk kompetanse. Design og bygging forekom på Omega Verksteds lokaler.

En oversikt over akronymer som brukes i rapporten finnes i vedlegg A.

Sammendrag

Studien av vandrende roboter, eller roboter som benytter bein til forflytning har lenge vært et aktivt forskningsfelt. De siste årene har det vært stor utvikling innen fagfeltet. Arbeidsgruppa ville vise at det er mulig å produsere en mindre vanlig type robot med bein, en *tripod*, uten at kostnadene skulle bli uoverkommelige. Roboten er en platform med tre bein, bestående av tre ledd, plassert i hvert sitt hjørne av en likesidet trekant. Oppgaven i samarbeid med Omega Verksted var todelt: En robot skulle bygges, og nyttige moduler til bruk i forskjellige prosjekter skulle utvikles. En essensiell del av oppgaven var å dokumentere prosjektet etter Omega Verksteds standarder, eventuelt utarbeide gode standarder der det ikke fantes. Arbeidsgruppen fant etter et innledende litteraturstudie ut at komponenter benyttet i andre lignende prosjekter er veldig varierende i pris, med stor overvekt av deler med en prislapp som ikke er rimelig for dette prosjektets økonomiske rammer. Arbeidsgruppen sto fritt til å velge en robotarkitektur, og dette førte til at den resulterende arkitekturen var kompatibel med de komponentene som var i vår prisklasse. Elektronikken ble designet med hensyn til at forskjellige kommunikasjonsprotokoller skulle kunne anvendes, ettersom de forskjellige modulene skulle være gjenbrukbare i forskjellige prosjekter. Altium Designer ble benyttet for design av elektronikk og kretskort ble bestilt fra (hvor) og produsert på Omega Verksted. Arbeidsgruppen fant ut at *Beaglebone Black* var en fornuftig sentralenhet til styring av det overordnede systemet. Mekaniske deler ble designet i SolidWorks og produsert ved hjelp av 3D-printing og andre metoder. Arbeidsgruppen implementerte standardfunksjoner for de ulike modulene, IMU, motormodul og vektsensor. Avanserte funksjoner som regulatordesign, støyfiltrering og sensorfusjon ble undersøkt. Mye arbeid gjenstår fortsatt pr. Juni 2018, som er beskrevet i rapporten, men basisfunksjonalitet er implementert, dokumentert og designet ferdig.

Takk til:

Forfatterne vil gjerne takke Geir Mathisen for veiledning omhandlende prosessen samt tekniske spørsmål, styret i Omega Verksted anno 2018 for tro på prosjektet, og Omega Verksted's pangsjonister for veiledning.

Innhold

Forord	i
Sammendrag	ii
Takk til	iii
Innhold	iv
1 Introduksjon	2
1.1 Motivasjon	2
1.2 Oppgavevalg og -beskrivelse	3
1.2.1 Prosjektbeskrivelse	3
1.2.2 Funksjonell beskrivelse	3
1.3 Oppdragsgiver: Omega Verksted	5
1.4 Utførelse og Prosjektmodell	8
1.4.1 Work breakdown diagram	9
1.5 Markedsanalyse for moduler	9
2 Matematisk Modell	11
2.1 Innledning	11
2.2 Initielle antagelser	12
2.3 Strategi for modellering og utarbeidelse av gangarter	15
2.4 Oversikt	17
2.5 Modell av motormodul	21
2.5.1 MRAC for motormodul	21
2.5.2 Utvikling av gangart	22
2.6 Observabilitet og Styrbarhet	24
2.7 Identifikasjon	24
3 Litteraturstudie	27

3.1	Tidligere arbeider om trebeinte roboter	27
3.2	Sentralenhet	28
3.3	Aktuatorer	28
3.4	IMU	28
3.5	Kraftsensor/fotsensor	29
3.6	Enkodere og vinkelmålere	29
3.7	Mekanisk	30
3.8	Overordnet beskrivelse av arbeidet	30
3.9	Valg av materialer	30
3.10	Toleranser	31
3.11	Plattform	32
3.12	Skulderledd	32
3.13	Kneledd	33
3.14	Festing av kraftmålere	34
3.15	Kommentar til tekniske tegninger:	39
4	Elektronikk	40
4.1	IMU	41
4.2	Lastcelle-måler	42
4.3	Motor	44
4.4	Motordriver	46
4.5	Potmeterkort	48
4.6	Sentralenhet	49
4.7	RS485 Bus	50
5	Programvare	51
5.1	IMU	51
5.2	Lastcelle	51
5.3	Motordriver	51
5.4	Kommunikasjonsprotokoll	52
5.5	RS485 drivere til Beaglebone	53
6	Videre arbeid	54
6.1	Generelt	54
6.2	Trådløs styring og batteri	54
6.3	Aktiv fot	55

6.4	Mekanisk	55
6.5	Elektronikk	56
6.6	Programmering	56
6.7	Matematisk modell	57
A	Akronymer	58
A	Vedlegg	60
A.1	Kretstegninger	60
A.2	Utlegg	66
A.3	Bill of materials	70
A.4	Tekniske tegninger	72
A.5	Annen mekanisk dokumentasjon	91
A.6	Prosjektbeskrivelse	99
	Figurer	101
	Tabeller	104

Kapittel 1

Introduksjon

Denne delen beskriver motivasjon for arbeidet, hvordan arbeidet i grove trekk ble utført og i hvilken rekkefølge dette skjedde, og oppgavebeskrivelsen.

1.1 Motivasjon

Til å begynne med må det påpekes at målet med prosjektet, i grove trekk, er todelt.

- Bygge en robot: Dette for å øke robotikk-kompetansen på OV, imponere og rekruttere nye medlemmer, og øke kompetansen i OV som omhandler prosjektstyring og gjennomføring på profesjonelt nivå der det er klare tekniske krav å forholde seg til.
- Utvikle modulene som roboten består av på en slik måte at det er lett å bygge om roboten, og at modulene kan stå for seg selv som nyttige komponenter som kan brukes av medlemmer i forskjellige prosjekter og selges i komponentavdelingen i OV.

Robot

I prinsippet er det få innlysende anvendelser for en robot på tre bein. Selv om de av og til fremstilles i forskjellige science fiction filmer er det få roboter i praksis som har blitt utviklet for å ha tre bein.

Eksempler på der roboter med tre bein har blitt brukt er som mikroroboter¹ for der masseproduksjon er målet og roboter som benyttes for å forske på hvorfor det ikke finnes dyr som ha tre bein². Det er også designet kontrollsystemer for skadede quadropeds som skal returnere til tross for ett dysfunksjonelt bein [9]. Siden prosjektet har store økonomiske begrensninger i forhold til det meste av tidligere arbeid, sammenligner man seg med prosjekter som i [2], er en trebeint robot en mer økonomisk løsning, siden kravet om høy presisjon er mindre enn ved tobeinte roboter, og aktuatorer og sensorer fra et eventuelt fjerde bein kan trekkes fra den mer vanlige quadroped-robottypen. I vårt tilfelle er roboten en plattform som beveger seg ved hjelp av tre bein, dermed er *tripod* en passende beskrivelse.

1.2 Oppgavevalg og -beskrivelse

Omega Verksted deltar årlig på arrangementer som innebærer å rigge opp diverse utstyr for å vise frem til de som måtte synes mer eller mindre profesjonelt utførte prosjekter er interessante. Et robotprosjekt er ideelt for å ta med på slike arrangementer. At prosjektgruppen bestemte seg for å bygge en robot som vist i den tidlige skissen 1.1, er basert på en generell interesse for roboter med bein for enkelte av forfatterne og en generell interesse for å bygge elektromekaniske maskiner.

1.2.1 Prosjektbeskrivelse

1.2.2 Funksjonell beskrivelse

Dette er en gjengivelse av den funksjonelle beskrivelsen som ble utviklet i tråd med Omega Verksted tidlig i prosjektet. Dette er i grove trekk mye likt motivasjonen for prosjektet.

Hovedpunkter

- Vi skal bygge en robot som skal kunne gå
- Roboten skal bestå av moduler som kan selges på OV

Mål for roboten:

¹[10]

²[11]

- Roboten er en ca 0.5 m høy trebeint symmetrisk plattform der hvert bein har tre aktuatorer og passive føtter.
- Roboten skal forsynes eksternt med strøm via kabler som monteres på en slik måte at det får liten innvirkning på dynamikken. Batteri kan vurderes på sikt.
- Beaglebone Black skal brukes for å styre roboten
- De ulike leddene som roboten består av er; Aktuatormodul med rotasjonsakse, aktuatormodul med to normale rotasjonsakser, sentralenhet med IMU og Beaglebone Black, kraftsensorer til å detektere krefter som virker på de forskjellige beina. Dette innebærer å detektere hvorvidt et bein er i bakken.
- Roboten skal kunne bevege seg raskt nok til å kunne ha en gangart som ikke inkluderer sleping av bein.
- Roboten skal kunne stå stille
- Delene på roboten skal være designet for å tåle demontering samt bruk på stands og arrangementer
- Det skal være mulighet for å styre roboten med ledninger eller trådløst.

Det er for leseren kanskje opplagt at noen avgjørelser ble gjort tidlig i arbeidet, siden arbeidsgruppen var godt i gang med litteraturstudiet ved dette tidspunktet.

Generelle mål for modulene:

- Generelle nok til å være nyttige i mange forskjellige prosjekter.
- Billige nok til å selges på OV.

Spesielt for aktuator/motor-modul

- Utføre intern regulering.
- Kontrolleres fra ekstern enhet direkte.
- Måle strømtrekket til motoren(e).

- Måle vinkelposisjonen(e) til leddet/leddene.
- Ha mulighet for kommunikasjon over CAN, RS485, UART, I2C og SPI.
- Ha GPIO som endestoppbrytere og liknende kan kobles til.
- Være reprogrammerbar.

Spesielt for sentralenheten

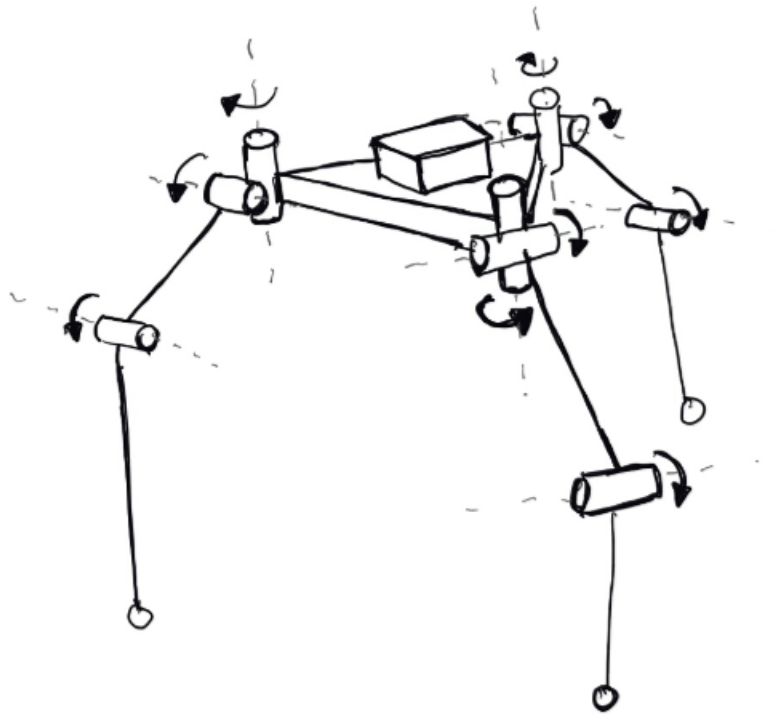
- Kjøre regulator og algoritmer som trengs raskt nok.
- Avlese IMU, annen sensordata.
- Kjøre filtrering av sensordata i sanntid.
- Ha mulighet for kommunikasjon over CAN, RS485, UART, I2C og SPI samt en form for trådløs kommunikasjon for menneske-robot interaksjon.
- Være reprogrammerbar.

Spesielt for kraftsensor-modulen

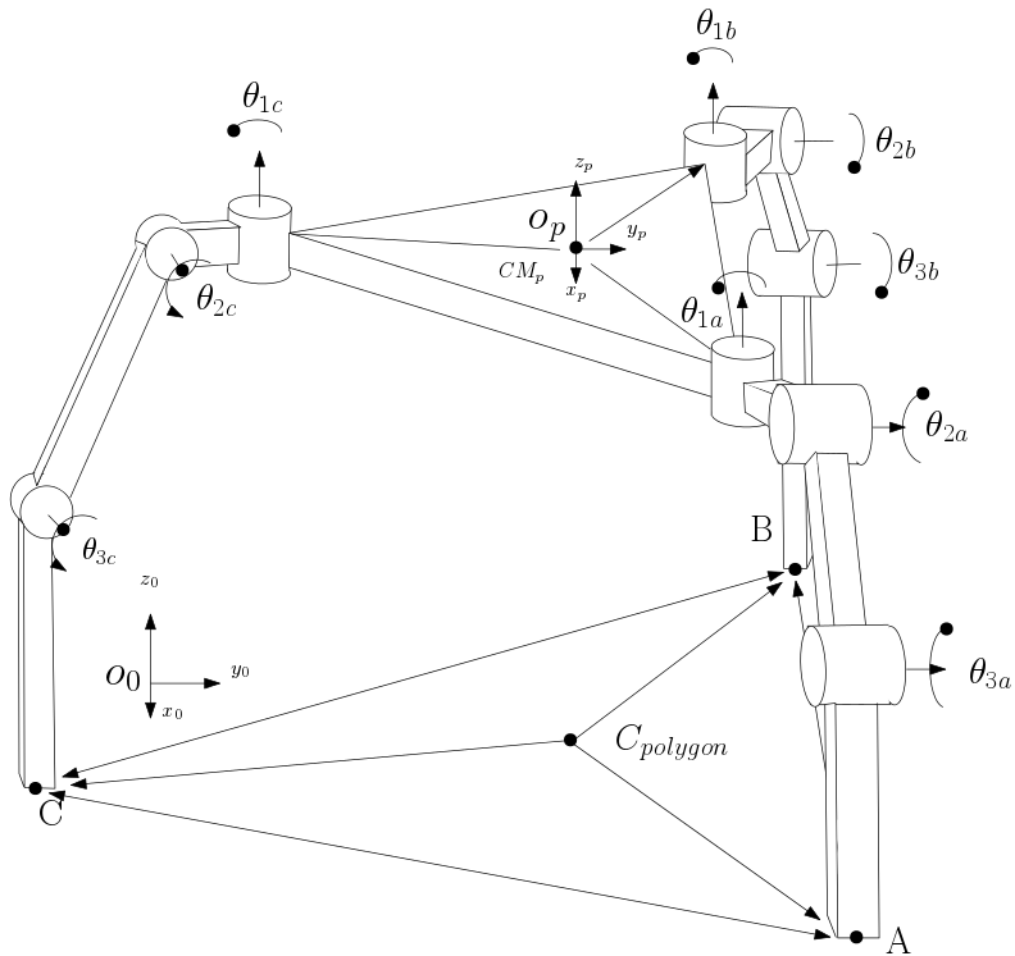
- Måle krefter i forskjellige akser.
- Kunne kjøre filtrering og estimering.
- Kjøre filtrering av sensordata i sanntid.
- Ha mulighet for kommunikasjon over CAN, RS485, UART, I2C og SPI .
- Være reprogrammerbar.

1.3 Oppdragsgiver: Omega Verksted

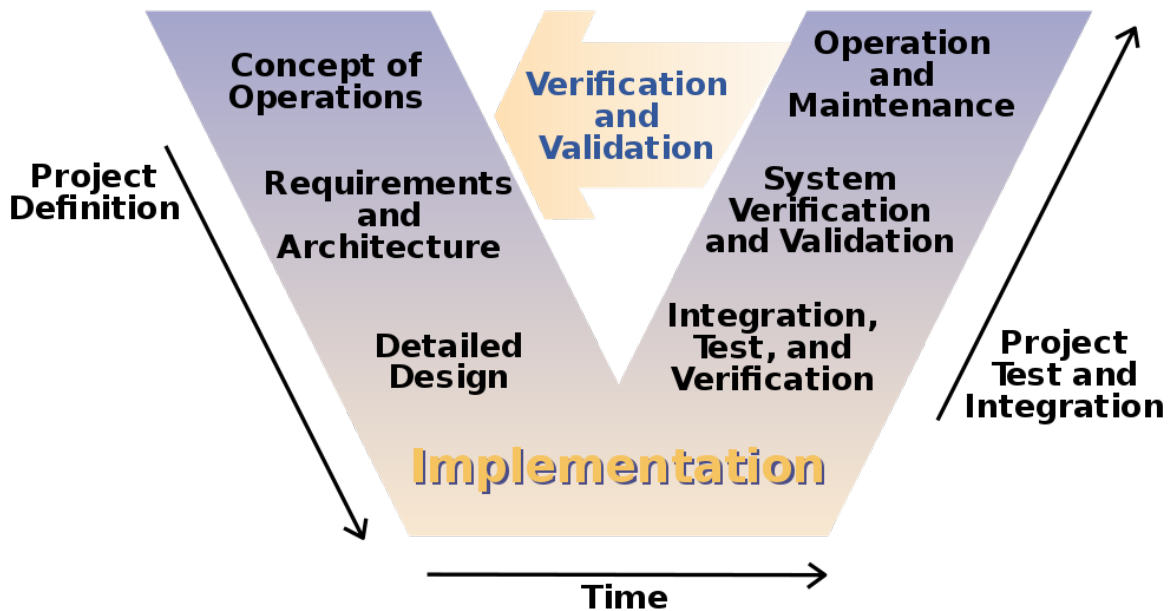
Omega Verksted er en frivillig studentorganisasjon for folk som er interesserte i praktisk elektronikk og annet. Foreningens lokaler er plassert i kjelleren på El-bygget på Gløshaugen ved OS. Bragstads plass 2. Gjennom tidene har det elektronikkprosjekter vært en sentral beskjeftigelse på OV, og prosjekter med



Figur 1.1: En tidlig konsepttegning fra en av de første møtene.



Figur 1.2: En overordnet tegning over hvordan robotens ledd er satt sammen, og hvor de generaliserte koordinatene er



Figur 1.3: V-modellen

forskjellige grader av seriøsitet har blitt gjennomført. I midlertid viser det seg at prosjektets hensikt ikke trenger å være av en seriøs standard for at virkningen og læringsutbytte skal være verdifult. For at prosjekter oftere skal være vellykket, motiverende å jobbe med og mulige å utbedre eller reparere av noen andre har det vist seg å være særdeles nyttig med flittig bruk av standardiserte dokumentasjonsplattformer og prosjekteringsverktøy. I enkelte tilfeller har vi allerede fungerende systemer som vi har bygget ut der dette var nødvendig. I andre tilfeller måtte hele systemet bygges opp, maler og retningslinjer utarbeides, og godkjennes på styremøtet.

1.4 Utførelse og Prosjektmodell

Vi hadde statusmøter flere ganger i uken mens arbeidet foregikk, opplyste OV på mandagsmøter om vår progresjon, møter med veileder ca. to ganger i måneden, og enkelte veiledningsmøter med erfarne seniorer i OV etter behov. For å kunne sikre fremgangen og sporbarheten til kravene i prosjektet er det benyttet en prosjektmodell. Vi har benyttet V modellen. Dette er en enkel modul mye bruket i industrien.

Den består av to sider, en utviklings side og en verifisering side. For hvert nivå man tar med utvikling, verifiseres det.

1.4.1 Work breakdown diagram

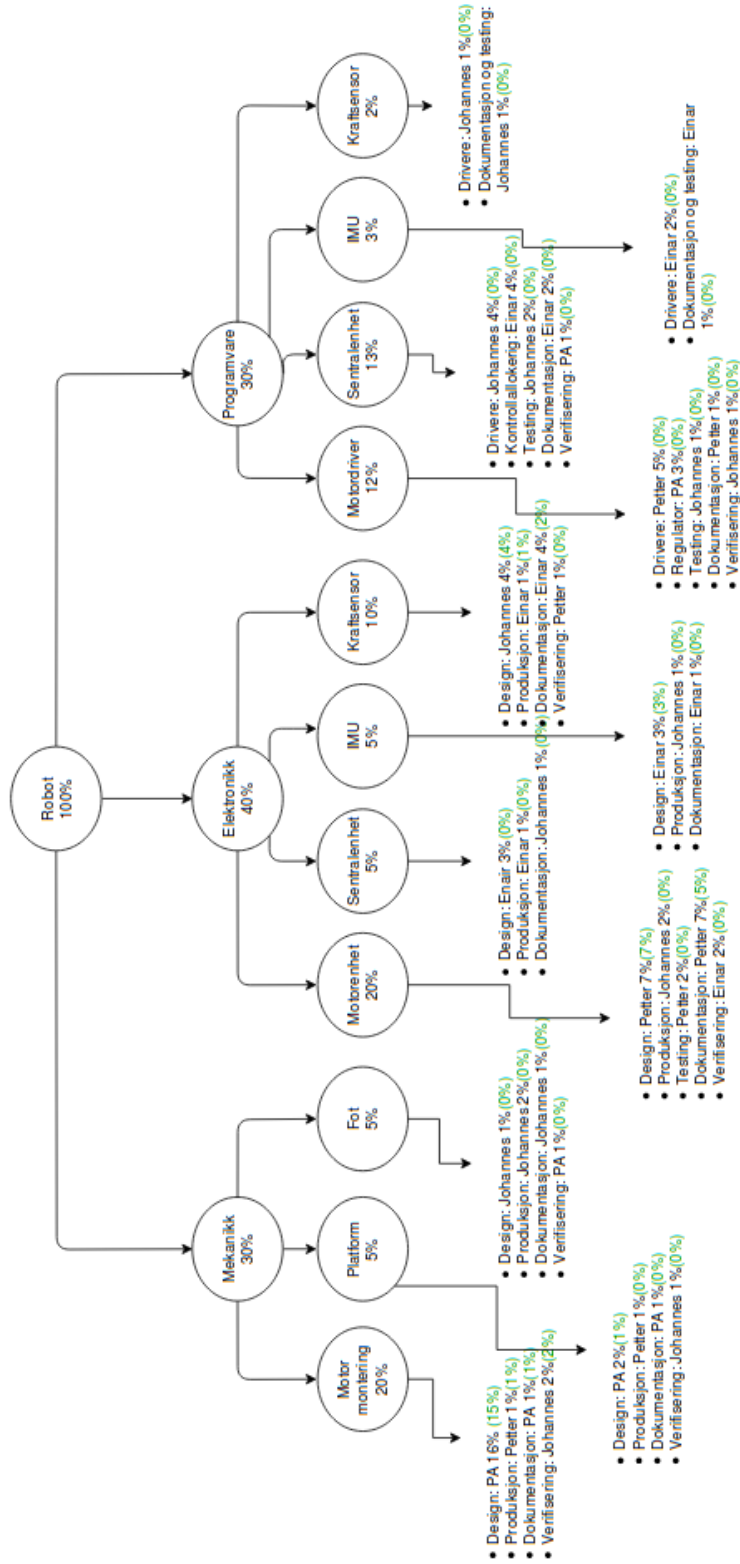
Prosjektet ble delt opp etter kategoriene mekanikk, elektronikk og programvare. Dette ble satt inn i et work breakdown diagram, som gir en oversikt over hvor mye arbeid som er gjort, arbeidsmengden til de ulike delene av prosjektet. Det forteller også hvem som skal utføre oppgaven. I figur 1.4 kan et slikt en se diagrammet omtrent mitt i utviklingsprosessen.

Diagrammet hjelper både prosjektgruppen til å organisere seg, og til å kommunisere arbeidsfordelingen til veileder.

1.5 Markedsanalyse for moduler

Basert på tidligere salg av elektromekaniske moduler som finnes i Omega Verksteds sortiment fant vi veiledende priser på modulene vi utviklet. Ved ekstrapolering av salg av tilsvarende komponenter i databasen over alle salg og enkle beregninger, antok vi at en pris på ca. 250 NOK ville være overkommelig for en motormodul. IMU og kraftsensor-modulene ble ansett som mindre betydningsfulle som hyllevarer for Omega Verksted, og ble ikke videre analysert prismessig. Prisen på en motormodul var utslagsgivende for valg av motor. Motormodulen som arbeidsgruppen utviklet i løpet av prosjektet antas for å fylle et hull i Omega Verksteds motorutvalg.

Work Breakdown Diagram



Figur 1.4: Work breakdown diagram fra midten av prosjektet

Kapittel 2

Matematisk Modell

I denne delen beskrives modelleringsprosessen, forenklinger for å kunne beskrive tekniske krav til beregningshastighet, motorkraft, krav til RPM for motorer og liknende. Meste parten av innholdet i dette kapitlet er basert på eller hentet direkte fra [13], som ble utarbeidet samtidig som prosjektet foregikk av en av forfatterne av denne rapporten. For en mer detaljert beskrivelse av tematikken i dette kapitlet ber vi leseren undersøke nevnte prosjektrapport.

2.1 Innledning

Selv om vi kommer til å lage roboten modulbasert slik at antall aktuatorer og plasseringen kan endres har vi fokuset på å lage en *tri-pod*, en plattform med tre bein, symmetrisk plassert til hvert sitt hjørne av en likesidet trekant. Hvert bein skal ha et skulderledd montert i plattformen og et kneledd. Hver aktuator skal ha en enkoder som måler vinkelen i tillegg til at hver fot skal ha en kraftsensor, og plattformen skal utstyres med en IMU. Det er ikke sikkert at alle disse sensorene blir nødvendige for at roboten skal være i stand til å gå, men dersom forskjellige algoritmer skal benyttes i fremtiden kan det være nyttig å ha muligheten til disse målingene.

2.2 Initielle antagelser

For å kunne velge sensorer og aktuatorer til en fysisk robot er det vanlig å ha en grundig basis i teoretiske studier og simuleringer. Siden hele dette prosjektet ble gjennomført på et semester hadde vi ikke muligheten til å gjøre en veldig grundig analyse før modulene ble designet. Estimerer ga oss likevel en god indikasjon. I prosjekter av større skala, der det gjerne er et lite team med forskere som jobber med prosjektet i flere år er det gjerne også helt andre økonomiske rammer[2]. I denne seksjonen undersøker vi kravene til de betraktelig mye billigere komponentene som vi benyttet i vårt prosjekt.

Noen enkle beregninger ble gjort på forhånd slik at vi skulle kunne finne motorer som var egnet for utprøving:

$$\tau = \mathbf{r} \times \mathbf{F} \quad (2.1)$$

$$\tau_{max} = l_l(m_m + \frac{1}{2}m_{leg} + m_b)g \quad (2.2)$$

$$\ddot{\theta}I = l_l(m_m + \frac{1}{2}m_l + m_b)\sin\theta g \quad (2.3)$$

$$\omega_{max} = \frac{1}{I}l_l(m_m + \frac{1}{2}m_l + m_b)gt + \omega_0 \quad (2.4)$$

Ligning 2.1 til 2.4 gir forståelse for hvordan vekt-rotasjonsmoment-forholdet til motorene er. Det er også lett å utvide for et ytterligere ledd til. ω_{max} gir en pekepinn på hvor raskt vi kan forvente å bevege beina.

De mest sannsynlige gangartene som vi kommer til å prøve å implementere for denne roboten er gangarter slik som i [6] eller gangarter som forflytter massesenteret utenfor konfigurasjonspolygonet og dermed begynner å vippe. Den overordnede dynamikken kan ses på som en invertert pendel og hastigheter er gitt av pendelligningen i 2.5:

$$\ddot{\alpha} = -\frac{g}{l}\sin\alpha \quad (2.5)$$

$$T = 2\pi\sqrt{\frac{l}{g}} \quad (2.6)$$

Uten å ha en komplett matematisk modell med simuleringer er det vanskelig å si eksakt hva hastighetskravet til motorene er, men det er naturlig å tenke at vinkelhastigheten til beina burde være en god del raskere enn halvparten av en periodisk svingning, 2.6, for roboten. Lengde på ben kan også velges etter disse ligningene. Det er grunn til å tro at en økning i lengde på hvert ben l vil føre til

treger dynamikk, og dermed blir enklere å regulere. I ligning 2.6 antar vi at utslagsvinkelen er liten og $\sin \alpha \approx \alpha$.

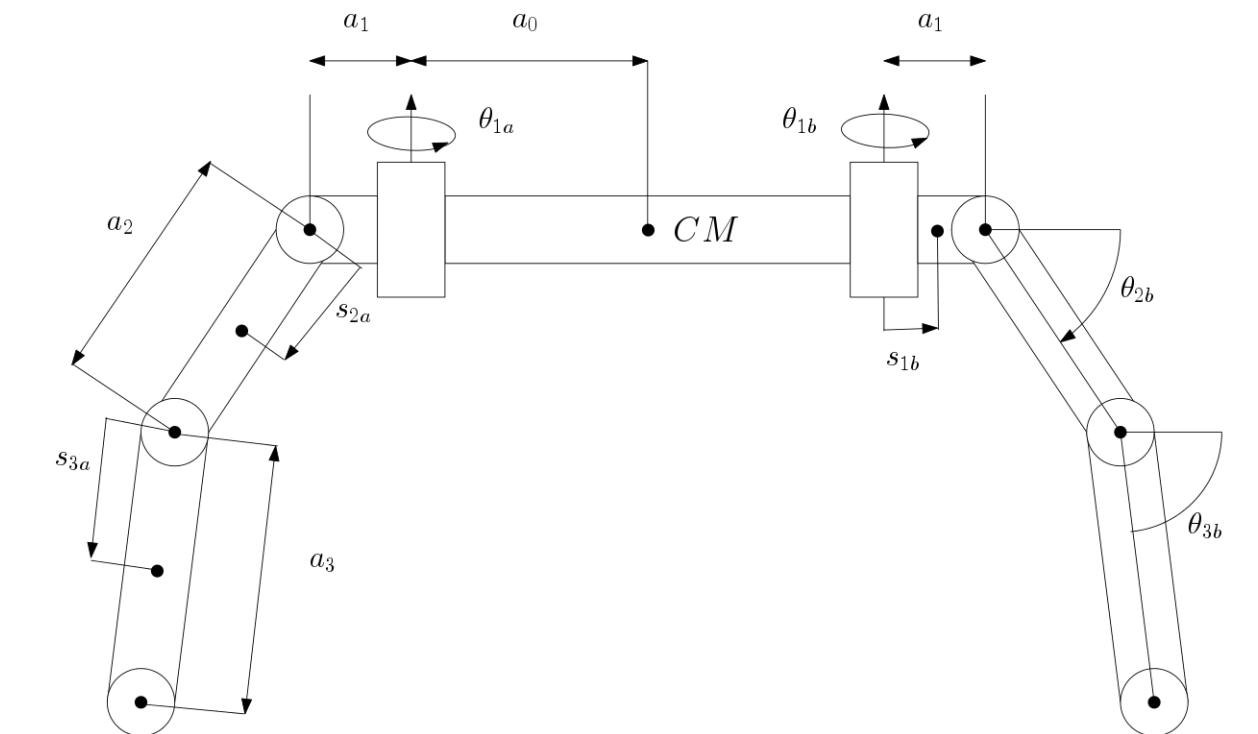
Liste med antagelser:

Disse antagelsene presiseres og kommenteres videre i analysen.

1. Elastisitet i materialer er neglisjerbar
2. Konstant plassering av massesenter
3. Ingen slark i girkasse
4. Eksakt vinkelverdi er tilgjengelig
5. Ingen tidsforsinkelser forekommer
6. Ingen ytre krefter
7. Ingen forekomst av skliing
8. Sammenstøt er fullstendig uelastiske
9. Bein regnes som punktmasser i føtter og kneledd
10. Bein regnes som masseløse
11. Massesenter befinner seg på midtaksen
12. Beregningstid er neglisjerbar

Uten å måtte forholde seg til noen av antagelsene kan vi med stor sikkerhet si at lengden på beina til roboten vil være bestemmende for hvor rask dynamikk vi vil få. Vi regner problemet med å stabilisere roboten i tilfellet der alle tre beina er på bakken som løselig ved bruk av *statisk stabilitet*, dvs. at massesenteret er plassert innenfor konfigurasjonspolynomet utspent av de tre føttene. Ved mindre enn tre bein i bakken vil roboten ha tilsvarende passiv dynamikk som en invertert pendel. Selv om vi kommer til å benytte leddene til å endre treghetsmomentet og massesenteret til roboten vil dynamikken til en invertert pendel gi oss en pekepinne på hvilke variabler som får størst betydning.

Konfigurasjonen i figur 2.1 er kjent fra [6], men gangarten som benyttes av *SemiQuad*-roboten kan ikke brukes hvis vi ønsker å alltid ha to bein på bakken. En modifikasjon kan forøvrig trolig benyttes.



Figur 2.1: en todimensjonal fremstilling av roboten, med beskrivelse av lengder, vinkler og massesentre på hvert ledd

2.3 Strategi for modellering og utarbeidelse av gangarter

Denne seksjonen og den neste kommer til å være preget av flere matematiske og kontrollteoretiske termer. Dersom leseren ikke er kjent med disse termene kan det anbefales å undersøke [18], [16], [17], [3] og [5] ettersom disse er sentrale innenfor temaet.

Det er mange muligheter når det kommer til ulike måter å modellere en slik robot på og hvordan finne algoritmer for å få roboten til å ha ønsket oppførsel. I utgangspunktet er målet med vår robot at du kan velge forskjellige måter, både å modellere og designe gangarter og regulatorer. En ikke uvanlig fremgangsmåte er feedforward control som [11]. Denne fremgangsmåten krever mindre ressurser selv om vårt mekaniske design er ganske annerledes, og at det dermed blir mer avansert. En fremgangsmåte som ble vanlig innenfor roboter på bein de siste femten årene er strategien som kalles *Hybrid Zero Dynamics* [2]. Stegene for å tilnærme seg et slikt design er sådan:

- Bestemme de generaliserte koordinatene for roboten
- Finne passive og aktive frihetsgrader for systemet og hvilke generaliserte koordinater som tilhører disse.
- Beskrive posisjonene til alle massesenterne til alle leddene med de generaliserte koordinatene. Gjerne i form av DH
- finne uttrykk for lineær og rotasjons-Jacobian. Dette vil si å finne uttrykk for hastighetene til massesenterne
- Beskrive kinetisk og potensiell energi for roboten
- Bruke Euler-Lagrange eller Newton-Euler metode for å finne bevegelsesligningene
- Beskrive fysiske og virtuelle begrensninger som generaliserte krefter som virker på de respektive generaliserte koordinatene
- Velge en relasjon av den passive dynamikken og beskrive den med standard robotligning
- Bestemme hva som er ønsket bevegelse ved å tegne opp faseportrettet til den passive dynamikken
- Bestemme en *bevegelsesgenerator-funksjon* med aktive koordinater som argumenter, og substituere inn i den passive dynamikken.

- Undersøke at den bevegelsesgeneratoren gir en løsning som bevarer 0-verdien til *bevegelsesintegralet*
- Lage en impulsmodell for roboten og beskrive når dette skal være den gjeldende modellen
- Undersøke at syklisk stabilitet oppnås ved at *Poincaré returtransformasjonen* for systemet eller det reduserte systemet ved HZD gjør initialbetingelsene invariante under impulsmodellen og den andre dynamikken
- Designe en regulator som gjør de virtuelle begrensningene og initialbetingelsene invariante. Denne blir en regulator for alle posisjoner og hastigheter til de aktuerte generaliserte koordinatene.

θ_{i_x}	koordinat for ledd nr. i på bein x der $x \in \{a, b, c\}$
ψ	yaw: Rotasjon om x-aksen på plattformen
ϕ	roll: Rotasjon om z-aksen på plattformen
θ	pitch: Rotasjon om y-aksen på plattformen
x_0	Koordinat i x-retning for plattformen
y_0	Koordinat i y-retning for plattformen
z_0	Koordinat i z-retning for plattformen
ω	Vinkelhastighet
\mathbf{v}	Lineær hastighet
J_{ω_i}	Jacobian for vinkelhastighet i ledd nr. i
J_{v_i}	Jacobian for lineær hastighet i ledd nr. i
K_p	Kinetisk energi for plattformen
P_p	Potensiell energi for plattformen
K	Kinetisk energi
P	Potensiell energi
q	Generalisert koordinat
λ_i	Lagrange-multiplikator
f_i	Generalisert kraft
R_i	Rotasjonsmatrise i ledd nr. i
m_i	Massen til ledd nr. i
g	Tyngdeakselerasjonen
o_b^a	Lineær koordinat i transformasjonen fra koordinatsystem a til b

Tabell 2.1: Oversikt og forklaring av symboler som blir brukt i seksjonen

2.4 Oversikt

En oversikt over alle variabler kan ses i tabell 2.1 .

Siden orienteringen ikke oppfyller kravet om at x-aksen i plattformens koordinatsystem og z-aksen i det initielle koordinatsystemetnødvendigvis krysser kan vi ikke bruke DenavitHartenberg konvensjon på den homogene transformasjonen fra bakken til plattformen. På de andre leddene er begge *DH-betingelsene* oppfylt, og vi kan bruke fire variabler til å beskrive leddene. Disse fire variablene beskrives i Denavit-Hartenberk konvensjonen som Rotasjon om z-aksen, translasjon i z-aksen, translasjon i den nye x-aksen og til sist rotasjon om den nåværende x-aksen. Hvert ledds koordinatsystemer kan å beskri-

Link	a_i	α_i	d_i	θ_i
0	a_0			θ_{0_i}
1_i	l_1	$\frac{\pi}{2}$		$\theta_{1_i}^*$
2_i	l_2			$\theta_{2_i}^*$
3_i	l_3			$\theta_{3_i}^*$

Tabell 2.2: Denavit-Hartenberg-diagram fra det initielle aksekorset på overflaten til bein nr. i

ves konsistent med de fire variablene i diagrammet 2.2. De matematiske uttrykkene som komprimeres i DH-diagrammet realiseres som *homogene transformasjoner* som finnes i *Spesiell Euklidisk gruppe* eller $\mathcal{SE}(n)$. [18].

En generell homogen transformasjon:

$$\begin{pmatrix} c(\theta) c(\phi) & c(\phi) s(\theta) s(\psi) - c(\psi) s(\phi) & s(\phi) s(\psi) + c(\phi) c(\psi) s(\theta) & x_0 \\ c(\theta) s(\phi) & c(\phi) c(\psi) + s(\theta) s(\phi) s(\psi) & c(\psi) s(\theta) s(\phi) - c(\phi) s(\psi) & y_0 \\ -s(\theta) & c(\theta) s(\psi) & c(\theta) c(\psi) & z_0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Denavit-Hartenberg transformasjon:

$$\begin{pmatrix} c(\theta^*) & -c(\alpha^*) s(\theta^*) & s(\theta^*) s(\alpha^*) & a c(\theta^*) \\ s(\theta^*) & c(\theta^*) c(\alpha^*) & -c(\theta^*) s(\alpha^*) & a s(\theta^*) \\ 0 & s(\alpha^*) & c(\alpha^*) & d \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Uttrykkene for kinetisk og potensiell energi beskrives ved hjelp av

$$K = \frac{1}{2} \dot{\mathbf{q}}^T \left[\sum_{i=1}^n \{m_i J_{v_i}(q)^T J_{v_i}(q) + J_{\omega_i}(q)^T R_i(q) I_i R_i(q)^T J_{\omega_i}(q)^T\} \right] \dot{\mathbf{q}} \quad (2.7)$$

$$P = \sum_{i=1}^n m_i g r_{ic} \quad (2.8)$$

$$(2.9)$$

fra [18]. Her er lineær og vinkelhastighets-Jacobian beskrevet med

$$J_\omega = [\rho_1 z_0 \cdots \rho_n z_{n-1}] \quad (2.10)$$

$$J_v = \left[\frac{\partial o_n^0}{\partial q_0} \cdots \frac{\partial o_n^n}{\partial q_n} \right] \quad (2.11)$$

Vi kan finne bevegelsesligningene for systemet ved å benytte *Lagrange-mekanikk* [18]

$$\mathcal{L} = K - P \quad (2.12)$$

$$\frac{d}{dt} \left[\frac{\partial \mathcal{L}}{\partial \dot{q}_i} \right] - \frac{\partial \mathcal{L}}{\partial q_i} = \sum \lambda_i \frac{\partial f_i}{\partial q_i} \quad (2.13)$$

Ved hjelp av Euler-Lagrange-ligningen, 2.13, fikk vi femten ligninger der seks var passive og ni var beskrevet av vinkelmomentene τ_{i_a} to τ_{i_c} .

Det neste steget er å introdusere begrensninger, både fysiske og virtuelle, for å redusere dynamikken og oppnå ønsket adferd[16].

bla bla

Vi vil nå ha minst en ligning som kan skrives på formen

$$\alpha(q)\ddot{q} + \beta(q)\dot{q}^2 + \gamma(q) = 0 \quad (2.14)$$

For å kunne bestemme bevegelsen til roboten må vi finne en bevegelsesgenerator i den passive dynamikken, beskrevet av aktuerte generaliserte koordinater og vise at denne bevegelsesgeneratoren er en mulig løsning. Dette kan gjøres ved å vise at bevegelsesgeneratoren medfører bevaring av nullverdien til *bevegelsesintegralet* 2.17, for den skalare robotligningen, 2.14, til den passive dynamikken. Det stilles også andre krav til den skalare robotligningen for den passive dynamikken og løsningen $r(t)$. Følgende kriterier må også stemme [16]:

$$\alpha(q_i) = 0 \implies \alpha(q_i \pm \epsilon) \neq 0 \quad | \quad \epsilon > 0 \quad (2.15)$$

$$r(t) \in \mathcal{C}^1 \quad (2.16)$$

Bevegelsesintegralet avslører om løsningen vil føre systemets tiltenkte passive, generaliserte koordinat

fra initialtilstanden til endetilstanden.

$$I(r, \dot{r}, r_0, \dot{r}_0) = r^2 - \psi(r_0, r) \left[\dot{r}_0^2 - \int_{r_0}^r \psi(s, r_0) \frac{2\gamma(s)}{\alpha(s)} ds \right] \quad (2.17)$$

$$\psi(r_0, r_1) = \exp \left\{ -2 \int_{r_0}^{r_1} \frac{\beta(\tau)}{\alpha(\tau)} d\tau \right\} \quad (2.18)$$

En måte å finne en bevegelsesgenerator på er å lage en relasjon som en funksjon av typen

$$r_p(t) = \phi(r_{a_1}(t), r_{a_2}(t), \dots, r_{a_{n-1}}(t)) \quad (2.19)$$

med

$$\dot{r}_p = \dot{\phi} \quad (2.20)$$

$$\ddot{r}_p = \ddot{\phi} \quad (2.21)$$

og substituere det inn de aktive ligningene for å finne uttrykk for kreftene dersom bevegelsesintegralet, 2.17, førte frem. Ofte kan et polynom som interpolerer ønsket kurve i faseportrettet være en gunstig løsning. Å finne en løsning på formen $\phi(t)$ er generelt vanskeligere.

Selv om vi nå i teorien kan planlegge en bevegelse for å styre de uaktuerte koordinatene i dynamikken på ønskelig vis, vil en ganske ofte inkludere en impulsmodell, siden Lagrange-meknikken ikke greier å beskrive det uelastiske støtet når en fot settes i bakken med hastighet ulik null. Her gjelder det å benytte Newtons impuls-lov for å finne de nye hastighetene etter sammenstøtet basert på hastighetene rett før støtet.[5] Vi ønsker oss en modell på formen:

$$x^+ = \Delta(x^-) \quad (2.22)$$

hvor

$$x^+ = (q^+, \dot{q}^+) \quad (2.23)$$

$$x^- = (q^-, \dot{q}^-) \quad (2.24)$$

er vektoren av posisjoner og hastigheter før (-) og etter (+) sammenstøtet.

Den totale dynamikken blir nå på formen

$$\dot{x} = f(x) + g(x)u, \quad x^- \notin \mathcal{S}, \quad (2.25)$$

$$x^+ = \Delta(x^-), \quad x^- \in \mathcal{S}, \quad (2.26)$$

hvor

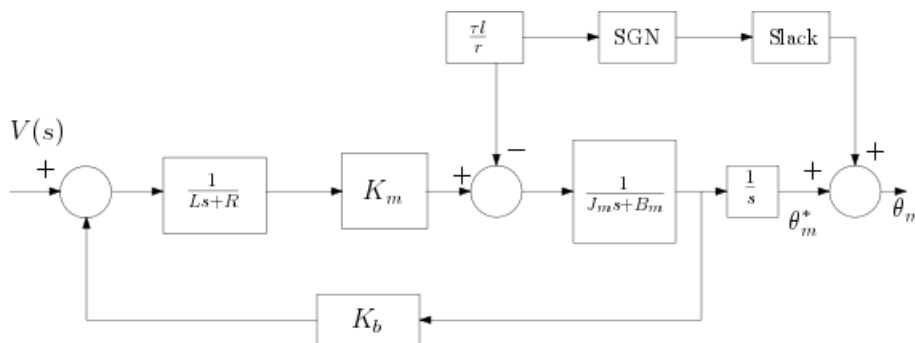
$$\mathcal{S} := \{(q, \dot{q}) \mid d(\mathbf{z}_{end}, \mathcal{P}(\mathbf{q})) = 0\} \quad (2.27)$$

er den aktive mengden når ben(a) treffer bakken, \mathcal{D} , i punktet \mathcal{P} , $\mathcal{P} \subset \mathcal{D}$ og δ er impulstransformasjonen dette kalles en hybridmodell. For at stabil syklisk gange skal oppnås må en regulator gjøre de virtuelle begrensningene invariante og Poincaré returtransformasjonen må være slik at de nye initialtilstandene til den dynamiske modellen gjensker endetilstander som igjen sørger for initialtilstander innenfor det invariante området.

2.5 Modell av motormodul

I denne seksjonen vil vi gå nærmere inn på motormodulen som skal benyttes å prøve å beskrive en mer detaljert ulinær dynamikk som vi kan forvente av en realistisk motor med girkasse og enkoder(e).

I figur 2.2 ser vi den generelle modellen for en motor med girkasse. I vårt tilfelle vil det også være en enkoder. Denne utvidede ulineære dynamikken bør inkluderes ved videre arbeid. Et slikt ulineært system vil ha mange avhengigheter. Ønsker vi å lage et system der regulatorparameterene ikke trenger å oppdateres manuelt for hver gang lasten på motoren endres kan det implementeres en adaptiv regulator. Detaljer om motormodellen kan ses i [3].

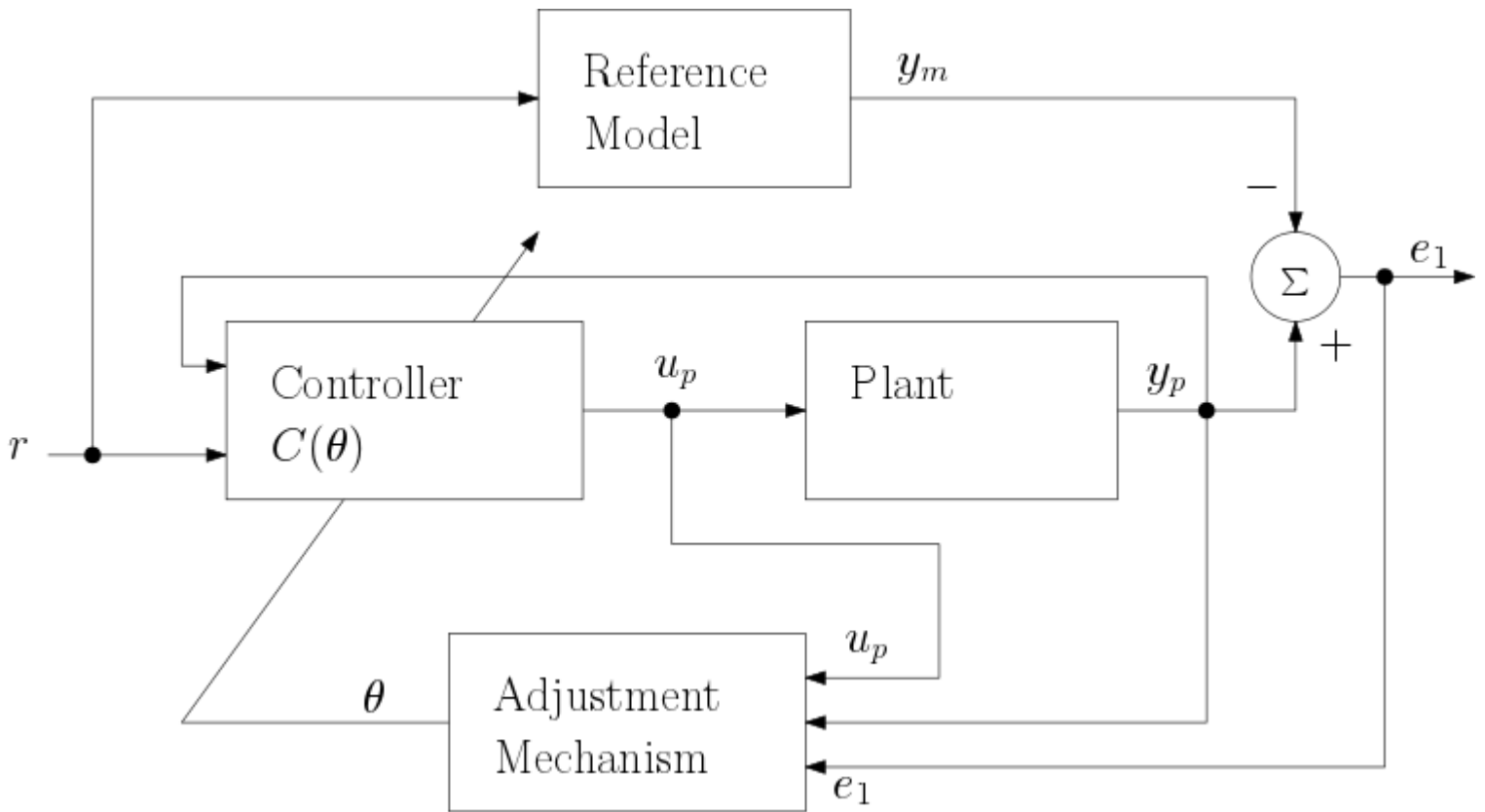


Figur 2.2: Bokkdiagram over generell motor med girkasse-modell hentet fra [3]

2.5.1 MRAC for motormodul

Model Reference Adaptive Control er en vanlig måte å implementere adaptiv regulering [8]. Gitt at systemet har minimum fase, at alle nullpunkter er stabile, kan denne metoden benyttes. Blokkdiagrammet

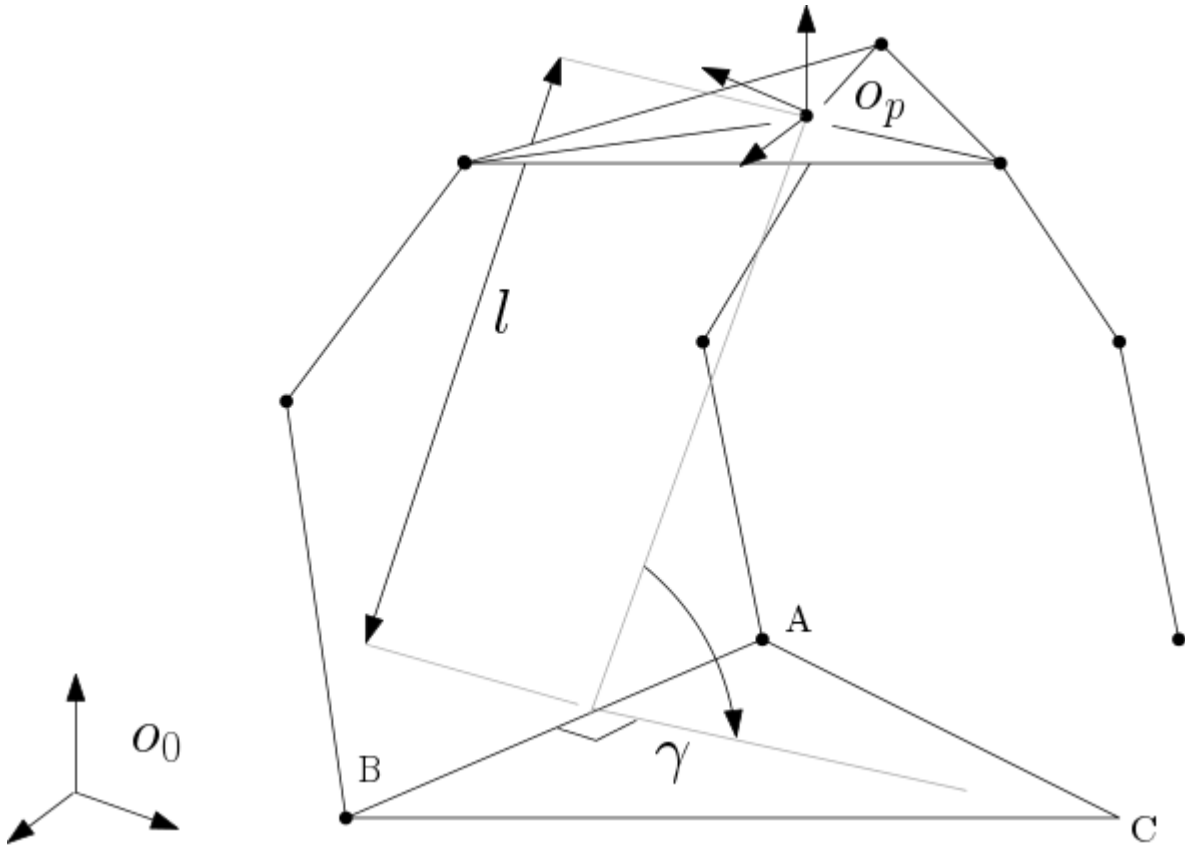
i figur 2.3 viser det generelle blokkdiagrammet for MRAC. Forfatterne anbefaler å implementere denne adaptive reguleringsløyfen ved videre arbeid. Figur 2.3 er hentet fra [8].



Figur 2.3: Blokkdiagram over generell "Model Reference Adaptive Control"

2.5.2 Utvikling av gangart

Ved gange er bevegelse av et løftet bein den eneste muligheten gitt antagelsene tidligere i kapitlet. I figur 2.4 er en slik bevegelse beskrevet. For å skåne elektronikken og de mekaniske delene, samt unngå forstyrrelser ved ikke-optimale hastigheter ved impuls, ønsker vi at hastighetene i systemet skal være små ved sammenstøt. En slik bevegelse er beskrevet i faseportrettet 2.5. Setter vi sammen mange slike bevegelser kan vi oppnå syklisk gange for roboten som illustrert i figur 2.6.



Figur 2.4: Bevegelsesgeneratoren γ ved to-bein-stilling

Ligningene for roboten i to-beins-posisjon kan sammenlignes analyseres med:

$$\ddot{\gamma} = \frac{gml}{I} \cos \gamma \quad (2.28)$$

$$E_p = \frac{1}{2} ml^2 \dot{\gamma}^2 + mgl \sin \gamma \quad (2.29)$$

$$\dot{\gamma} = \sqrt{\frac{2}{l} \left[\frac{1}{2} l \dot{\gamma}_0 + g(\sin \gamma_0 - \sin \gamma) \right]} \quad (2.30)$$

$$\dot{x}_p = l \dot{\gamma} \cos \gamma \quad (2.31)$$

$$\dot{z}_p = l \dot{\gamma} \sin \gamma \quad (2.32)$$

Disse ligningene kan brukes til å løse integraet beskrevet i ligning 2.17.

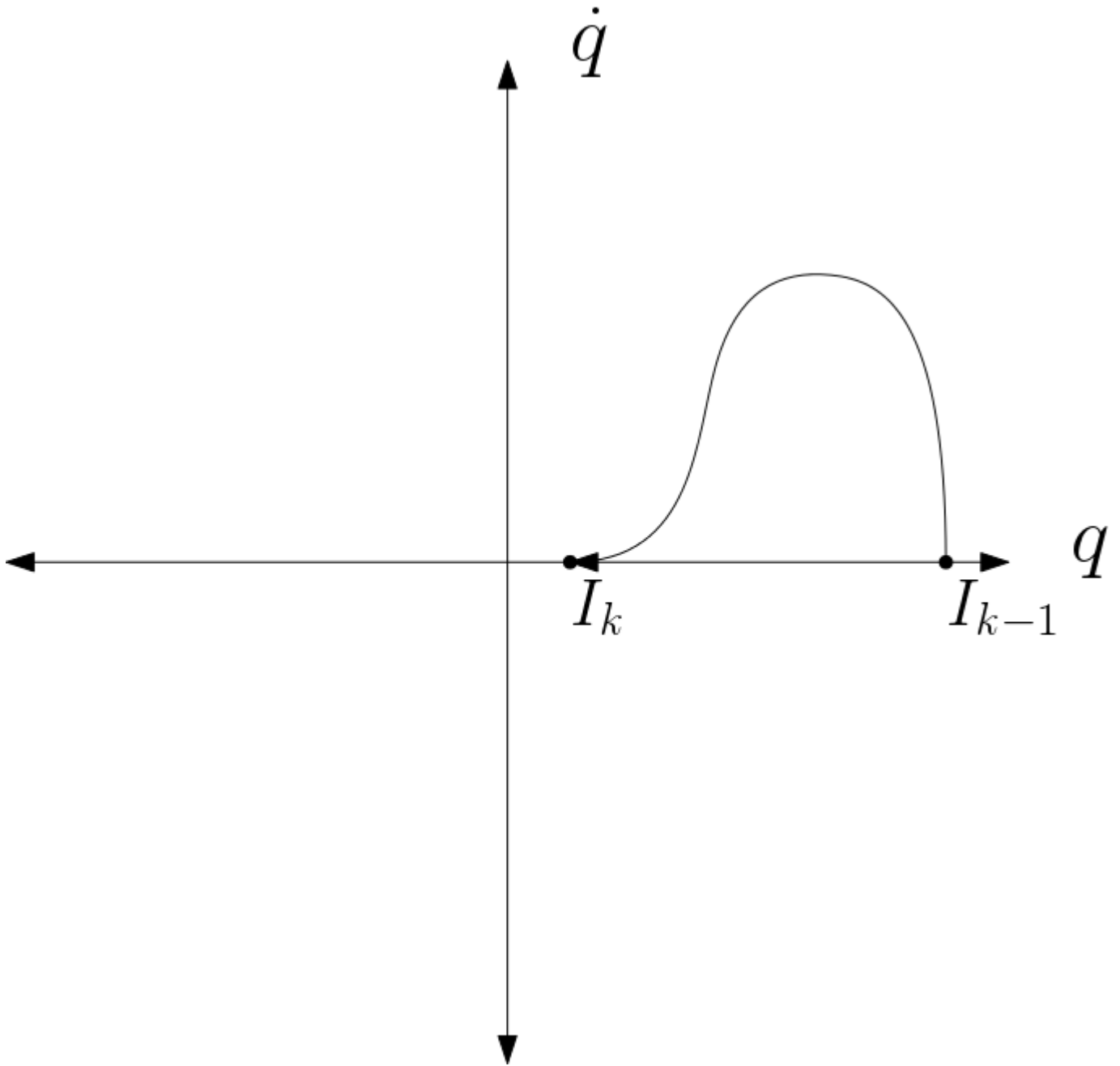
2.6 Observabilitet og Styrbarhet

Gitt at enkoderne på hvert bein er eksakte og det går an å måle IMU-data er nok til at modellen med antagelsen om ingen slark i girkasser er observerbar. Ved å legge til ekstra sensorer, måle strømtrekk og bruke filtre kan imidlertid gode estimater for den ulineære modellen med slark utvikles, og effektene kan bli minimale.

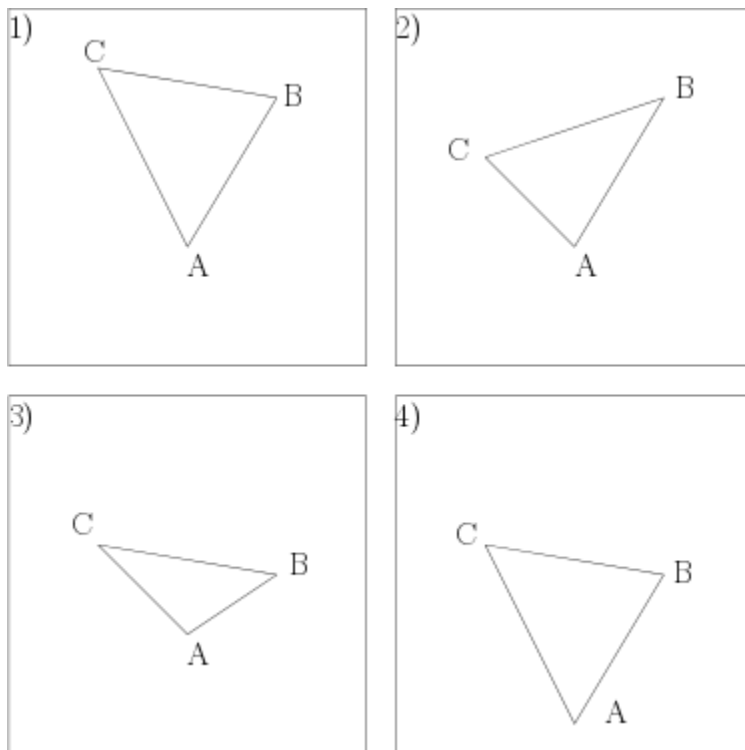
Ved tre bein i bakken er roboten overaktuert. Dersom roboten står på to eller et bein vil robotens dynamikk være underaktuert.

2.7 Identifikasjon

Med tanke på at antagelsene vist i 1 til 12 ofte ikke er virkelige, og at den virkelige modellen nesten aldri har de eksakt samme parameterne, er det ofte nyttig å gjøre parameteridentifikasjon. Dette kan gjøres ved å benytte metoder som *subspace identification* [15], eller en minste kvadraters-identifisering [8].



Figur 2.5: Ønsket bevegelse av et bein fra posisjonen I_{k-1} til I_k



Figur 2.6: En mulig strategi for bevegelse av et ben om gangen for å oppnå fremdrift

Kapittel 3

Litteraturstudie

I denne delen presenteres noen av de viktigste delene av det vi fant ut gjennom litteraturstudiet som foregikk i starten av prosjektet. I slike prosjekter som dette er det vanlig å begynne med den matematiske modelleringen og arbeidet med å finne passende reguleringsalgoritmer og gangarter lenge før den fysiske roboten utvikles. Siden vi i dette prosjektet ikke har hatt tid til å vente til den matematiske modellen er ferdig har vi måttet ta noen antagelser på forhånd. En av antagelsene er at delene i denne seksjonen ville inngå i byggingen av roboten, og kunne studeres før den matematiske modellen var komplett.

3.1 Tidligere arbeider om trebeinte roboter

Trebeinte roboter er ikke så populære som naboene med to eller fire bein. Hovedgrunnene for at tobeinte roboter er interessante å analysere har med likheten mellom tobeinte roboter og mennesker. Når det kommer til firebeinte roboter begynner det å bli industrielle applikasjoner, men disse er ganske sjeldne. Det er likevel noen anvendelser for den mindre vanlige trebeinte robottypen. Et omfattende forskningsprosjekt som prøver å avdekke hvorfor dyr ikke har tre bein produserer forskjellige familier av trebeinte roboter, og bruker forskjellige genetiske eller tradisjonelle algoritmer for å generere gangarter [11]. Industrielle roboter i som skal gjøre mikroskopiske bevegelser men kunne masseprodusere ble utviklet av en gruppe forskere, der tre bein ble benyttet [10]. Disse ble kalt *Nanowalkers*. Det gjenstår å se om studie av trebeint robot kan være relevant for analyse av gange for mennesker som har bevegelsesproblemer og dermed bruker stokk eller andre hjelpemidler.

3.2 Sentralenhet

De mest aktuelle sentralenhetene var Raspberry Pi og Beaglebone. Begge er egnet som sentral prosesseringsenhet i et komplekst elektronisk system. Beaglebone har innebygd støtte for real-time, og er derfor ofte brukt i robotprosjekter. Beaglebone har også god støtte for RS485 og CAN.

3.3 Aktuatorer

Valg av aktuator er en viktig del av roboten som i stor grad er uavhengig av styringsystemet. Videre vil roboten aldri kunne fungere bedre enn det aktuatorene tillater. Ett vanlig valg for aktuatorer til små roboter av dette slaget er hobby servoer, da de er relativt billige, lette å styre og kan fåes i mange størrelser og styrker. Servoer ble ikke brukt, da tilbakekoblingen på en ferdig servo er intern og derfor ikke tillater eksperimenter med egne regulatorer. Steppermotorer og børsteløse DC motorer ble vurdert, men er dyre og kompliserte i drift i forhold til de vanlige (børstede) DC motorene som ble valgt brukt.

Styrkene til en DC motor er at de er lette og billige å skaffe, er veldig enkle å drive og finnes i mange versjoner. Fordi de er ganske svake, finnes det mange ferdige moduler av motor med girkasse i forskjellige konfigurasjoner.

3.4 IMU

Roboten må kunne vite sin orientering i forhold til romkoordinater. Det er flere løsninger på dette, der noen er selvstendige og andre krever eksterne systemer. På grunn av kravene som er satt til selvstendighet er det ikke aktuelt å bruke eksterne systemer, som motion capture. Akselerometer sammen med gyroskop og kompass kan tilfredstille kravene.

Dataene fra de ulike sensorene har tilsammen 9 frihetsgrader, men orienteringen de måler bare har 6. Vinkel i 3 akser og vinkelhastighet i 3 akser. For å sammenføre disse dataene kreves det et sensorsammensettingsfilter. Til dette er ofte kalmanfiltere brukt.

Et alternativ til kalmanfilter er et filter utviklet av Madgwick [12]. Dette filteret krever mindre

resurser enn et kalmanfilter, og krever lavere oppdateringsfrekvens. Det gjør det spesielt godt egnet til implementasjon på mindre mikrokontrollere. Filteret har en asymptotisk nøyaktighet mot oppdateringsfrekvens-funksjon. Det vil si at 50 Hz gir omtrent samme nøyaktighet som 500Hz.

3.5 Kraftsensor/fotsensor

Roboten må ha en form for tilbakekobling om kreftene lemmene utsettes for. Det er flere mulige måter dette kan gjøres på. Lastsensor, strekkklapp, trykksensitiv motstand, knapp er eksempler på slike sensorer.

3.6 Enkodere og vinkelmålere

For å kunne styre motorene nøyaktig er det nyttig å ha flere forskjellige sensorer tilgjengelig for regulatoren. Noen motorer kommer standard med optiske eller magnetiske enkodere festet på motorakslingen. For å unngå regulator problemer med slark i girene, er det ønskelig å ha målere på selve leddet også. Typiske kan et potentiometer brukes til vinkelmåling. For høyere nøyaktighet kan dette kombineres med en optisk enkoder. Endestopbrytere kan også være aktuelle for å finne en nøyaktig "homeposisjon på ett ledd.

3.7 Mekanisk

I denne seksjonen beskrives de ulike mekaniske modulene og delene, materialvalg og fremgangsmetode. OV har fra før ikke hatt en godt dokumentert metodikk for fremstilling av dokumentasjon av mekaniske deler. I dette prosjektet ble det også utformet en mal for hvordan dette burde gjøres. Denne malen er egnet for SolidWorks, et utviklingsverktøy for 3D-tegning som brukes mye på Omega Verksted. Alle de mekaniske delene er tegnet i SolidWorks i dette prosjektet. Det ble også utviklet en dokumentasjonsside om retningslinjer for hvordan lage tekniske tegninger generelt og ved hjelp av SolidWorks. En dokumentasjonsside for hvordan man bør forholde seg til 3D-printede deler og toleranser ble også opprettet. Disse kan betraktes i vedlegg A. 3D-modellene av kretskortene ble eksportert fra Altium Designer uten videre etterprosessering.

3.8 Overordnet beskrivelse av arbeidet

Det ble produsert 17 tegninger av deler, der 3 kretskort, hadde ferdig 3D-modell, en elektrisk komponent hadde ferdig maskintegning, men ikke 3D-modell, og en motor som ikke hadde tegning i hverken to eller tre dimensjoner. De gjenværende delene ble designet og produsert med 3D-printer eller maskinert. Metodikken benyttet for å designe de ulike komponentene var basert på erfaring med delproduksjon fra tidligere prosjekter. Valgene for de ulike utformingene ble tatt underveis i prosessen etter som det ble mer og mer klart hvordan den generelle arkitekturen ble. Noen av valgene er delvis basert på retningslinjene som ble utarbeidet. Se utarbeidelsen av artikler i vedlegg A.

3.9 Valg av materialer

Fra tidligere prosjekter, som [14], har det vist seg å være enkelt og effektivt å bruke 3D-printede deler der det er kompliserte strukturer, og forskjellige karbonfiberstrukturer der det er bærende konstruksjon. På roboten vår tenkte vi initielt å bruke firkantrør i karbonfiber til bein, og 3D-printede deler ellers. Grunnen for å benytte firkantrør i karbonfiber er:

- Karbonfiber er lett og strekt
- Du kan få passende mengder med karbonfiber for en prislapp som er i trå med vår økonomiske ramme

- Firkantør er lett å montere deler på slik at ting ikke vrir seg, selv om de ikke er like sterke som rør.

Vi benyttet likevel ikke karbonfiber siden disse ikke er spesielt egnet til å bore i, og det viste seg å forenkle designprosessen og mulighet for enklere montasje og vedlikehold av roboten. Karbonfiber kan sprekke og flise seg opp ved boring, spesielt nært kanten. Derfor ble det avgjort at firkantør i aluminium var et bedre valg. For OV sin del er det også bedre muligheter for å bearbeide aluminium enn karbonfiber.

Materialeegenskapene til plasten FLGPCL03 kan ses i tabell 3.9. Selv om produsenten av 3D-printeren benyttet, [21], hevder at du kan ha tilstrekkelige toleranser til å 3D-printe gjenger på delene vi skal benytte, valgte vi likevel å gjøre dette etter 3d-printing med gjengetapp, for å forsikre oss om å få presise gjenger. Ved redesign bør det vurderes å legge inn slisser for muttere, i modellen. Dette gir økt kvalitet på gjengene. Noen av delene som brukes i prototypen til roboten burde ikke 3D-printes på denne måten. Ved resin-printing bygges delen opp av tynne lag som blir UV-bestrålt med en laser. Delene kan bli utsatt for knappe impuls krefter som kan gjøre at delene knekker langs nettopp disse lagene [14]. Dette er en svakhet som vi må leve med. Dersom delene skal produseres i mer egnede materialer, dvs. støpt eller maskinert plast, aluminium eller lignende, må de muligens redesignes. Videre produksjonsmetodikk blir ikke diskutert videre i denne rapporten.

Alle kablene mellom moduler som beveger seg ble valgt til å være fleksible silikonkabler med mange ledere for å kunne tåle gjentatt bøyning.

Materialeegenskaper	Størrelse
E-modul	$1.6GPa$
Forlengelse ved brudd	12%
Flytspenning	$38MPa$

Tabell 3.1: Materialeegenskaper for fotopolymeret FLGPCL03 [14]

3.10 Toleranser

Med tanke på produksjonsmetode og formål med roboten, var ikke strenge toleransekrav veldig vesentlig. Dermed ble NS-ISO 2768-1 M, generelle toleranser for ikke-toleransesatte mål, med grad medium,

Materialegenskaper	Størrelse
E-modul	134GPa
Flytspenning	1.65GPa
Massetetthet	1500 $\frac{kg}{m^3}$

Tabell 3.2: Materialegenskaper for karbonfiber hentet fra [14]

Materialegenskaper	Størrelse
E-modul	69GPa
Flytspenning	26GPa
Massetetthet	2700 $\frac{kg}{m^3}$

Tabell 3.3: Materialegenskaper for aluminium hentet fra [20] og [7]

benyttet på mange av delene[7]. I prinsippet kunne det vært fornuftig å spesifisere geometriske toleranser, men finpussing må uansett forekomme før montering pga. støttemateriale fra printing og liknende.

3.11 Plattform

Plattformen består av delene:

- Monteringsplate
- Senterfeste
- 3 stk. vektsensorer
- Forsenkede bolter

I tillegg monteres Beaglebone Black, IMU, vektcellemodul, og eventuelt batteri på monteringsplaten.

3.12 Skulderledd

Skulderleddet består av to moduler:

- Motormodul montert på firkantrør
- motormodul montert på vektsensor

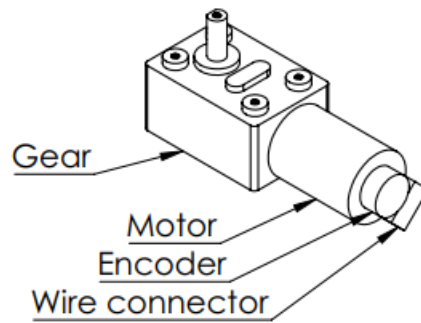
Disse er plassert i hvert av de tre hjørnene til plattformen. Hovedforskjellen på de to modulene er delene *motorbrakett 1* og *motorbrakett 2*. Den ene er for montering på vektsensor , mens den andre er for montering på firkantrør. Den komplette listen over delene i skulderleddene er:

- Motorbrakett 1
- Motorbrakett 2
- Benfeste x 2
- Monteringsring for optisk skive x 2
- Optisk skive x2
- Kretskort med posisjonspotmeter x 2
- Hovedkretskort for motormodul x 2
- Flens med akseltapp x 2
- Settskruer x 4
- Forsenkede bolter x
- Bolt og mutter x 2
- Firkantrør for sammenkobling
- Firkantrør til bein
- Motor med girkasse x 2

3.13 Kneledd

I sin helhet er kneleddet likt motormodul montert på firkantrør. Delelisten er dermed ganske lik:

- Motorbrakett 2

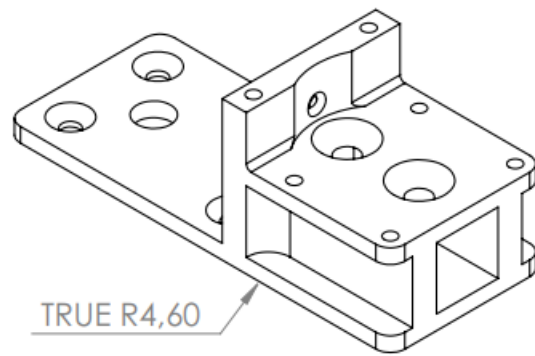


Figur 3.1: Motoren som vi valgte å bruke i alle leddene

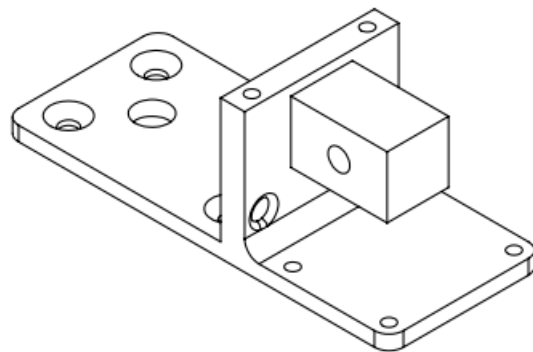
- Benfeste
- Monteringsring for optisk skive
- Optisk skive
- Kretskort med posisjonspotmeter
- Hovedkretskort for motormodul
- Flens med akseltapp
- Settskruer x 2
- Forsenkede bolter
- Bolt og mutter
- Firkantør til bein
- Motor med girkasse

3.14 Festing av kraftmålere

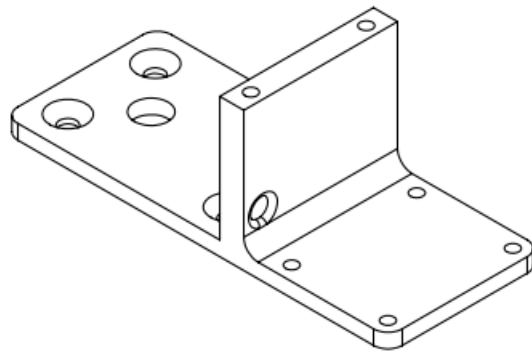
Vektsensorene som kan ses i 3.9 ble festet fra midten av senterfestet til hvert bein, som kan ses i A.4. Dette er for å kunne få en indikasjon av hvilke bein som er i bakken samt en kraftfordeling på de ulike beina. Begrensningen om at platformen skal være perpendikulær med underlaget gjør denne plasseringen av kraftmålere optimal.



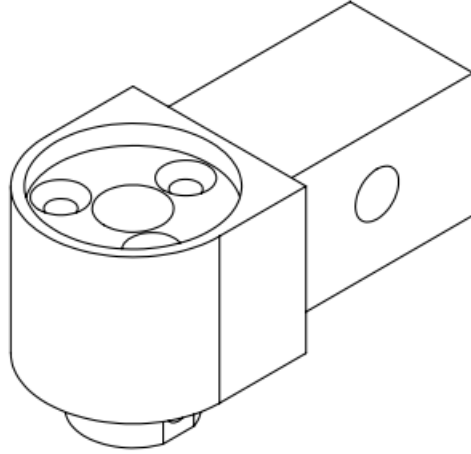
Figur 3.2: Motorbrakett 1; til montering på vektsensor



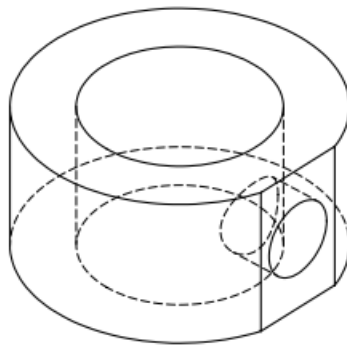
Figur 3.3: Motorbrakett 2; til montering på firkantrør



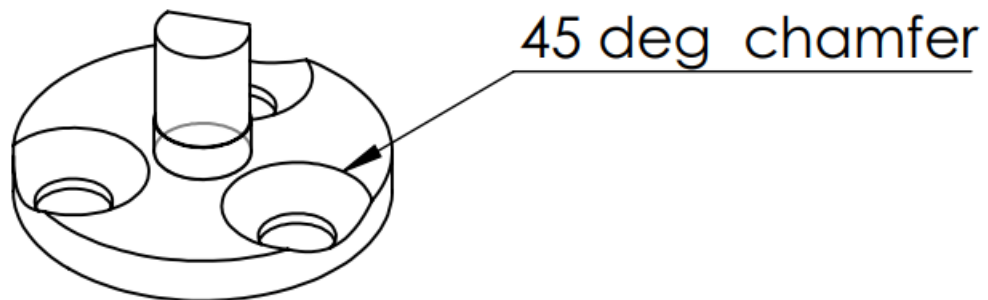
Figur 3.4: Motorbrakett 3; til mer generelle prosjekter



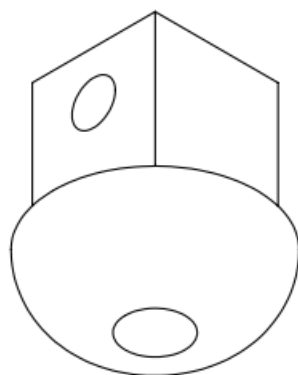
Figur 3.5: Benfeste for firkantrør



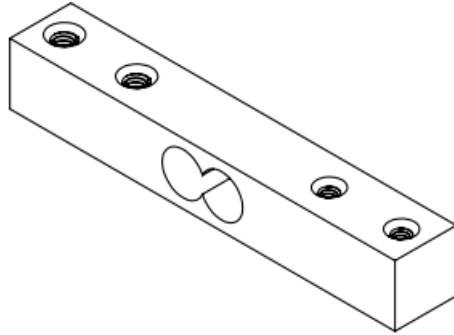
Figur 3.6: monteringsring for optisk skive for enkoder



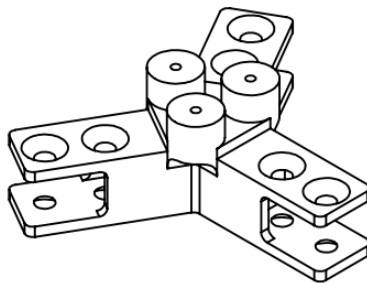
Figur 3.7: Flens med akseltapp for å passe inn i potmeter



Figur 3.8: Fot for roboten



Figur 3.9: Vektsensorer som monteres ut fra midten på plattformen



Figur 3.10: Senterfeste til vektsensorer og plattform

3.15 Kommentar til tekniske tegninger:

Alle tekniske tegninger kan finnes i A.4. De tekniske tegningene kommer i to formater. Det ene formatet, som forfatteren begynte med, er basert på standardinnstillinger i SolidWorks. Etter litt utprøving ble det utarbeidet en mal som er bedre tilpasset OV-prosjekter. I dette prosjektet er begge typene gode nok til å beskrive de nødvendige aspektene.

Kapittel 4

Elektronikk

Det er laget tre egenutviklede moduler i tillegg til en kommersielt tilgjengelig sentralenhet. Kretstegninger og utlegg for modulene finnes i vedlegg A.1.

Sentralt i designet av alle modulene er mikrokontrolleren. Den er lik for alle kortene for å øke mengden kode som kan gjenbrukes. De bruker en ATXMega64A3U mikrokontroller fra Microchip (tidligere Atmel). Denne ble valgt på grunn av at den har tilstrekkelig resurser, og at Omega Verksted har erfaring med å bruke den i prosjekter tidligere. I tillegg har verkstedet et betydelig lager med dem, som gjør produksjon billigere.

Kretstegninger og utlegg har blitt tegnet i Altium Designer, som er mye brukt på OV siden lisens er tilgjengelig gjennom NTNU. Verkstedet har over tid opparbeidet seg et bibliotek av komponenter til bruk i kretsdesign i Altium. De komponentene som har blitt brukt og som ikke allerede fantes i dette biblioteket har blitt opprettet, og symbol og footprint har blitt lagt til databasen. Dette gjelder for eksempel IMU-en, lastcelle-chippen og potensiometeret i hvert ledd i beinene. Det har til tider vært problemer med nedetid på NTNU sin Altium-lisensserver, som dessverre har ført til forsinkelser i arbeidet, da prosjektet ofte har vært avhengig av Altium.

Kretskort har blitt prototypet på OV, og deretter bestilt fra www.allpcb.com. Kretskortene derfra var både rimelige og av mer enn god nok kvalitet.

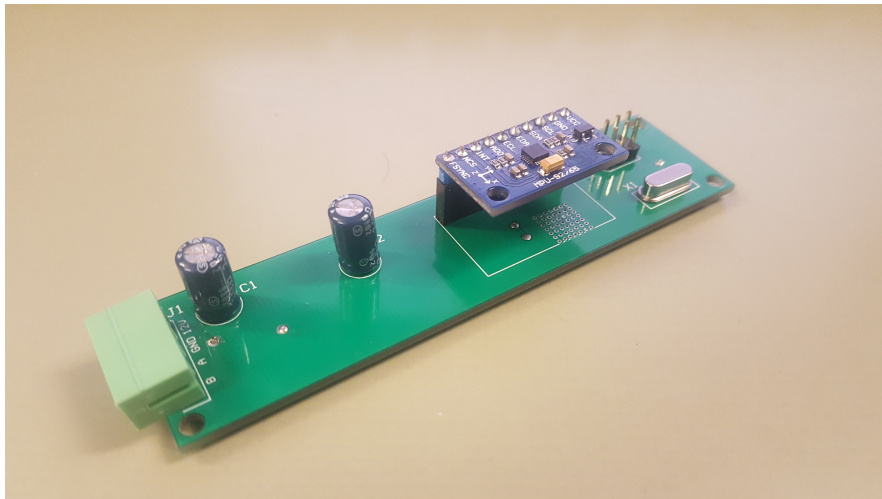
Siden modulene var utviklet til bruk på Omega Verksted, veide verkstedets standardutvalg tungt inn på komponentvalget. Deler som ikke var på OV's lager ble kjøpt inn fra verkstedets hovedleverandør Farnell (no.farnell.com). Spesifikke deler som sensor-breakouts og motorer ble kjøpt på eBay. Bill of

materials finnes i vedlegg A.3.

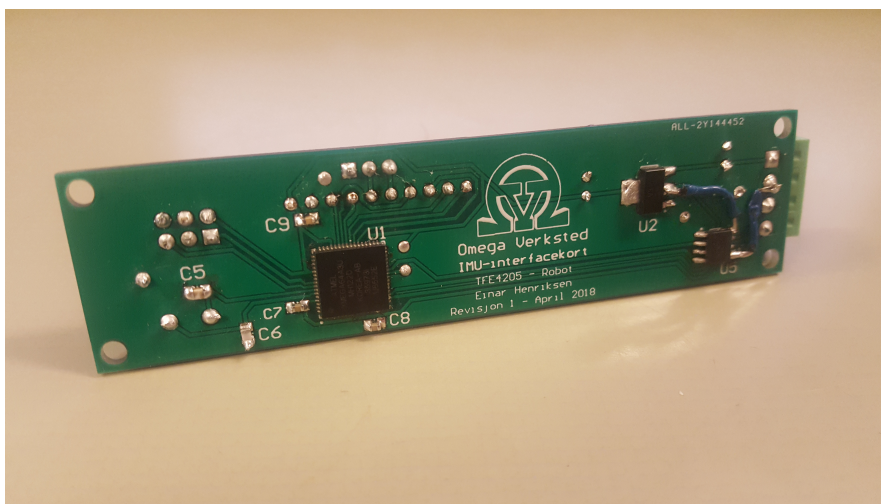
4.1 IMU

Denne modulens oppgave er å holde styr på robotens orientering i rommet. Dette er implementert med en IMU som inneholder akselerometer, gyroskop og kompass. Denne typen chip finnes det en del av på markedet fordi de er mye brukt i droner. IMU-en som blir brukt er MPU-9255 fra InvenSense. Denne brikken er godt støttet av åpne kildekodebiblioteker, som hjelper til å redusere systemets utviklingstid. Det ble brukt en breakout med de nødvendige passive komponentene på for at designet av modulen skulle bli billigere og raskere. Måledata fra IMU-en leses over SPI.

Modulen bruker en ATXMega 64A3U som prosessor, og kommuniserer med robotens sentralenhet over RS485.



Figur 4.1: IMU-modulen



Figur 4.2: IMU-modulen

4.2 Lastcelle-måler

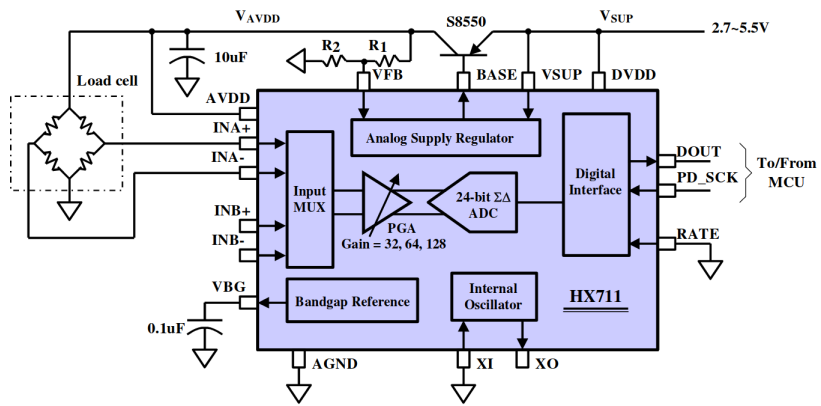
Modulens oppgave er å lese av en vektcelle festet til robotens ben. En vektcelle består av strekkklapper satt opp i en wheatstone bro. Når lastcellen utsettes for krefter vil spenningen til wheatstone broen endres med noen mV. For å lese av disse små endringene i spenning er det brukt en spesial IC laget for akkurat dette formålet. Det er valgt å bruke en spesial chip fordi den gir en både billigere, enklere og bedre løsning enn hva som kunne lages med generiske komponenter.

Kretsen er basert rundt den spesiallagde lastcelle-chippen HX711. Den har innebygget forsterker og adc. Grensesnittet mot mikrokontrolleren er en enkel seriell protokoll. Det ble valgt å bruke en ferdig integrert løsning til dette da det reduserer utviklingstiden, og reduserer risikoen ved prosjektet.

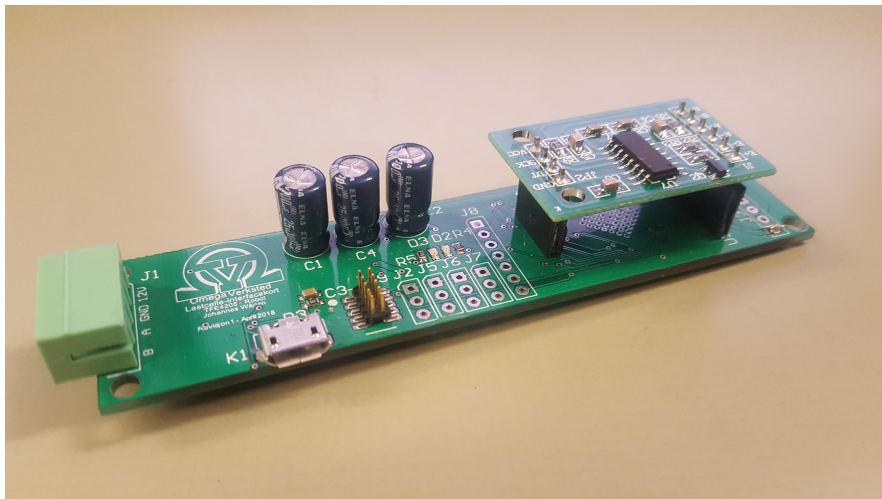
For å ytterligere redusere risikoen ble et ferdig laget kort for HX711 brukt. Dette kortet har alt av passive komponenter som chippen trenger.

Avlesingen og prosesseringen av data er gjort ved hjelp av en ATXMega 64A3U, som også er brukt i resten av systemet. Den kommuniserer videre med RS485 bussen.

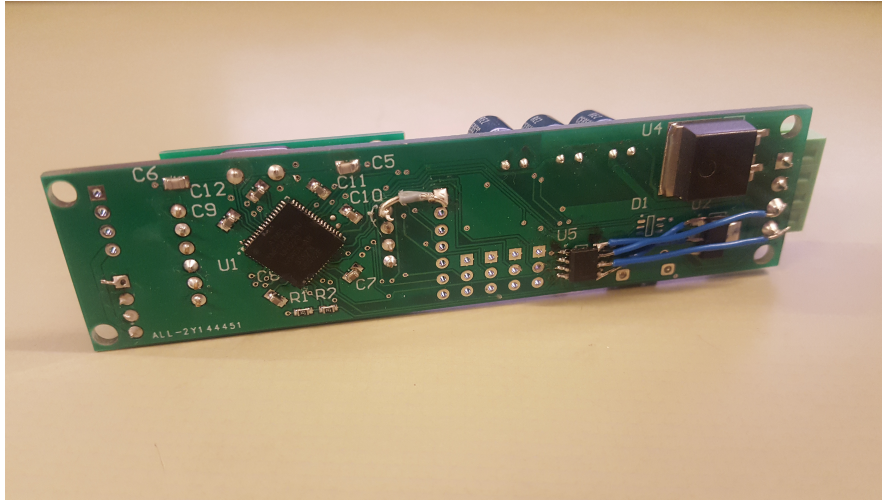
Kretskortet er designet for å være selvstendig og kun trenge strøm og RS485 for å fungere. Det har to lineære regulatorer som brukes for å redusere spenningen fra 12V til 3.3V og 5V. Det er valgt å bruke lineære regulatorer for redusere kompleksiteten, sammenlignet med å bruke mer effektive switch regulatorer.



Figur 4.3: Kretsdiagram av HX711



Figur 4.4: Lastcelle-modulen



Figur 4.5: Lastcelle-modulen

4.3 Motor

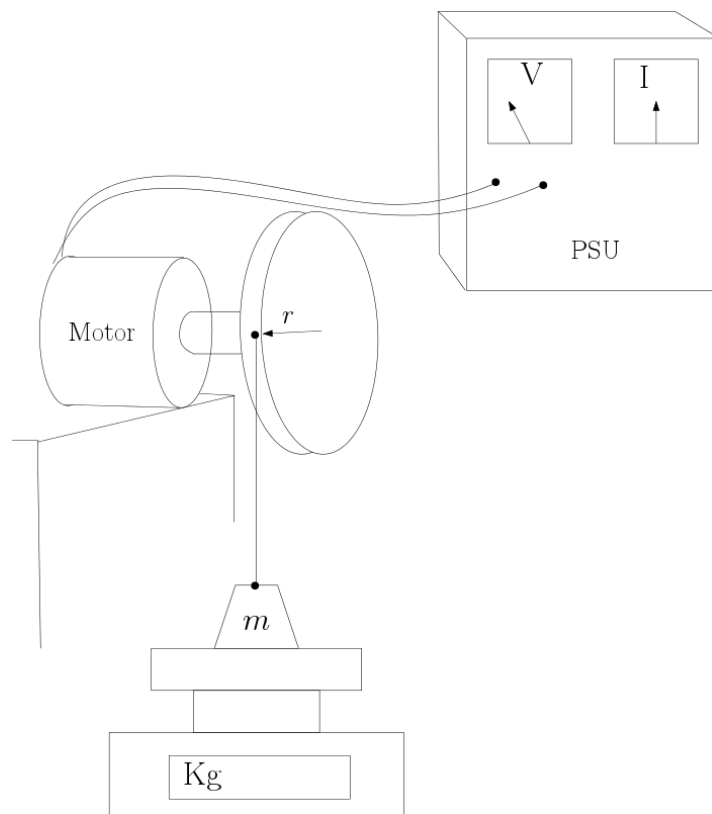
Motorene er blant de viktigste delene av roboten, da roboten aldri kan bli bedre enn det motoren klarer. Det er også en av de dyreste løse delene på roboten. Under litteraturstudie ble det allerede tydelig av pris og kompleksitetshensyn at valget kom til å falle på en giret DC motor. Valg av motor ble derfor en oppgave i å finne den billigste gira DC motoren som var god nok til oppgaven. Da den matematiske modellen for roboten ikke var kjent, ble beslutninger tatt på bakgrunn av estimat av forventet vekt og lengde på ben, uten at det var kjent om hastighet eller kraft var den viktigste faktoren.

Motoren som ble valgt er en giret DC motor på 12V med en utgangshastighet på 40RPM, med en magnetisk quadrature enkoder på motor akslingen. Motoren ble testet med et oppsett som vist i figur 4.6. Målet var å finne motorens torque, og dette ble gjort ved å se hvor mye av vekten til ett lodd den klarer å løfte opp fra en vekt. Motorens torque kunne da regnes ut som:

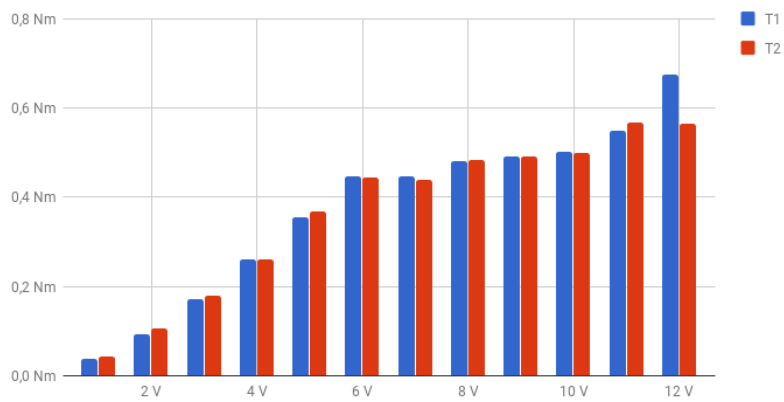
$$T_{motor} = (m_{lodd} - Vekt) * r_{hjul} * g \quad (4.1)$$

Radius på hjulet var $r = 0,050\text{m}$ og vekten som ble løftet var på $m = 2,001\text{Kg}$. Hvert løftble foretatt to ganger. Tabell 4.1 viser de målte resultatene. Dette gav et godt utgangspunkt for design av motordriver modulen, da vi ser at motoren ved nominell 12V spenning kan trekke opp mot 1,2A når akslingen holdes igjen. Momentet viste seg å være i underkant av det som var ønskelig, da skulderleddet har

vanskeligheter med å løfte hele beinet.



Figur 4.6: Testoppsett for testing av motor



Figur 4.7: Plot av motor torque ved spenning

Spenning [V]	Vekt 1 [Kg]	Strøm 1 [A]	Vekt 2 [Kg]	Strøm2 [A]	T1 [Nm]	T2 [Nm]
1,00	1,923	0,089	1,914	0,089	0,0382590	0,0426735
2,00	1,812	0,176	1,785	0,176	0,0927045	0,1059480
3,00	1,650	0,260	1,634	0,261	0,1721655	0,1800135
4,00	1,470	0,340	1,472	0,342	0,2604555	0,2594745
5,00	1,280	0,425	1,250	0,430	0,3536505	0,3683655
6,00	1,090	0,500	1,094	0,503	0,4468455	0,4448835
7,00	1,090	0,531	1,106	0,590	0,4468455	0,4389975
8,00	1,020	0,620	1,018	0,655	0,4811805	0,4821615
9,00	0,998	0,710	0,998	0,702	0,4919715	0,4919715
10,00	0,980	0,770	0,986	0,785	0,5008005	0,4978575
11,00	0,885	0,830	0,845	0,851	0,5473980	0,5670180
12,00	0,628	0,790	0,850	1,200	0,6734565	0,5645655

Tabell 4.1: Motor moment Spenning

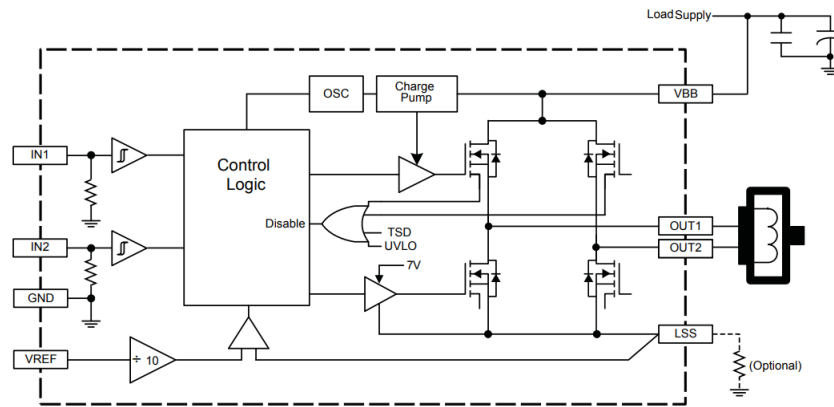
4.4 Motordriver

Denne modulens hovedoppgave er å drive motoren, lese flere motor-relaterte sensorer samt kommunisere over diverse protokoller. Vårt design bruker RS485 for kommunikasjon mellom enheter, men modulen har også støtte for kommunikasjon via UART, SPI, TWI og USB. Dette er for at den skal være mest brukbar også i andre applikasjoner. I konektorene for eksterne sensorer er det lagt opp støtte for 2 potmeterer (ADC innganger) og 1 quadrature enkoder. I tillegg kan kommunikasjonspinner som ikke er i bruk brukes som generisk GPIO, til f.eks. endestopp brytere. En lokal mikrokontroller (ATXMEGA64A3U) tar hånd om alle de motorrelaterte oppgavene.

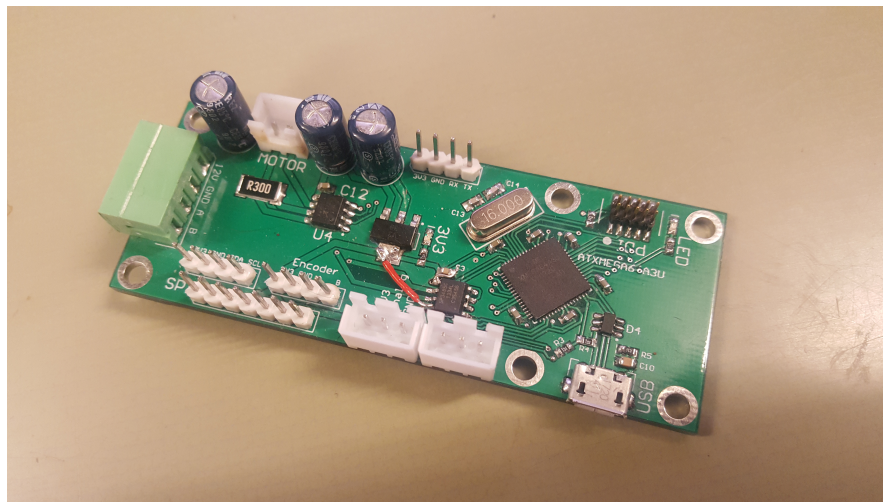
For å drive DC motoren brukes en H bro. For at motordriver modulen skulle bli så allsidig som mulig ble det satt noen krav til H-broen. Den må kunne takle et bredt spekter av spenninger og strømmer. Motor hastighet må kunne kontrolleres og strøm gjennom motoren bør kunne måles. Til slutt bør den ha en fornuftig pris. Allegros A4950, en full bro PWM motor driver passer disse kravene bra. Den kan operere med spenning opp til 40V og kjøre motorer opp mot 3,5A. Hastigheten på motoren kan kontrolleres ved PWM, men den kan også selv begrense strømmen (såkalt "chopping") ved å sammenlikne en spenningsreferanse med en spenningen over en current sense motstand If formelen $I_{TripMAX} = \frac{V_{REF}}{10 \cdot R_s} \cdot V_{REF}$ er koblet til en DAC utgang av ATXmegaen slik at man kan lage og endre referanse slik man måtte ønske. Spenningen over current sense motstanden er også koblet til en ADC

inngang på Xmegaen om en fremtidig algoritme ønsker å bruke dette målet direkte. Figure 4.8 viser en oversikt over A4950s oppbygning.

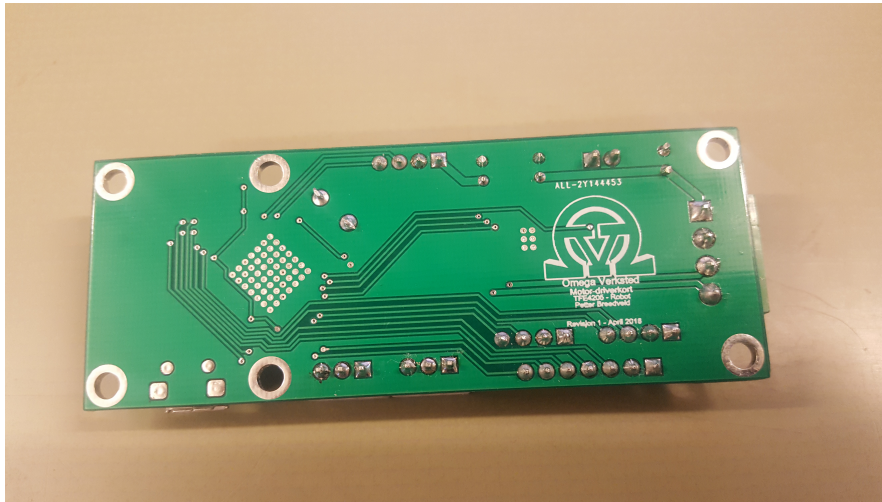
Current sense motstanden er valgt som $0,300\Omega$ slik at $V_{REF} = 3,3V$ fører til en maks strøm på $I_{TripMAX} = 1,1A$



Figur 4.8: Blokkdiagram over Allegro A4950 motor driver



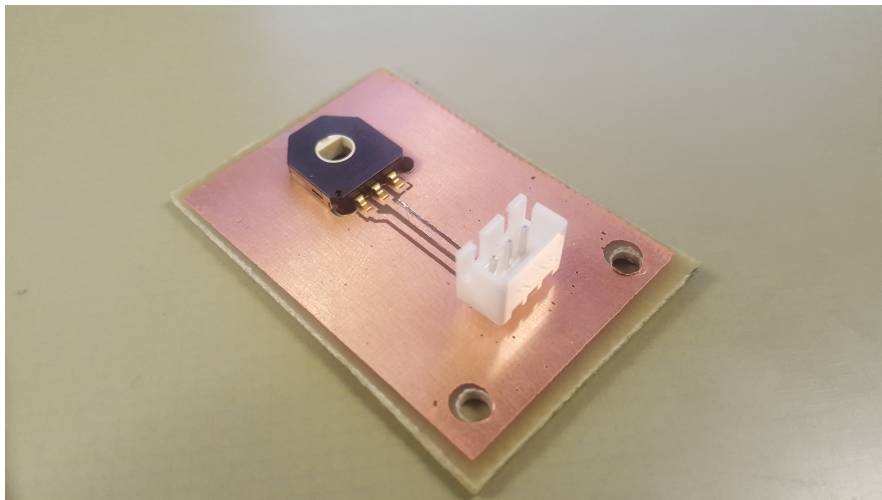
Figur 4.9: Motordriveren



Figur 4.10: Motordriveren

4.5 Potmeterkort

På hvert ledd i robotens bein trengs et potensiometer for å måle leddets vinkel. Derfor har dette kortet blitt laget. Muratas potmeter SV01A103AEA01 har blitt valgt fordi det er flatt og lett å feste i en anvendelse som denne. Kortet har til nå kun blitt etset på OV, men er tenkt å bestilles fra samme leverandør som resten av kortene i fremtiden.



Figur 4.11: Potmeterkortet, etset på OV

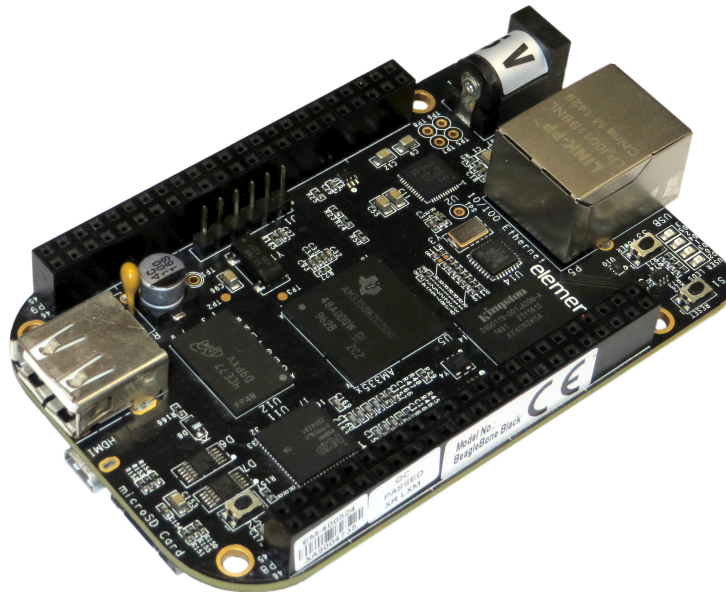
4.6 Sentralenhet

Oppgaven til sentralenheten er å kjøre de overordnede kontrollalgoritmene til systemet. Det er forventet at disse blir mer avanserte enn hva som kan prosesseres på en mikrokontroller. Det er flere alternativer som kan tilfredstille kravene blant annet, Beagle Bone Black og Raspberry Pi. På mange områder er disse to platformene ekvivalente, men Beagle Bone Black er bedre når det kommer til sammenkobling med ekstern hardware. Siden sammenkobling med eksterne enheter er sentralt for prosjektet er det valgt å bruke en Beagle Bone Black.

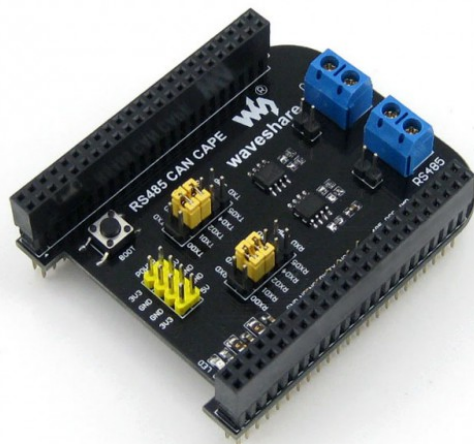
Beagle Bone Black kan programmeres i alle språk som støtter linux.

Beaglebone hardwaren har støtte for både vanlig seriell og rs485, men det trengs en ekstern driver for å få de rette elektriske nivåene. Dette er tilfredsstilt ved hjelp av et Waveshare RS485 CAN plug-in kort som er satt på toppen av Beagleboneen. Dette pluginkortet har støtte for både RS485 og CAN.

Det ble gjennomført en test av hvor raskt Beaglebone kan motta, behandle og sende meldinger. På baudrate 921600 ble det målt en pakke-til-pakke-tid på $14,2\mu\text{s}$ på UART-en, som tilsvarer over 7000 pakker per sekund. Det ble konkludert med at dette er raskt nok til våre formål.



Figur 4.12: Beagle Bone Black



Figur 4.13: Waveshare RS485 CAN Cape

4.7 RS485 Bus

Alle enhetene må kommunisere sammen, for å redusere mengden ledninger var nødvendig var det ønskelig å bruke en felles buss. Valget av standard var i hovedsak mellom CAN og RS485. Til dette systemet ble RS485 valgt på grunn av økt fleksibilitet og enkel tilkobling. Bakdelen er at protokollen må defineres og implementeres.

Alle enhetene er koblet sammen på en RS485-buss. Den består av ett differensialpar, to ledinger. Kun én enhet kan sende samtidig. For å unngå kollisjoner på bussen er sentralenheten satt opp som master. Det vil si at de andre enhetene kun sender når sentralenheten har spurt om det.

Kapittel 5

Programvare

Det er blitt skrevet flere drivere til modulene i prosjektet. Her er en oversikt over dem.

5.1 IMU

For IMU-modulen har det blitt implementert funksjonalitet for å hente ut måledata over SPI og sende det ut på RS485-bussen. MPU-9250-driveren er for det meste basert på arduinobiblioteket til Brian Chen [1], men tilpasset for å kjøre på en ATXMega.

5.2 Lastcelle

Programvaren til lastcellen er relativt enkel. Avlesingen fra hx711 chippen er basert på et bibliotek av Siddharth Talesra[19]. Dette er modifisert til å bruke de rette pinnene for designet.

5.3 Motordriver

På motordriver modulen er det mange muligheter til reguleringsalgoritmer som kan implementeres. Dette er ikke implementert på dette stadiet, da det ikke er fokuset av prosjektet og vil det falle under videre arbeid. RS485 driveren er lik den som kjører på de andre modulene, men konfigurert for motor-

relaterte kommandoer. Driveren for H-broen er på et enkelt men fungerende stadie, der det er lagt opp til at mer komplisert funksjonalitet slik som DAC kontrollert strøm begrensnig lett kan hukes inn. En ADC driver kan brukes til å lese posisjonen til potensiometer sensorer samt den spenningen over current sense motstanden.

5.4 Kommunikasjonsprotokoll

Subsystemene må enkelt kunne kommunisere med hverandre over en delt buss. For å gjøre dette enkelt, fleksibelt og utvidbart er en standardisert protokoll implementert. Protokollen inneholder en adresse og en kommando med nødvendige parametre til kommandoen. Hver melding starter med ! og avsluttes med 0x0a(newline) eller 0x0d(carriage return). Alle tegnene i meldingen, utenom den siste, består av skrivbare tegn i ascii settet. Dette gjør det enkelt å feilsøke ved å skrive meldinger inn i en seriellterminal. Dette har litt mer overhead enn å sende binære kommandoer, men det gjør den vanskeligere å koble seg opp mot. Som en følge av at noen av kravene setter enkelhet til bruk høyt, er det derfor valgt å bruke lesbare meldinger.

Alle tall i parametrene er skrevet i heksadesimal med små bokstaver. Alle kommandoer og parameternavn er skrevet med store bokstaver. Tallene kan være enten 16 eller 8 bit. Flyttall er ikke støttet. Kommandoer som slutter på S skriver data til en enhet, de som slutter på G spør en annen enhet om data.

Hver enhet kan ha flere adresser. Adresser kan også deles mellom enheter, som foreksempel kan brukes til å sende den samme meldingen til flere enheter samtidig.

```
!a8MCSP3a1c          //Setter posisjonen til en motor controller.  
!08IGA              //Spør om IMU akselerasjon.  
!00IASX4321Y2343Z346 //Sender xyz akselerasjon ut.
```

For å lett kunne implementere dette i koden, er det skrevet et bibliotek som gjør det enkelt å legge til nye kommandoer. Biblioteket er implementert slik at det vil kjøre en brukerdefinert funksjon når kommandoen blir tatt imot. Brukeren av bibiloteket trenger bare konfigurere funksjonen som skal brukes til hver kommando.

Driveren for kommunikasjons protokollen bruker to andre moduler, rs485 og et sirkulært buffer. Protokoll modulen og rs485 modulen er laget til prosjektet. Sirkulærbufferet er basert på et eksempel fra

embeddedartistry.com[4]. Dette er modifisert for å fungere i lag med de andre modulene.

5.5 RS485 drivere til Beaglebone

Det finnes flere distribusjoner av linux til Beaglebone, blant disse er Debian og Angstrom. Debian er en svært utbredt distribusjon som er mye brukt til både embedded og server linux. Angstrom er en distribusjon laget for å være liten, og passer godt til innevevdesystemer. Bakdelen for dette prosjektet med den er at det mangler en del vanlige pakker som gjør den enklere å bruke. Derfor ble Debian valgt.

Beagle Bone Black har støtte for RS485, men den nyeste versjonen av Debian for beaglebone er ikke disse driverene inkludert. For å få de inn på systemet måtte kjernen recompileres. Siden Beagle Bone Black bruker en ARM arkitektur krever å krysskompilere fra en PC. Det er også mulig å compilere på Beagle Bone Black, men dette tar mye lengre tid en å compilere på PC.

Kapittel 6

Videre arbeid

I dette kapittelet blir en kort oppsummering av videre arbeid som burde gjøres, både essensielle deler og mindre essensielle deler.

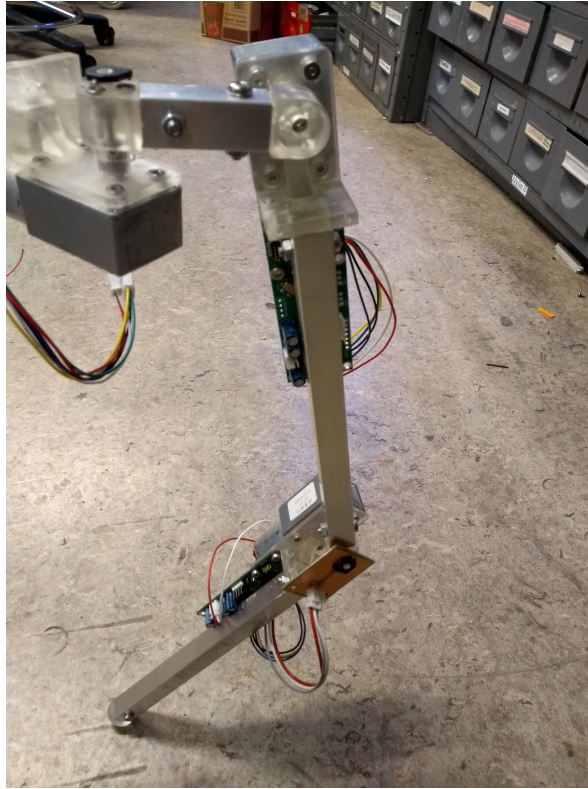
6.1 Generelt

Etter som alle motorene ¹ ikke kom i tide, er det først og fremst nødvendig å montere roboten ferdig med de delene som er ferdig designet. Roboten som den er pr juni 2018 kan sees i figur 6.1

6.2 Trådløs styring og batteri

For å kunne brukes og testes trengs en mulighet for operatøren av roboten å styre den. I prinsippet ser arbeidsgruppen for seg en trådløs kommunikasjonsform som BLE, WIFI eller lignende. Etter hvert er det også ønskelig at roboten har eget batteri montert på platformen.

¹pr. juni 2018



Figur 6.1: Delvis sammensatt ben av roboten

6.3 Aktiv fot

Selv om vektmålere i beinfestet kan gi gode indikasjoner på robotens vektfordeling, kan det være nyttig å innføre føtter med streklapper innebygd for å detektere vinkel på foten, eventuelt hvis det er ønskelig å kunne bevege seg på ikke-rette, ikke-homogene overflater. Dette er likevel ikke et teknisk krav som forfatterne vil prioritere høyt ved kontinuasjon av prosjektet.

6.4 Mekanisk

Videre arbeid på det mekaniske er hovedsakelig i form av å identifisere svake punkter i de mekaniske delene, og utbedre disse. Elementer som er planlagt å legge til er en festeanordning for optisk enkoder på motorfestene. Dette krever not utprøving av den optiske sensoren før det kan implementeres. Andre mekaniske utbedringer er forbundet med å feste eventuelle løse ledninger som kan by på problemer.

I tillegg må det før testing av gangarter lages en testjigg slik at roboten kan falle uten å ta skade. Når prosjektet har kommet langt kan det være ønskelig å designe de mekaniske komponentene etter profesjonelle fremgangsmåter, for eksempel topologisk autogenerering basert på kraftanalyse eller elementmetoden.

6.5 Elektronikk

Når de første fulle kortene var loddet opp ble det oppdaget at footprintet til RS485-brikken som var brukt på alle kortene var feil. Dette ble fikset midlertidig ved å lodde om, men det er tenkt å bestille nye kretskort hvor slike småfeil er rettet opp i.

Det ble også oppdaget at ADCen til ATXMEGA64A3U ikke kan lese hele rangen 0-3V3, og istened stopper på ca. 2,06V. For å kompensere for dette bør det festes en motstand i serie med potmeteret på 3V3 siden slik at sveiper ikke kommer mye over denne verdien.

Ellers gjenstår det å lodde opp resten av kortene som trengs til roboten, siden det så langt ikke har blitt loddet opp særlig mange flere enn det som trengs for å teste funksjonalitet.

Det skal også lages et senterkort for å minimere ledningsrot rundt sentralenheten.

6.6 Programmering

Enkelte av mulighetene som finnes på kretskortene, for eksempel USB-kontakt, er ikke implementert og trenger ikke å implementeres i forbindelse med dette prosjektet. En adaptiv regulator som nevnt i kapittel 2 burde implementeres, testes og justeres ved videre arbeid. På sentralenheten gjenstår fortsatt mye overordnet kode i forbindelse med den matematiske modellen og kontrollallokering. På IMU-driveren gjenstår det å lage funksjonalitet for å lese kompasset. Dette er mer innviklet enn å lese akselerometer og gyroskop, da disse kan leses direkte over SPI mens man for å lese kompasset må sende I2C-kommandoer over SPI til IMU-en.

6.7 Matematisk modell

Det er store deler av den matematiske modellen som det gjenstår å få ferdig, som forfatterne mener er uvesentlig å nevne i denne rapporten. Leseren bes derfor undersøke denne delen i [13]. Noe av arbeidet med den matematiske modellen som er spesifikt for roboten er tett knyttet til programmering av sentralenheten. I tillegg er det en omfattende identifiseringsjobb som må forekomme for å verifisere faktorer ved modellen som ikke er lett målbare. Dette er spesifisert i kapittel 2.

Tillegg A

Akronymer

OV Omega Verksted

IMU Inertial Measurement Unit

ITK Institutt for Teknisk Kybernetikk

IES Institutt for Elektronisk Systemdesign

I2C Inter-Integrated Circuit (seriell databuss)

PWM Pulse width modulation, pulsbreddemodulering

DC Direct current, Likestrøm

AC Alternating current, Vekselstrøm

RPM Revolutions per minute, omdreininger per minutt

RT-OS Real time operating system, sanntidsoperativsystem.

IC Integrated circuit, integrert krets

QFN Quadratic Flat Non-lead package, en pakke type for integrerte kretser.

QFP Quadratic Flat Package, pakke type for integrerte kretser.

SPI Serial-Periferal Interface, seriell databus

PRU Programmable Real-time Unit

DOF Degrees of freedom, frihetsgrader

DH Denavit-Hartenberg-konvensjon, en måte å finne generaliserte koordinater for roboter

HZD Hybrid Zero Dynamics, en kontrollstrategi for syklisk stabil gange

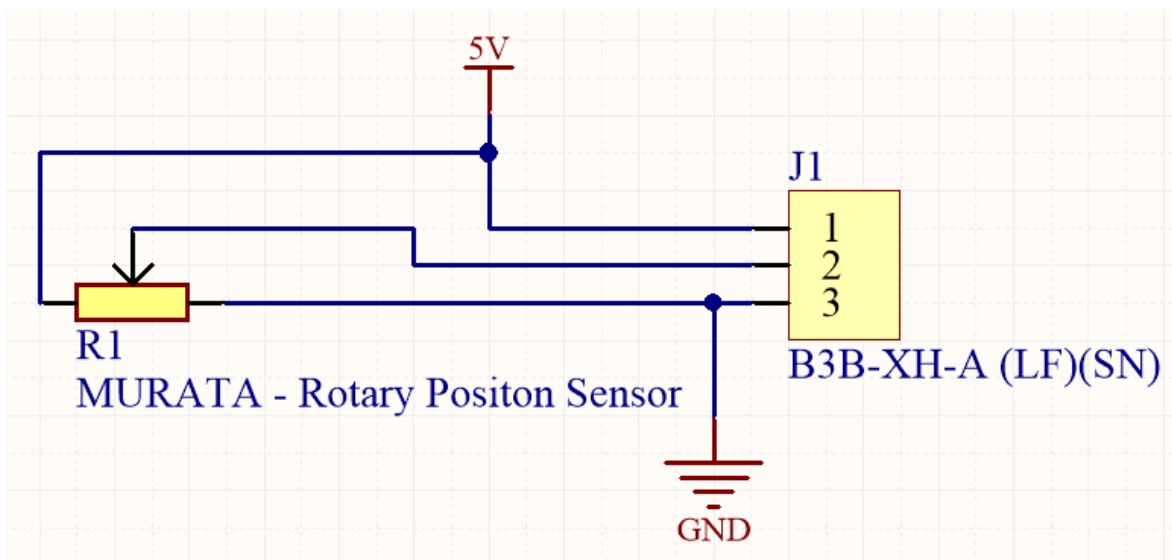
CP Configuration Polygon, det konvekse polygonet der hjørnene er enden på bena som er i kontakt med bakken

SS State Stability, konservativ kontrollform for roboter

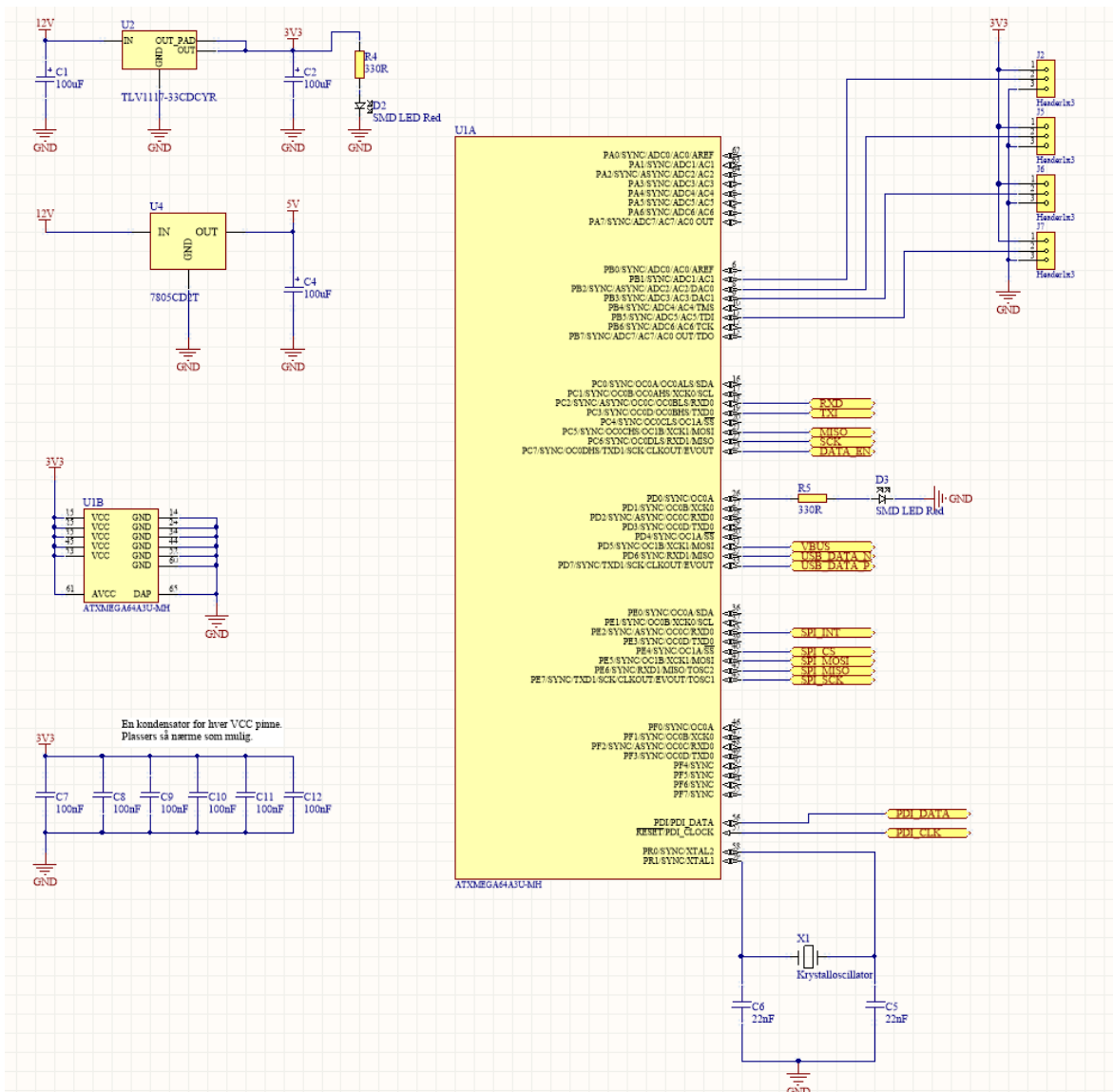
Tillegg A

Vedlegg

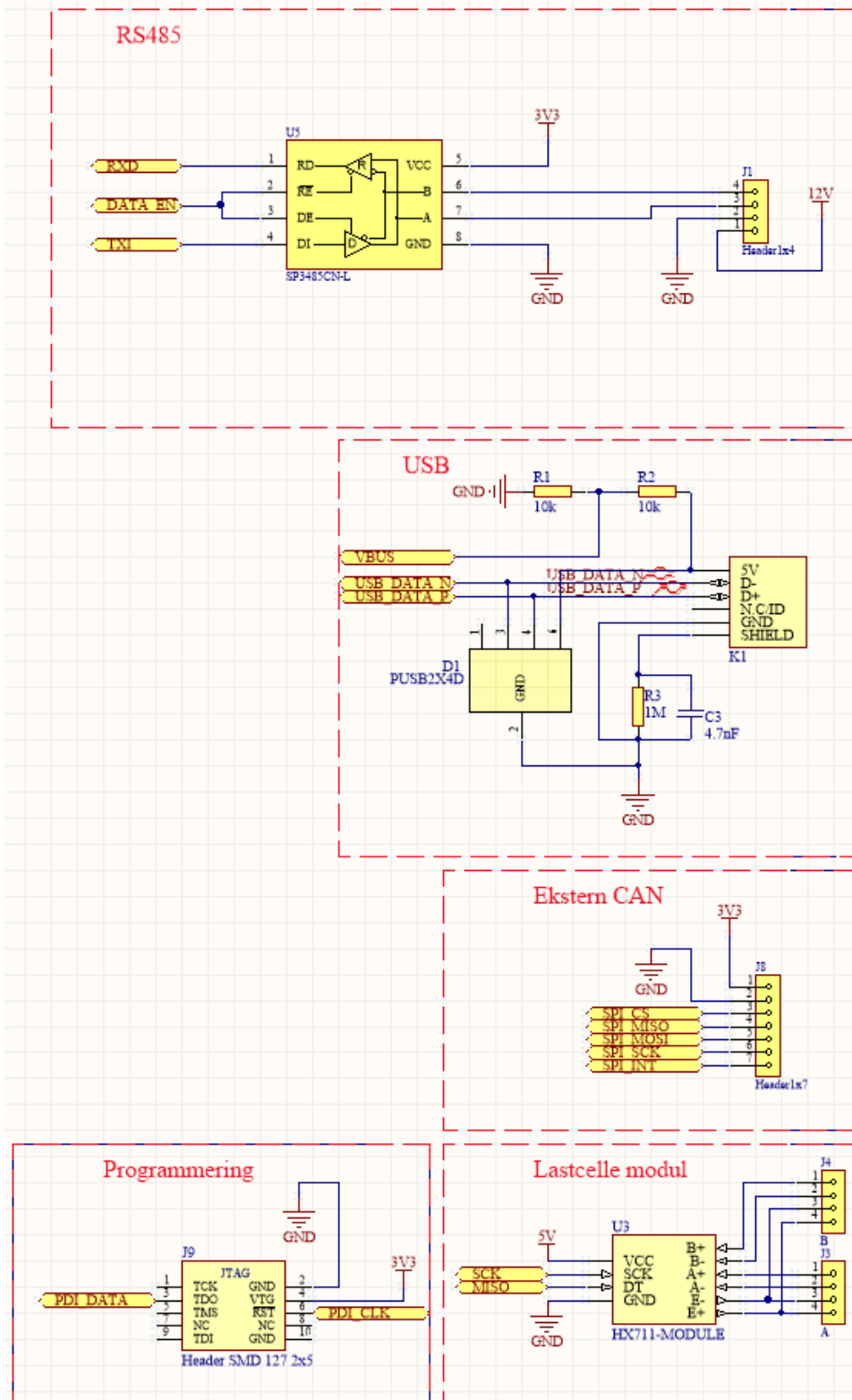
A.1 Kretstegninger



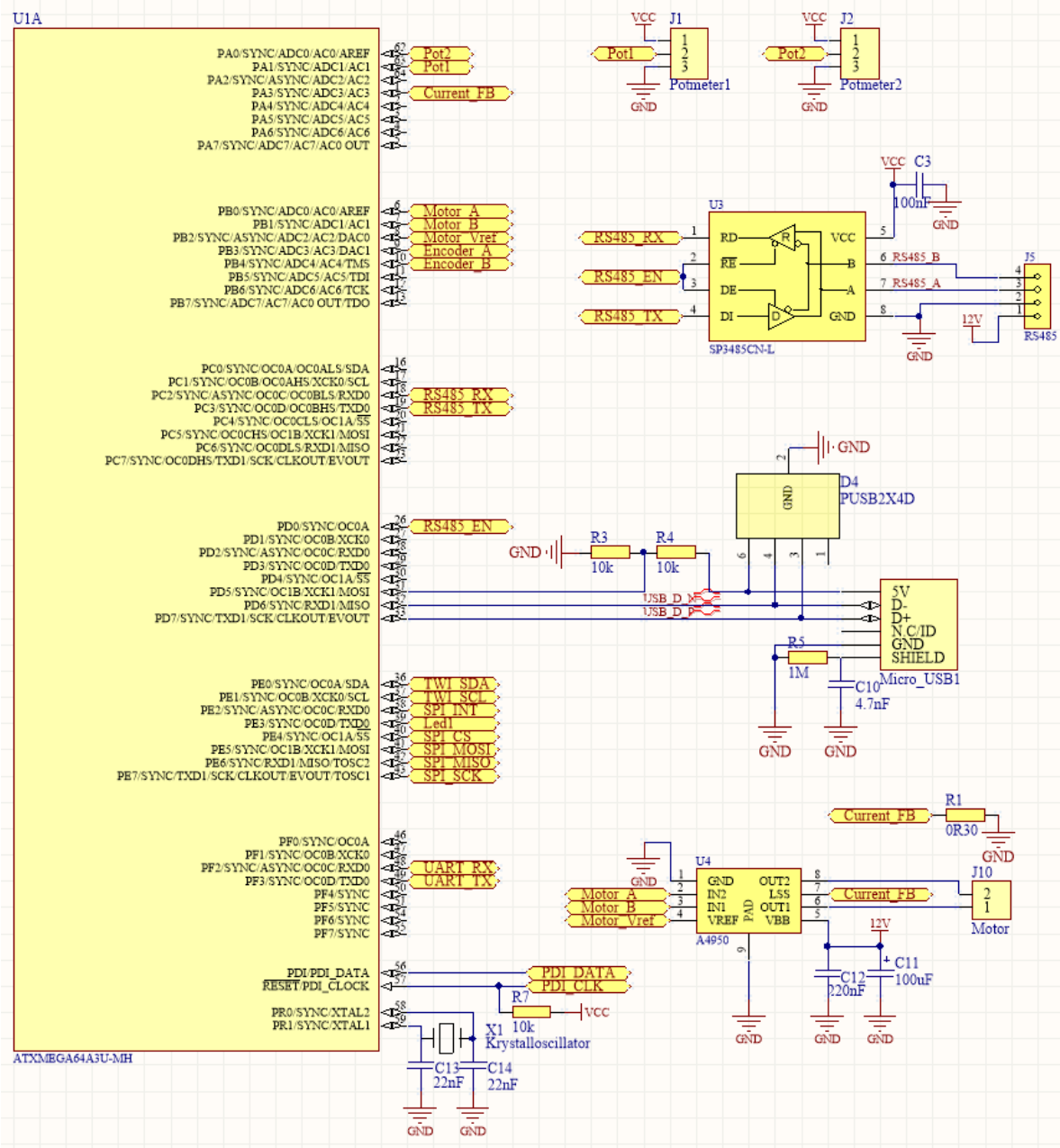
Figur A.1: Skjematikk av potmeterkort



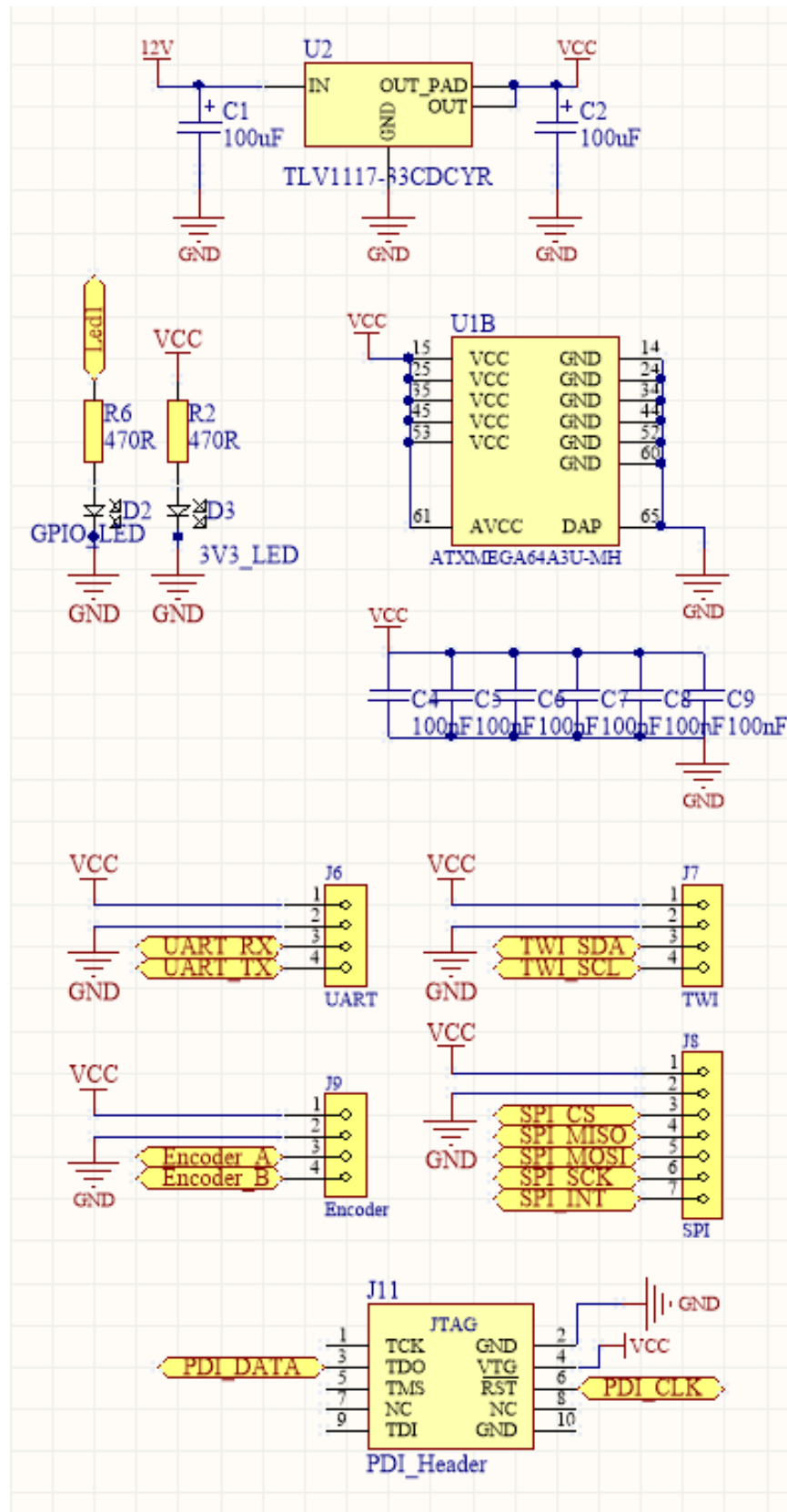
Figur A.2: Skjematikk av lastcelle-modul



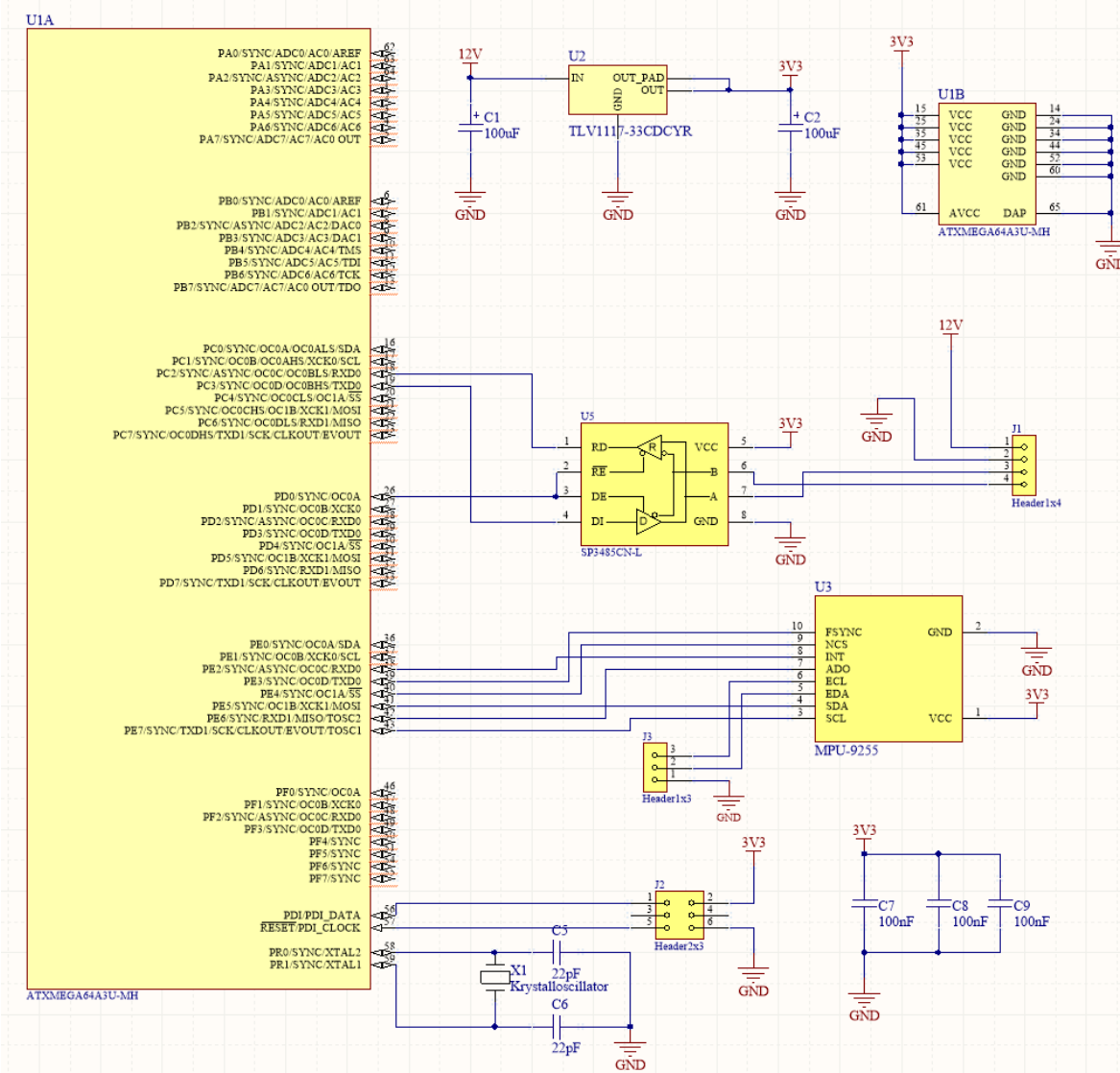
Figur A.3: Skjematikk av lastcelle-modul



Figur A.4: Skjematikk av motordriver

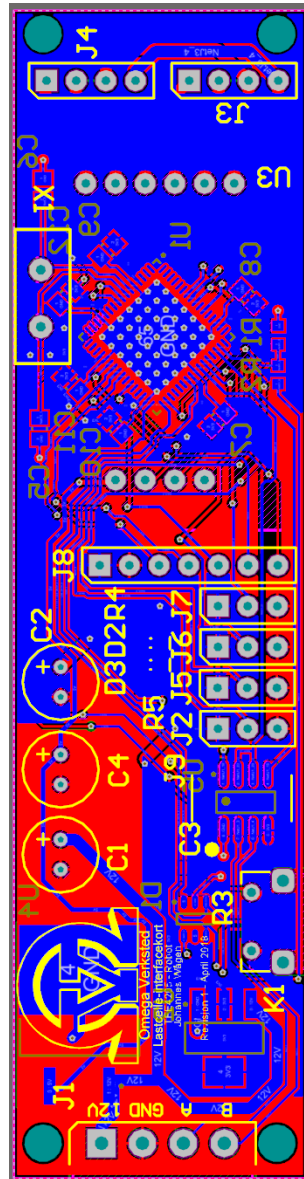


Figur A.5: Skjematikk av motordriver

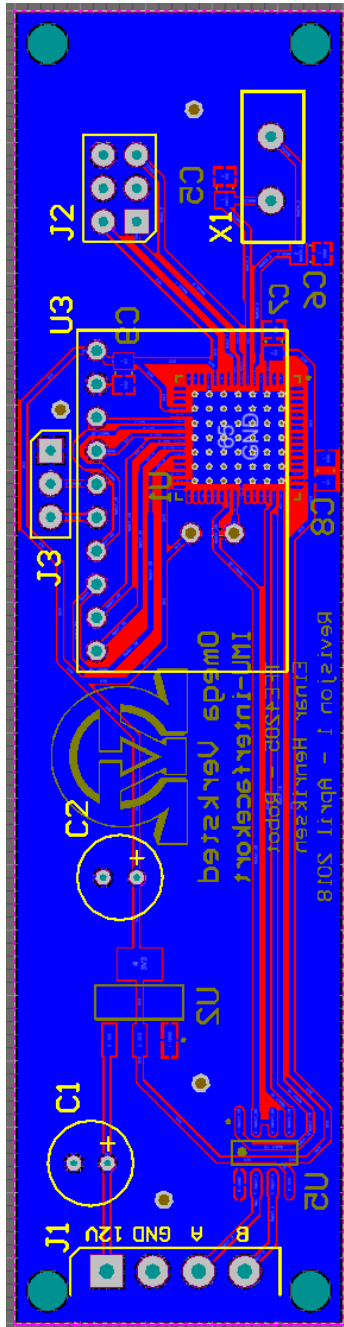


Figur A.6: Skjematikk av IMU-modul

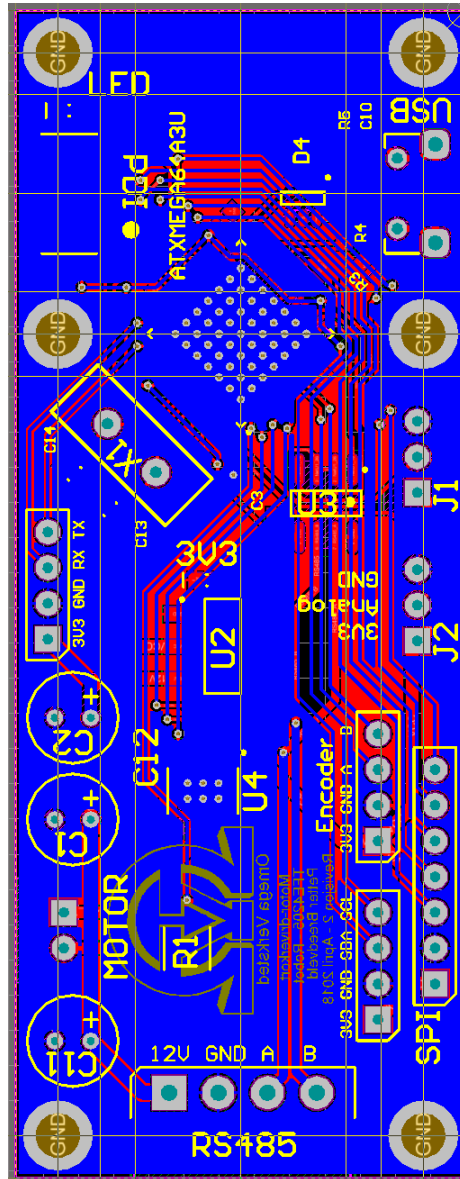
A.2 Utlegg



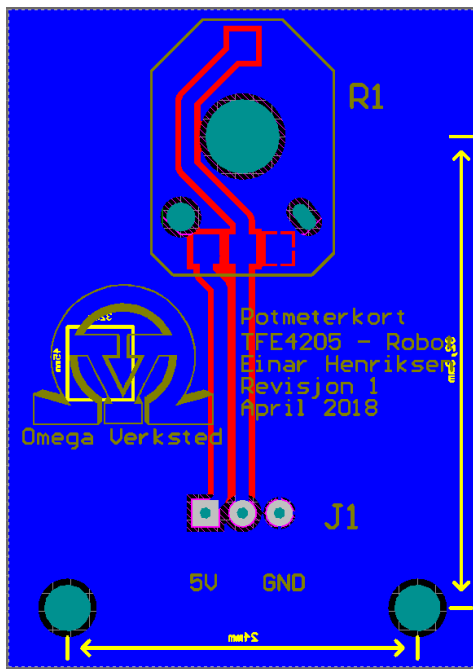
Figur A.7: Utlegg av lastcelle modul



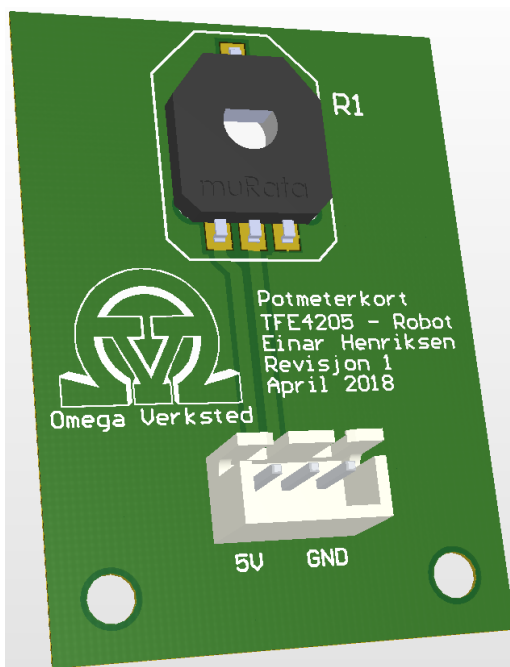
Figur A.8: Utlegg av IMU-modul



Figur A.9: Utlegg av motordriver



Figur A.10: Utlegg av potmeterkort



Figur A.11: Utlegg av potmeterkort

A.3 Bill of materials

Tabell A.1: BOM: motordriver

Comment	Description	Designator	Quantity
3V3_LED		3V3	1
Encoder	Header 1x4	3V3 GND A B	1
UART	Header 1x4	3V3 GND RX TX	1
TWI	Header 1x4	3V3 GND SDA SCL	1
ATXMEGA64A3U-MH		ATXMEGA64A3U	1
100uF	25V 2.5mm pitch	C1, C2, C11	3
4.7nF	NP0 50V 5%	C10	1
220nF	Capacitor	C12	1
22pF	X7R 50V	C13, C14	2
PUSB2X4D	USB protection diodes	D4	1
Potmeter1	Header 1x3	J1	1
Potmeter2	Header 1x3	J2	1
GPIO_LED		LED	1
Motor	Header 1x2	MOTOR	1
PDI_Header	Header 2x5	PDI	1
0R30	2W 1% Thick Film	R1	1
1M	0.1W 1% Thick film resistor	R5	1
RS485		RS485	1
SPI	Header	SPI	1
TLV1117-33CDCYR	3V3 Regulator	U2	1
SP3485CN-L	RS485 Tranciever	U3	1
A4950		U4	1
USB_CONNECTOR_5P	USB mini/micro	USB	1
Krystalloscillator	16 MHz, generisk	X1	1
470R	0.1W 1% Thick film resistor	Designator1	2
10k	0.1W 1% Thick film resistor	R3, R4	3
100nF	X7R 50V 5%	C3	7

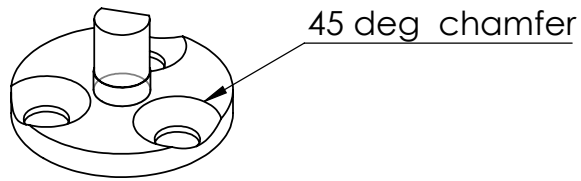
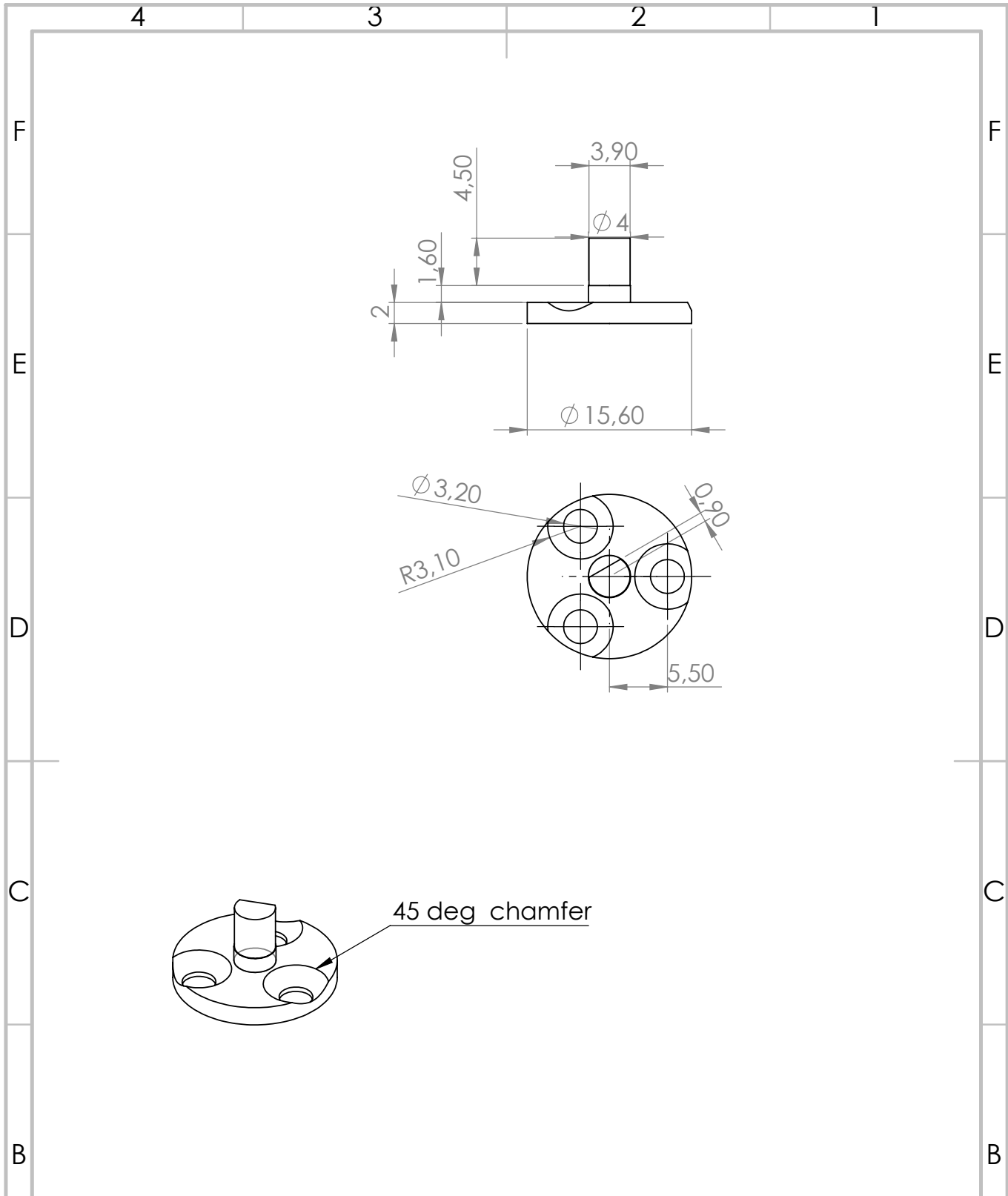
Tabell A.2: BOM: imu-modul

Comment	Description	Designator	Quantity
100uF	25V 2.5mm pitch	C1, C2	2
22pF	50v,	C5, C6	2
100nF	X7R 50V	C7, C8, C9	3
Header1x4		J1	1
Header2x3		J2	1
Header1x3		J3	1
ATXMEGA64A3U-MH		U1	1
TLV1117-33CDCYR	3V3 Regulator	U2	1
MPU-9255		U3	1
SP3485CN-L	RS485 Tranciever	U5	1
Krystalloscillator	16 MHz, generisk	X1	1

Tabell A.3: BOM: lastcelle-modul

Comment	Description	Designator	Quantity
100uF	25V 2.5mm pitch	C1, C2, C4	3
4.7nF	NP0 50V 5%	C3	1
22pF	X7R 50V	C5, C6	2
100nF	X7R 50V	C7, C8, C9, C10, C11, C12	6
PUSB2X4D	USB protection diodes	D1	1
SMD LED Red		D2, D3	2
Header1x4		J1, J3, J4	3
Header1x3		J2, J5, J6, J7	4
Header1x7		J8	1
Header SMD 127 2x5		J9	1
USB_CONNECTOR_5P	USB mini/micro	K1	1
10k	0.1W 1% Thick film resistor	R1, R2	2
1M	0.1W 1% Thick film resistor	R3	1
330R	0.1W 1% Thick film resistor	R4, R5	2
ATXMEGA64A3U-MH		U1	1
TLV1117-33CDCYR	3V3 Regulator	U2	1
HX711-MODULE	Load cell ADC module	U3	1
7805CD2T	5V Regulator	U4	1
SP3485CN-L	RS485 Tranciever	U5	1
Krystalloscillator	16 MHz, generisk	X1	1

A.4 Tekniske tegninger



UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MILLIMETERS
SURFACE FINISH:
TOLERANCES: NS-ISO 2768-1 M
LINEAR:
ANGULAR:

FINISH:
Rinsing in isopropanol
Hardening in UV-light

DEBURR AND
BREAK SHARP
EDGES

DO NOT SCALE DRAWING

REVISION 2

OMEGA VERKSTED

axel extension

TITLE:

DWG NO.

axel_extension

A4

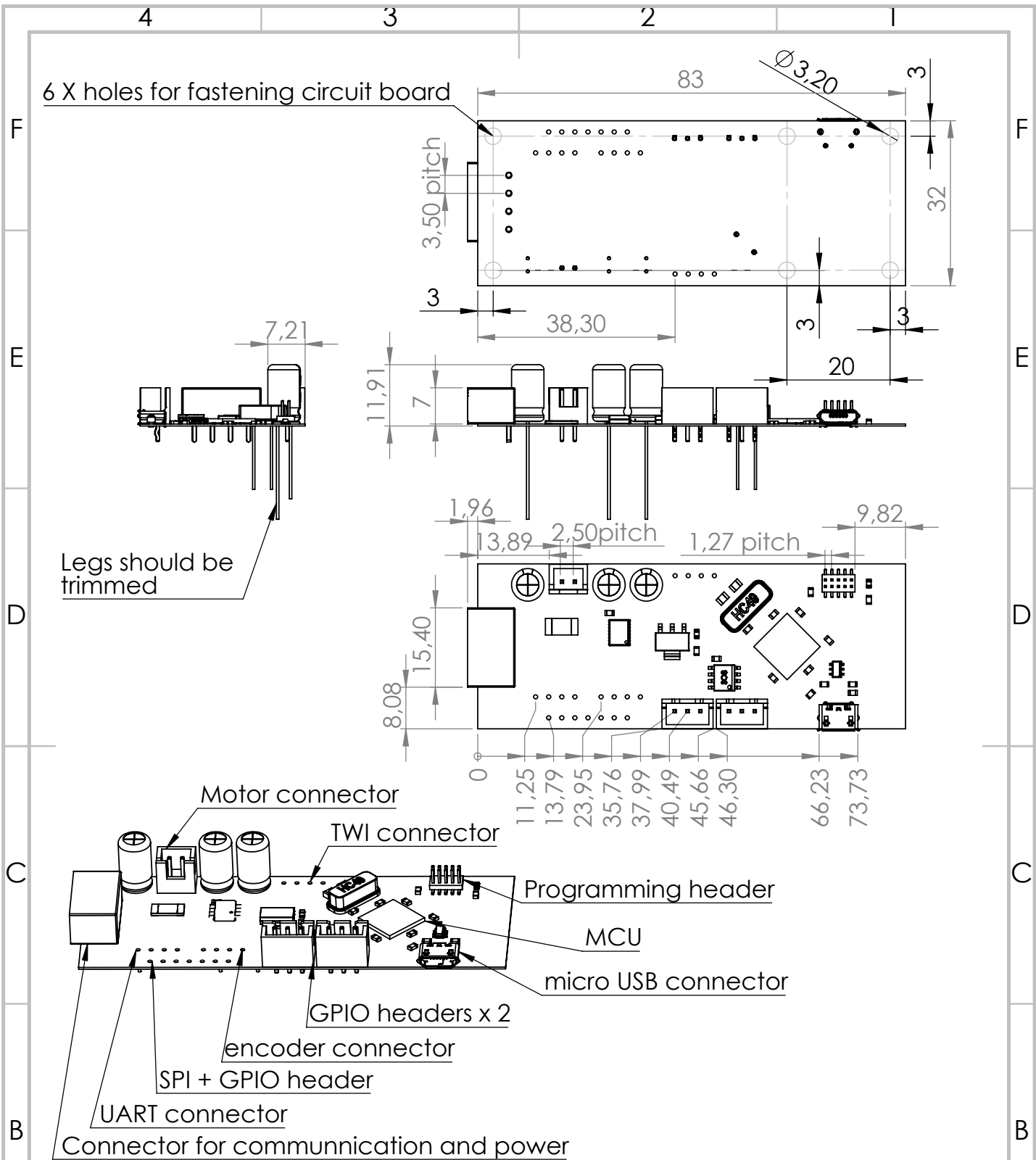
MATERIAL:

3D-printed photo polymer
FLGPCL03

WEIGHT:

SCALE:2:1

SHEET 1 OF 1



UNLESS OTHERWISE SPECIFIED:
 DIMENSIONS ARE IN MILLIMETERS
 SURFACE FINISH:
 TOLERANCES:
 LINEAR:
 ANGULAR:

PRODUCTION METHOD:
 Soldering and assembly



DO NOT SCALE DRAWING

DEBURR AND BREAK SHARP EDGES

OMEGA VERKSTED

TITLE:
 motor controller
 circuit board

NAME	SIGNATURE	DATE
DESIGNER Petter Breedveld		
DRAWN Per Anton Ø. Olsen		03.05.18
CHK'D Johannes Wågen		
TESTED Petter Breedveld		

POST-PRODUCTION REMARKS
 remove flux from circuit
 board afeter soldering

MATERIAL(S):
 Different materials

DWG NO/FILE NAME.

Driver_controller

A4

OTHER COMMENTS:
 Keep conducting surfaces away from top and bottom of
 circuit board after assembly. For detailed information
 about circuit, see "Driver_controller.PrijPcb"

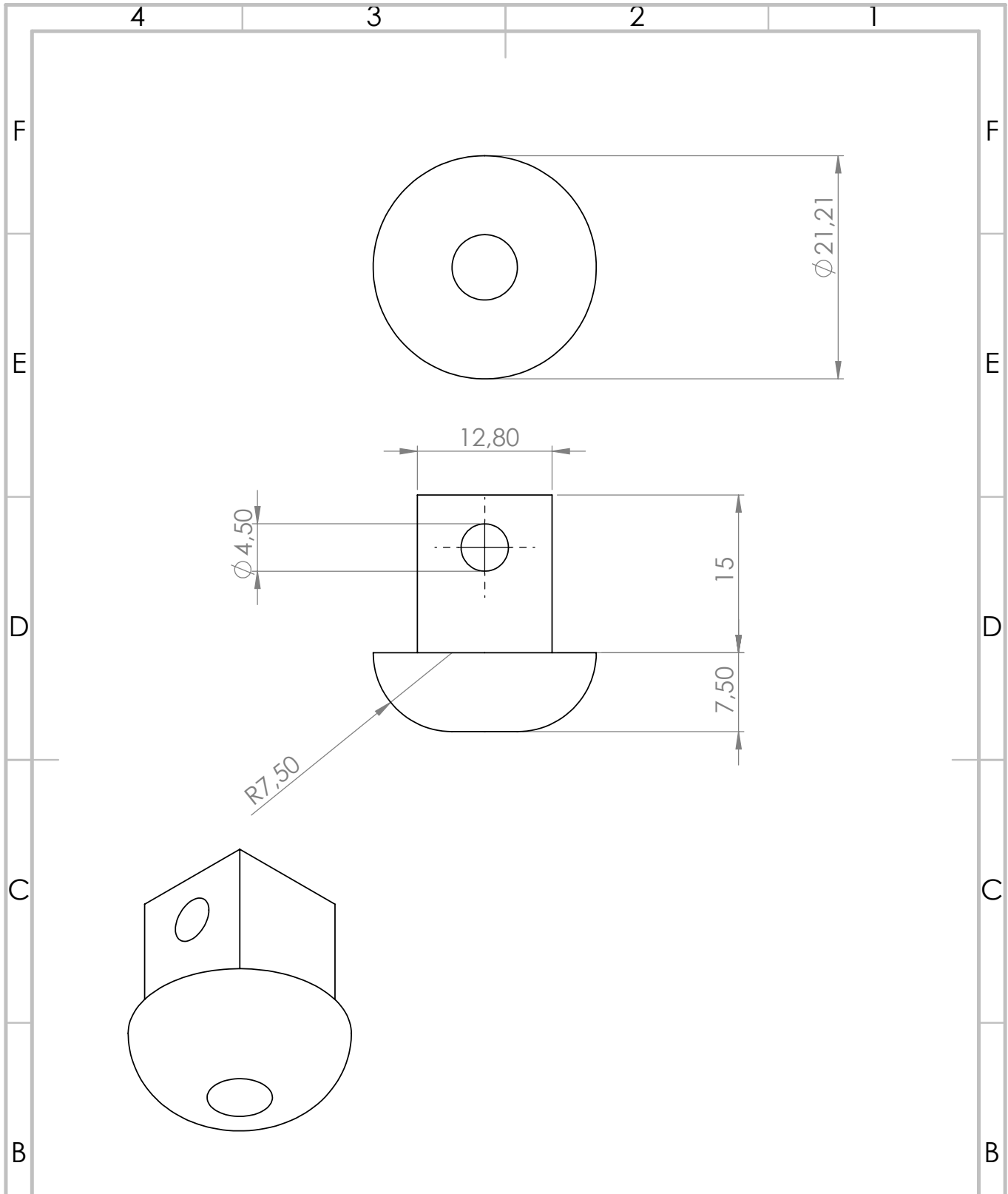
WEIGHT:

SCALE:1:1

SHEET 1 OF 1

REVISION

1



UNLESS OTHERWISE SPECIFIED:
 DIMENSIONS ARE IN MILLIMETERS
 SURFACE FINISH:
 TOLERANCES: NS-ISO 2768-1 M
 LINEAR:
 ANGULAR:

FINISH:
 Rinsing in isopropanol
 Hardening in UV-light

DEBURR AND
 BREAK SHARP
 EDGES

DO NOT SCALE DRAWING

REVISION 1

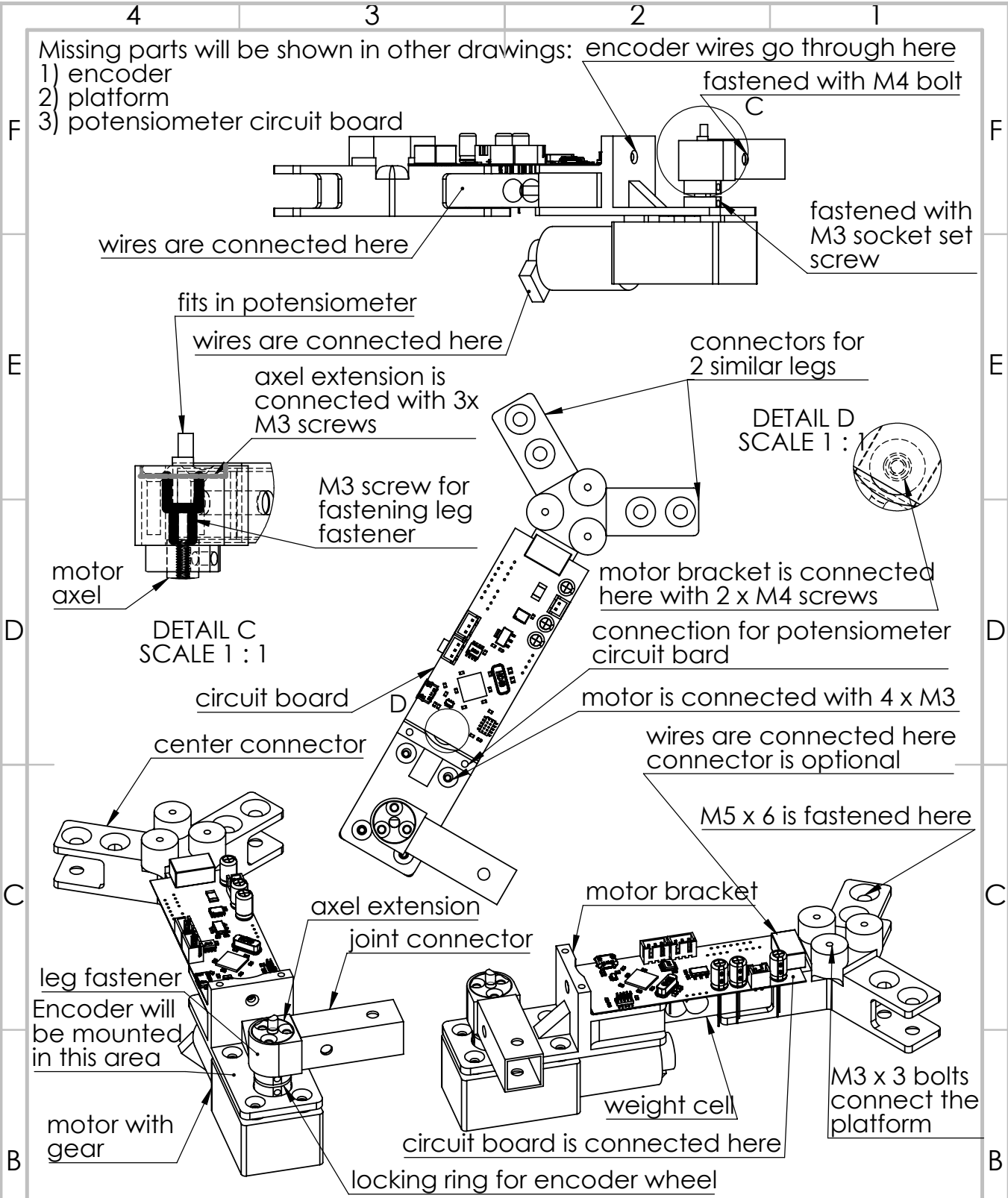
OMEGA VERKSTED

NAME	SIGNATURE	DATE		
DRAWN Per Anton Ø. Olsen		16.04.18		
CHK'D Johannes Wågen				
APPVD Johannes Wågen				
MFG				
Q.A				
MATERIAL:			3D-printed photo polymer FLGPCL03	
WEIGHT:				

TITLE:	Foot without sensors	
DWG NO.	foot	A4
SCALE:2:1	SHEET 1 OF 1	

Missing parts will be shown in other drawings: encoder wires go through here

- 1) encoder
- 2) platform
- 3) potentiometer circuit board



UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MILLIMETERS
SURFACE FINISH:
TOLERANCES:
LINEAR:
ANGULAR:

FINISH:

DEBURR AND
BREAK SHARP
EDGES

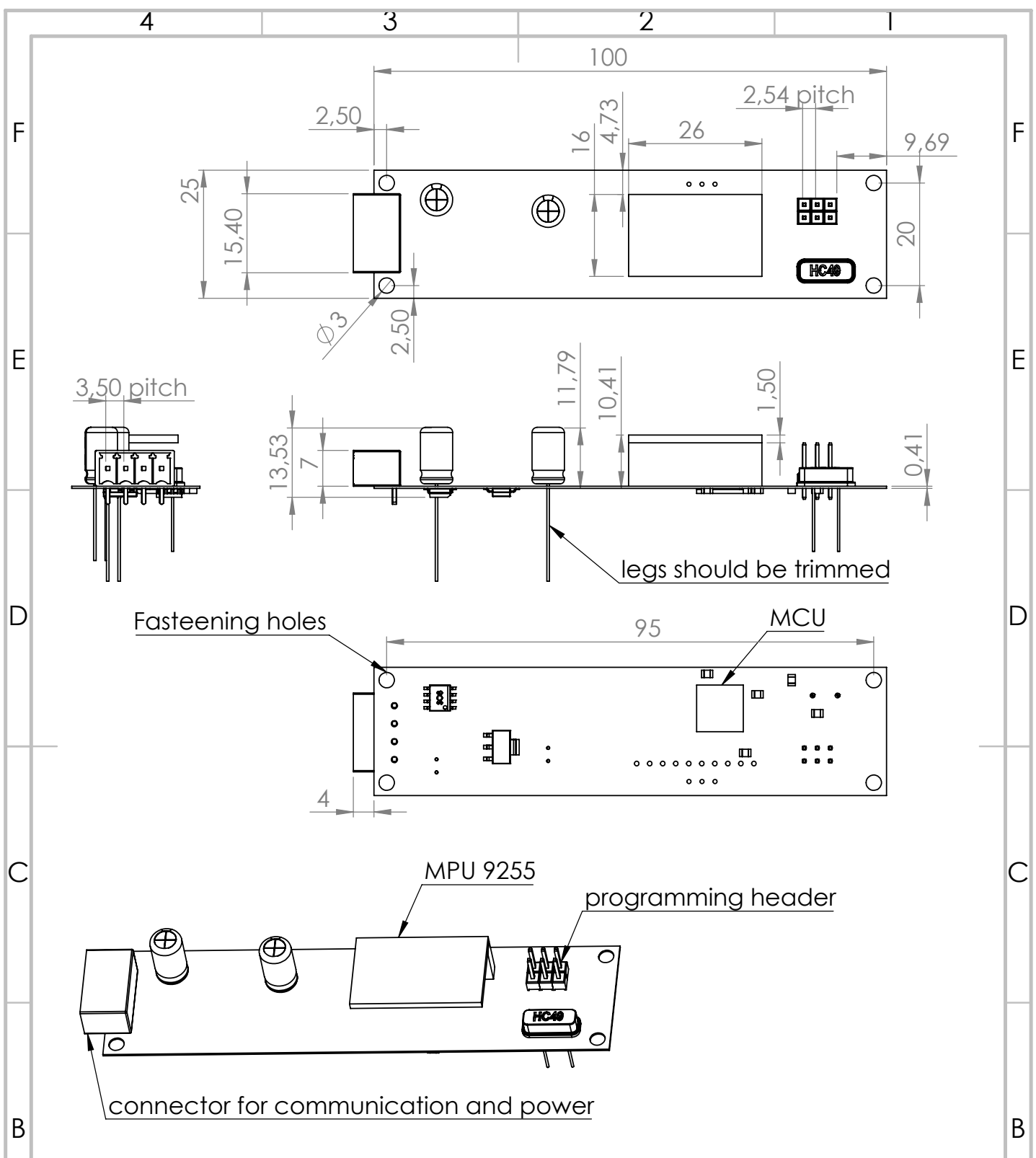
DO NOT SCALE DRAWING

REVISION 1

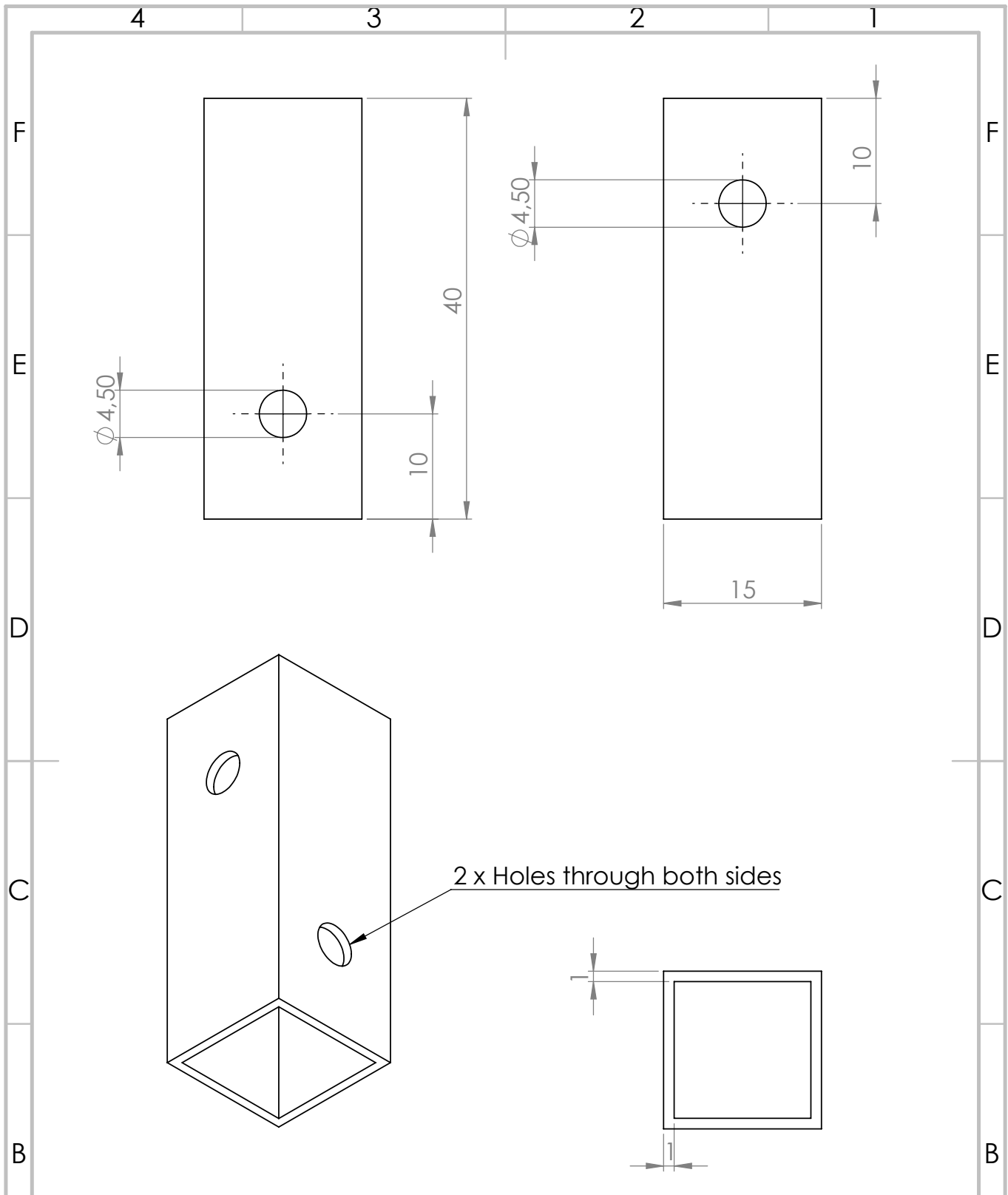
OMEGA VERKSTED

NAME	SIGNATURE	DATE
DRAWN Per Anton Ø. Olsen		30.04.18
CHK'D Johannes Wågen		
APP'VD Johannes Wågen		
MFG		
Q.A		
MATERIAL:		Different materials
WEIGHT:		

TITLE:	Assembly of center parts
DWG. NO.	
SCALE: 1:5	SHEET 1 OF 1



UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:		PRODUCTION METHOD: Soldering and assembly		DO NOT SCALE DRAWING		DEBURR AND BREAK SHARP EDGES	
						OMEGA VERKSTED	
NAME		SIGNATURE		POST-PRODUCTION REMARKS		TITLE:	
DESIGNER	Johannes Wågen			remove flux from circuit board after soldering		<h1>IMU circuit board</h1>	
DRAWN	Per Anton Ø. Olsen		03.05.18				
CHK'D	Johannes Wågen						
TESTED				MATERIAL(S):		DWG NO./FILE NAME.	
APP'VD				different materials		imu	
OTHER COMMENTS: Keep conducting surfaces away from top and bottom of circuit board after assembly. For detailed information about circuit, see 'imu.PriPcb'				WEIGHT:		SCALE:1:2	
						SHEET 1 OF 1	
						REVISION 1	



UNLESS OTHERWISE SPECIFIED:
 DIMENSIONS ARE IN MILLIMETERS
 SURFACE FINISH:
 TOLERANCES: NS-ISO 2768-1 M
 LINEAR:
 ANGULAR:

FINISH:
 no finish

DEBURR AND
 BREAK SHARP
 EDGES

DO NOT SCALE DRAWING

REVISION 1

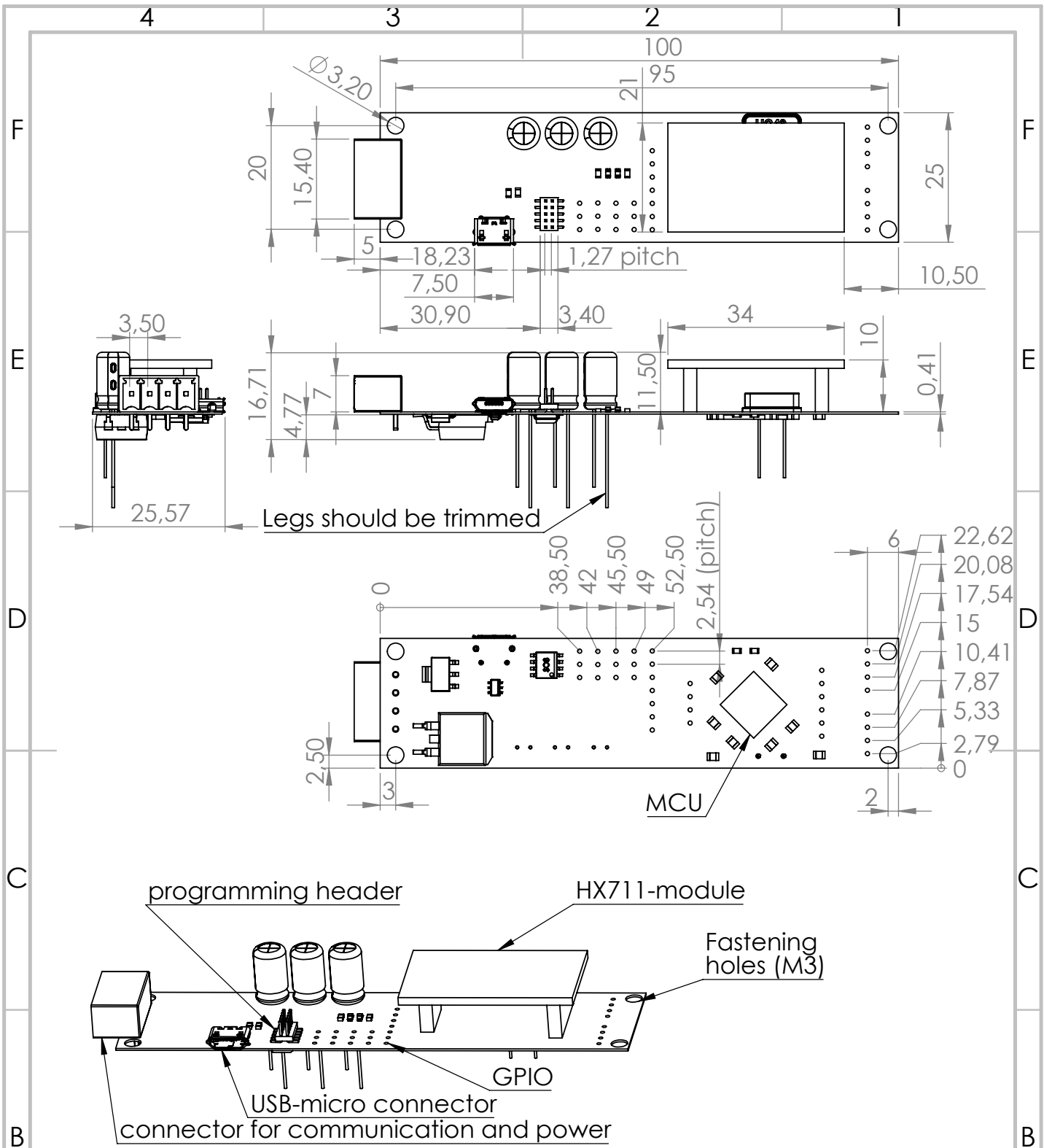
NAME	SIGNATURE	DATE
DRAWN Per Anton Ø. Olsen		16.04.18
CHK'D Johannes Wågen		
APP'VD Johannes Wågen		
MFG		
Q.A		
MATERIAL: Aluminium		
WEIGHT:		

TITLE:
Joint connector

DWG NO.
joint_connector

SCALE:2:1

SHEET 1 OF 1



UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MILLIMETERS
SURFACE FINISH:
TOLERANCES:
LINEAR:
ANGULAR:

PRODUCTION METHOD:
Soldering & assembly



DO NOT SCALE DRAWING

DEBURR AND BREAK SHARP EDGES

OMEGA VERKSTED

TITLE:

Load cell circuit board

DWG NO./FILE NAME:

last-celle

A4

NAME	SIGNATURE	DATE	POST-PRODUCTION REMARKS
DESIGNER: Johannes Wågen			remove flux from circuit after soldering
DRAWN: Per Anton Ø. Olsen		03.05.18	
CHK'D: Johannes Wågen			
TESTED:			

MATERIAL(S):
different materials

OTHER COMMENTS:
Keep conducting surfaces away from top and bottom of circuit board after assembly. For detailed information about circuit, see "robot last-celle.PrtPcb"

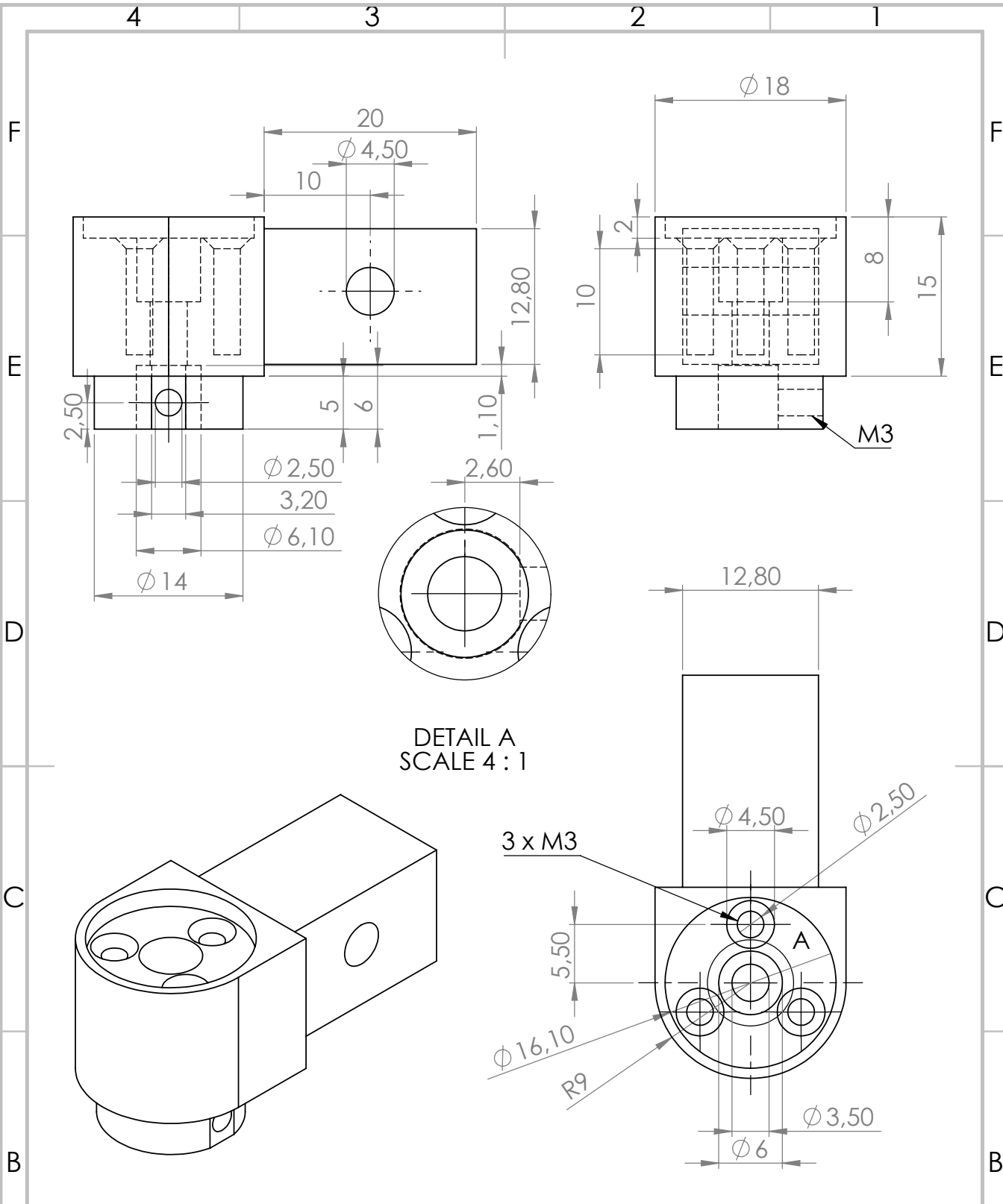
WEIGHT:

SCALE:1:1

SHEET 1 OF 1

REVISION

1



UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MILLIMETERS
SURFACE FINISH:
TOLERANCES: NS-ISO 2768-1 M
LINEAR:
ANGULAR:

FINISH:
Rinsing in isopropanol
Hardening in UV-lamp
Threading holes to M3

DEBURR AND
BREAK SHARP
EDGES

DO NOT SCALE DRAWING

REVISION 2

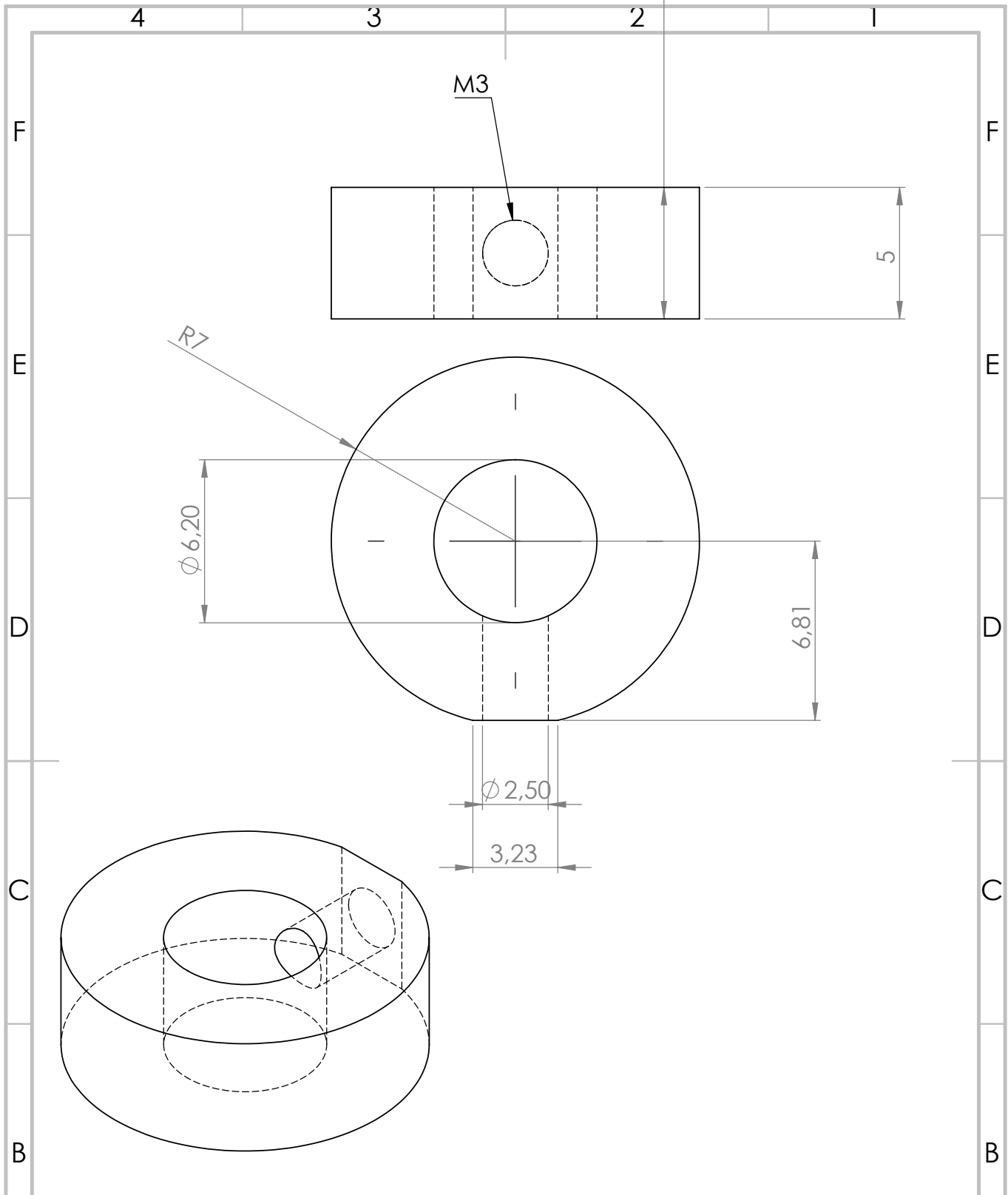
OMEGA VERKSTED

Leg fastner for
15mm x 15mm

Leg_fastner_2

NAME	SIGNATURE	DATE
DRAWN Per Anton Ø. Olsen		12.04.18
CHK'D Johannes Wågen		
APP'VD Johannes Wågen		
MFG		
Q.A		
MATERIAL:		
3D-printed photo polymer FLGPCL03		
WEIGHT:		

TITLE:	A4
DWG NO.	
SCALE:2:1	SHEET 1 OF 1



UNLESS OTHERWISE SPECIFIED:
 DIMENSIONS ARE IN MILLIMETERS
 SURFACE FINISH: -
 TOLERANCES: NS-ISO 2768-1 M
 LINEAR:
 ANGULAR:

FINISH:
 Rinsing in isopropanol
 Hardening in UV-light
 Threading hole to M3

DEBURR AND
 BREAK SHARP
 EDGES

DO NOT SCALE DRAWING

REVISION 2

OMEGA VERKSTED

locking ring

NAME	SIGNATURE	DATE
DRAWN Per Anton Ø.Olsen		06.4.18
CHK'D Johannes Wågen		
APPVD Johannes Wågen		
MFG		
Q.A		
MATERIAL:		
3d-printed photo polymer: FLGPCL03		
WEIGHT:		

TITLE:

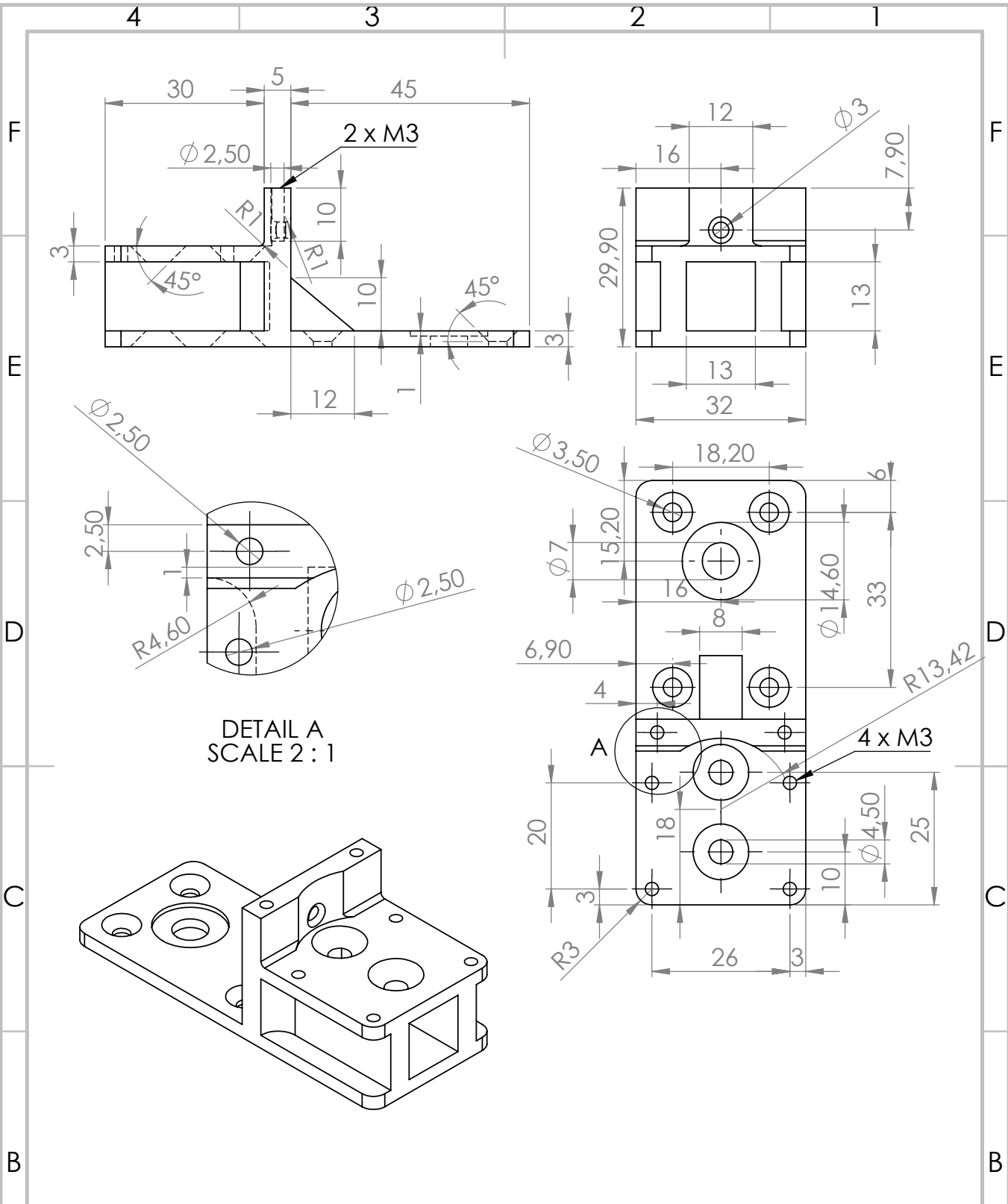
DWG NO.

locking_ring

A4

SCALE:5:1

SHEET 1 OF 1



UNLESS OTHERWISE SPECIFIED:
 DIMENSIONS ARE IN MILLIMETERS
 SURFACE FINISH:
 TOLERANCES: NS-ISO 2768-1 M
 LINEAR:
 ANGULAR:

FINISH:
 Rinsing in isopropanol
 Hardening in UV-light
 Threading holes to M3

DEBURR AND
 BREAK SHARP
 EDGES

DO NOT SCALE DRAWING

REVISION 6

OMEGA VERKSTED

TITLE:
 Motor bracket for
 shoulder joint

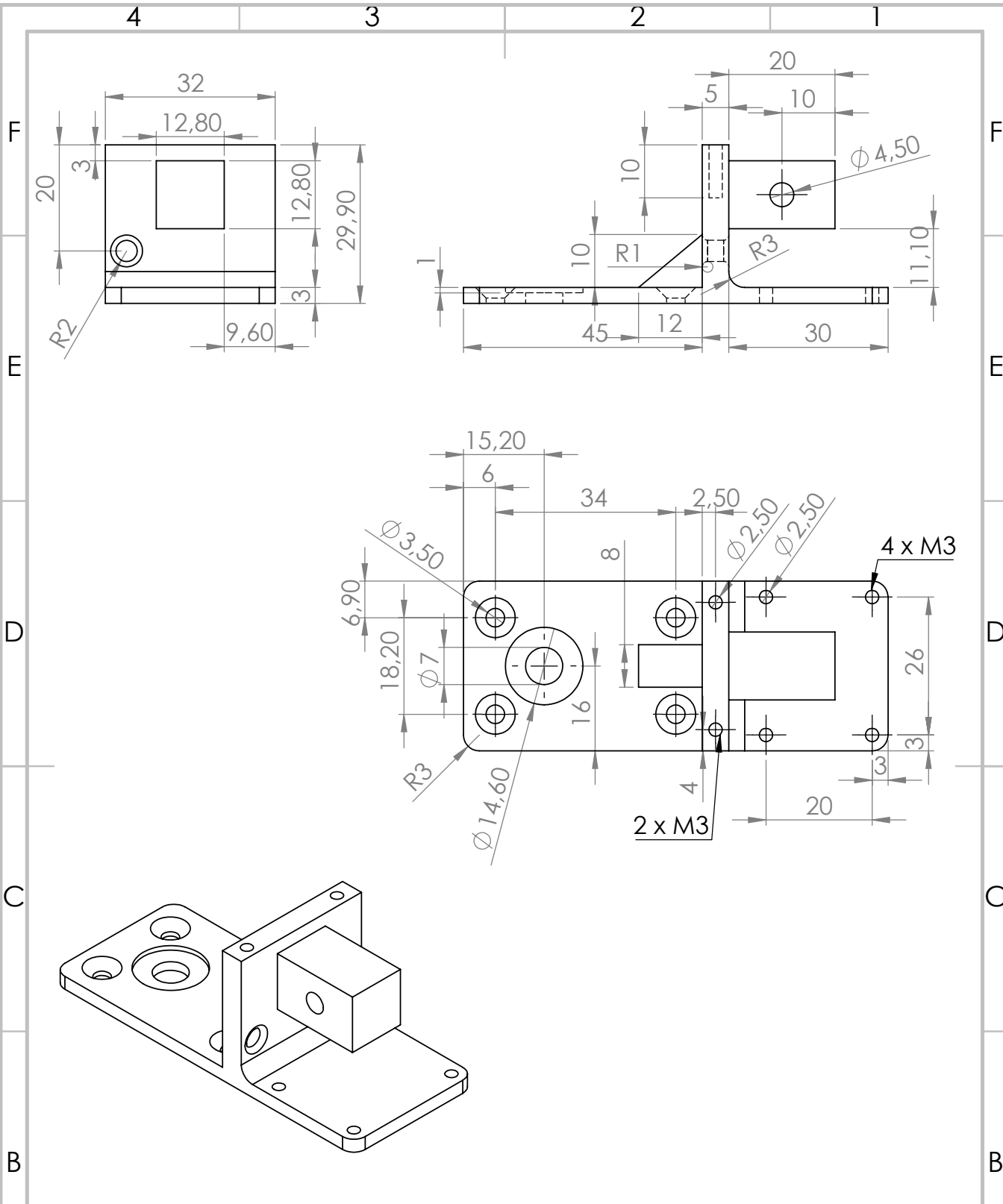
NAME	SIGNATURE	DATE
DRAWN Per Anton Ø. Olsen		14.04.18
CHK'D Johannes Wågen		
APP'VD Johannes Wågen		
MFG		
Q.A		
MATERIAL:		
3D-printed photo polymer FLGPCL03		
WEIGHT:		

DWG NO.
 motor_bracket_1

A4

SCALE:1:1

SHEET 1 OF 1



UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MILLIMETERS
SURFACE FINISH:
TOLERANCES: NS-ISO 2768-1 M
LINEAR:
ANGULAR:

FINISH:
Rinsing in isopropanol
Hardening in UV-light
Threading holes to M3

DEBURR AND
BREAK SHARP
EDGES

DO NOT SCALE DRAWING

REVISION 3

OMEGA VERKSTED

TITLE:
Motor bracket for
15mm x 15mm leg

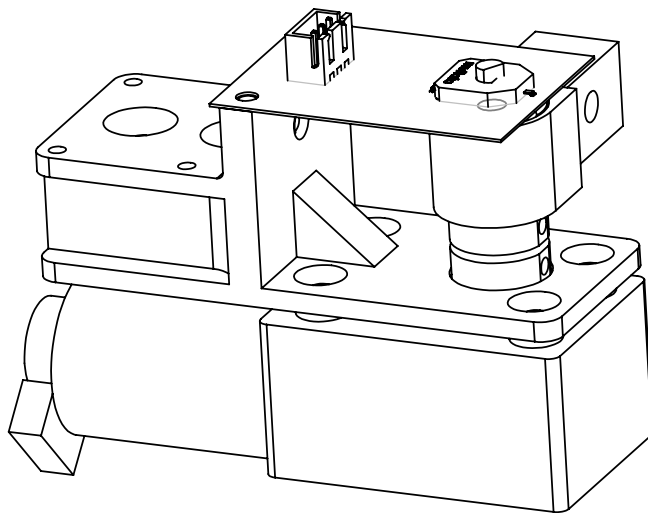
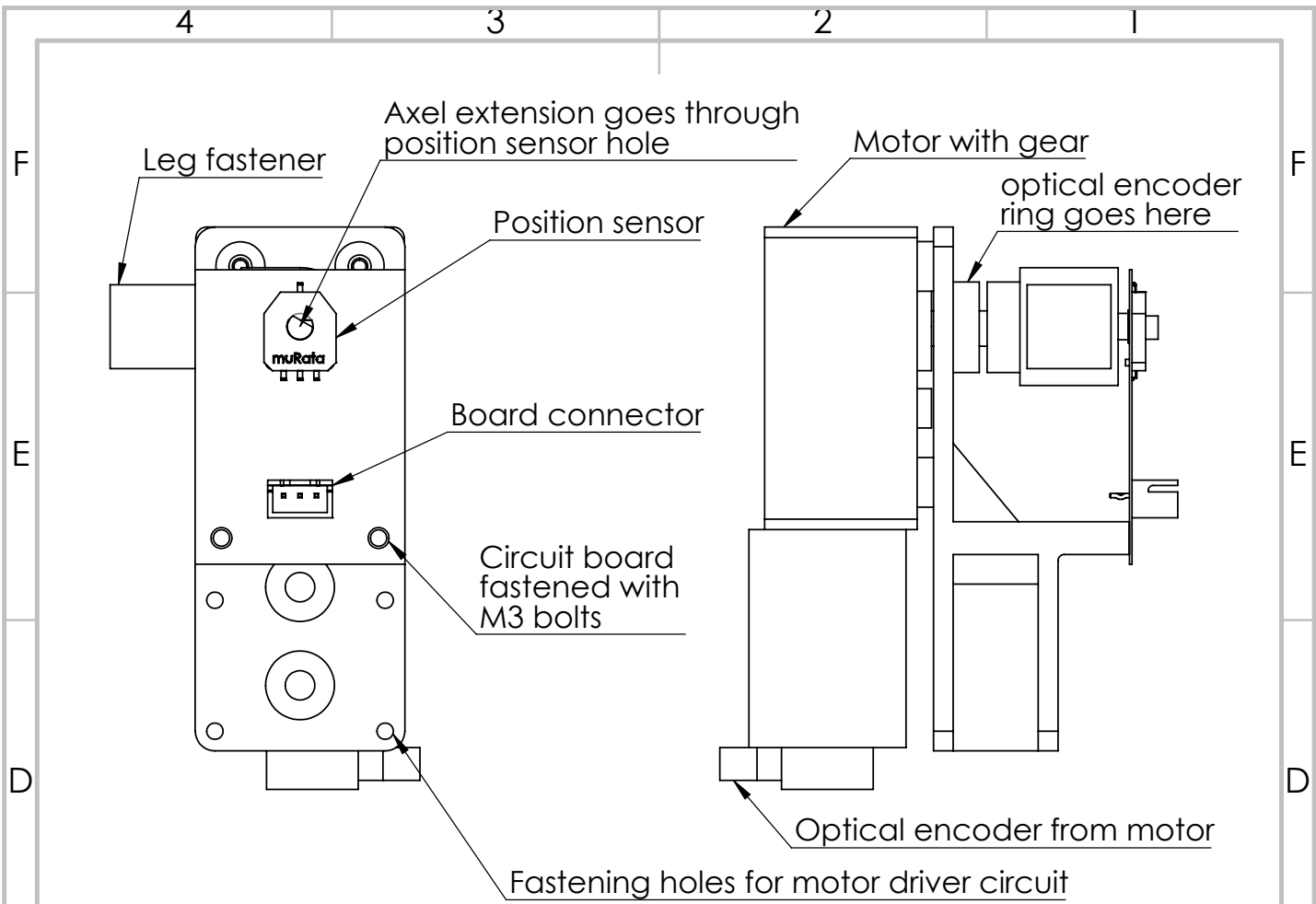
NAME	SIGNATURE	DATE
DRAWN Per Anton Ø. Olsen		13.04.18
CHK'D Johannes Wågen		
APPVD Johannes Wågen		
MFG		
Q.A		
MATERIAL:		
3D-printed photo polymer FLGPCL03		
WEIGHT:		

DWG NO.
motor_bracket_knee

A4

SCALE:1:1

SHEET 1 OF 1



UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MILLIMETERS
SURFACE FINISH:
TOLERANCES:
LINEAR:
ANGULAR:

PRODUCTION METHOD:
Assembly



DO NOT SCALE DRAWING

DEBURR AND BREAK SHARP EDGES

OMEGA VERKSTED

NAME	SIGNATURE	DATE	POST-PRODUCTION REMARKS
DESIGNER	Per Anton Ø. Olsen	14.04.18	REMARKS
DRAWN	Per Anton Ø. Olsen	09.05.18	
CHK'D	Johannes Wågen		
TESTED	Petter Breedveld		
APP'VD			

TITLE:
Motor unit assembly

OTHER COMMENTS:
For detailed description, see: center_connector_to_first_joint.shoulder_joint_assembly, driver_controller_and_palmeter.
Parts not included in this assembly: driver_controller (circuit board for motor control), encoder and encoder ring.

MATERIAL(S):
different materials

DWG NO./FILE NAME:
motor_module_with_pos_sens

A4

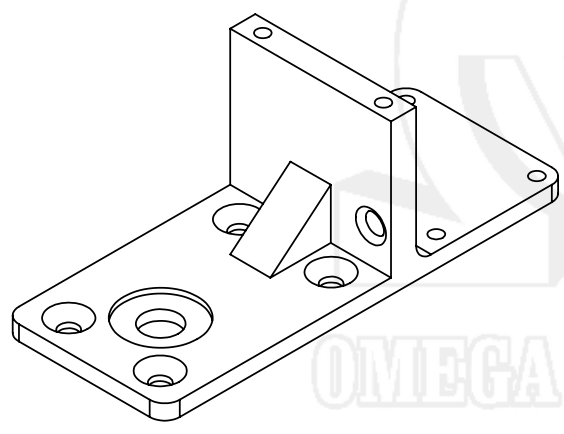
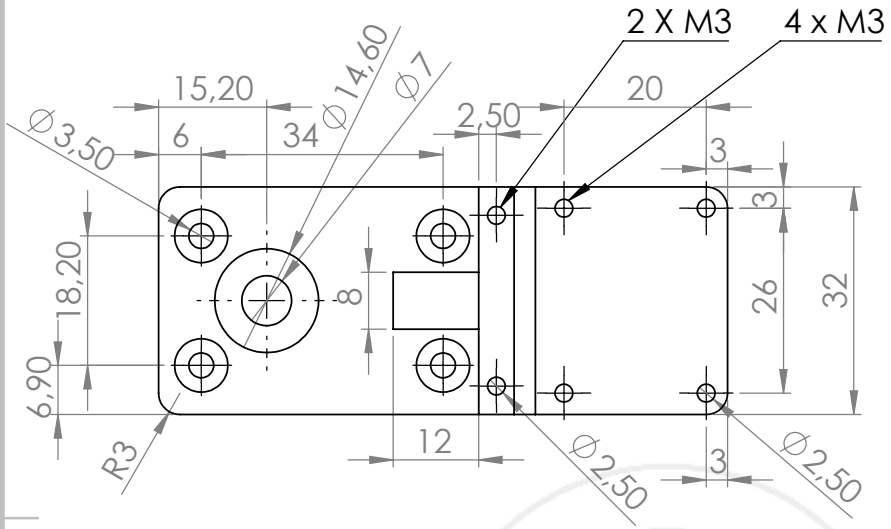
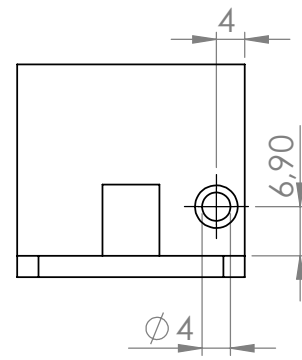
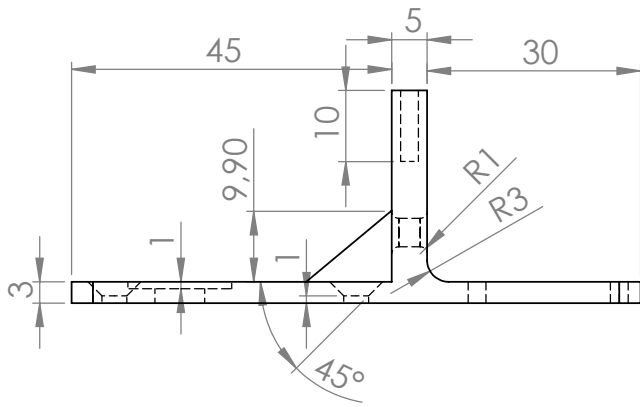
WEIGHT:

SCALE:1:2

SHEET 1 OF 1

REVISION

1



OMEGA VERKSTED

UNLESS OTHERWISE SPECIFIED:
 DIMENSIONS ARE IN MILLIMETERS
 SURFACE FINISH:
 TOLERANCES: NS-ISO 2768-1 G
 LINEAR:
 ANGULAR:

PRODUCTION METHOD:
 3D-printing



DO NOT SCALE DRAWING DEBURR AND BREAK SHARP EDGES

OMEGA VERKSTED

NAME	SIGNATURE	DATE
DESIGNER Per Anton Ø Olsen		03.05.18
DRAWN Per Anton Ø. Olsen		
CHK'D Johannes Wågen		
TESTED Johannes Wågen		
APP'VD Petter Breedveld		

POST-PRODUCTION REMARKS
 - Rinsing in isopropanol
 - Cutting of print supports
 - Hardening in UV-light
 - Drilling and threading

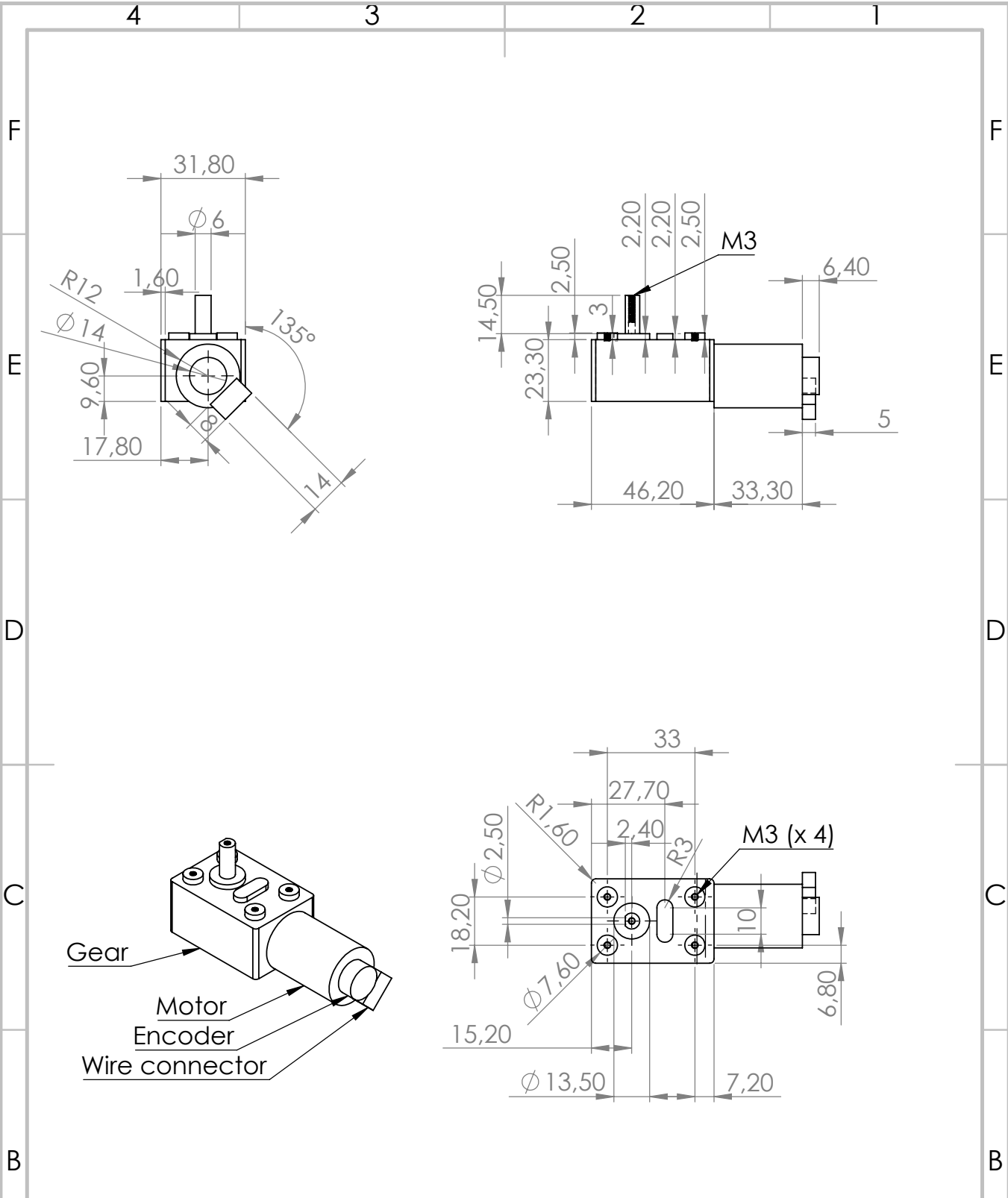
TITLE:
Standard Motor Bracket

OTHER COMMENTS:
 Holes for threading must be drilled manually to ensure optimal result.

MATERIAL(S):
 Photo polymer
 FLGPCLO3

DWG NO./FILE NAME:
 motor_bracket_standard A4

WEIGHT: SCALE:1:1 SHEET 1 OF 1 REVISION 5



UNLESS OTHERWISE SPECIFIED:
 DIMENSIONS ARE IN MILLIMETERS
 SURFACE FINISH: -
 TOLERANCES: NS-ISO 2768-1 M
 LINEAR:
 ANGULAR:

FINISH:
 No finish

DEBURR AND
 BREAK SHARP
 EDGES

DO NOT SCALE DRAWING

REVISION 6

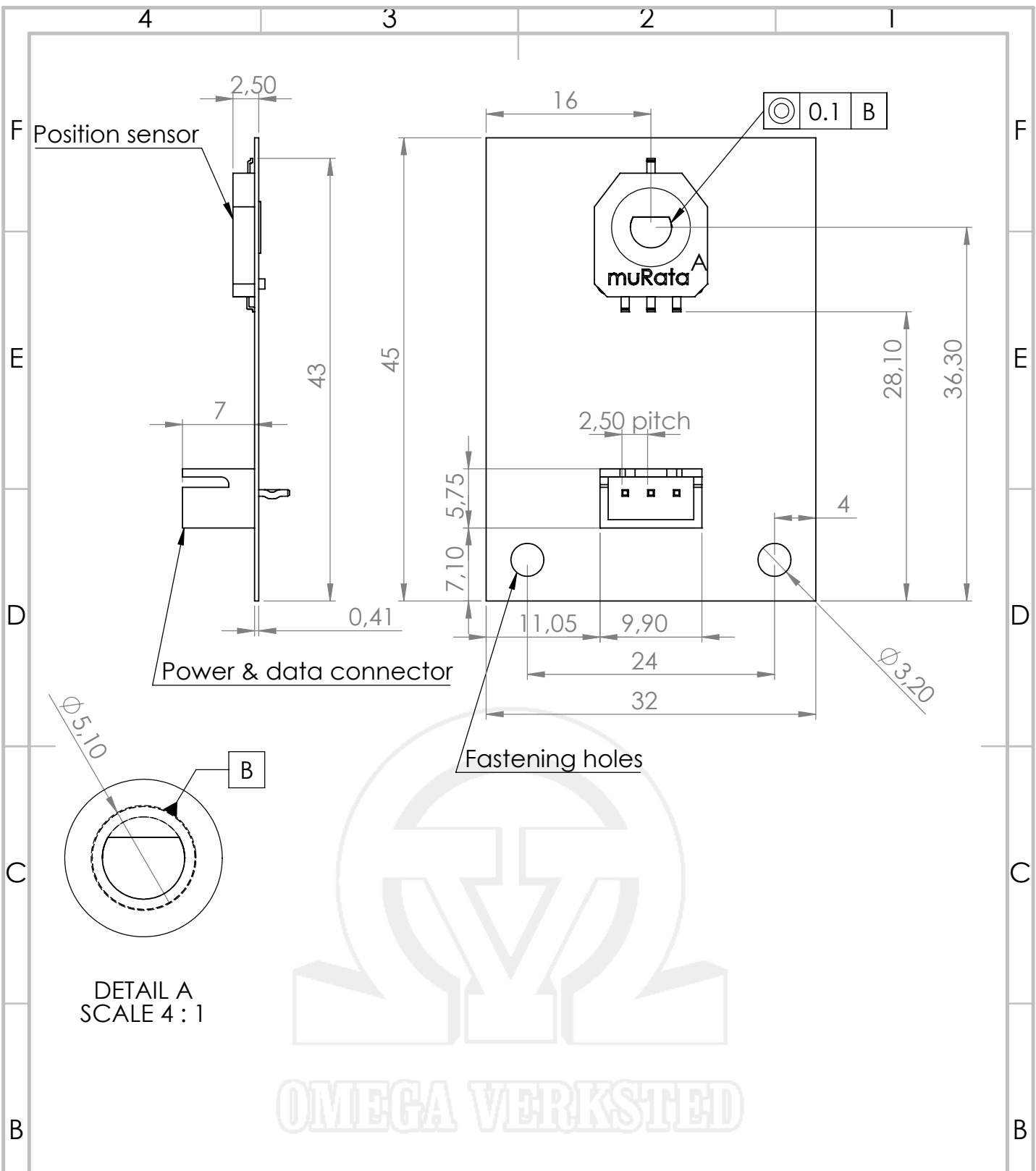
NAME	SIGNATURE	DATE
DRAWN Per Anton Ø. Olsen		16.3.18
CHK'D Johannes Wågen		
APPVD Johannes Wågen		
MFG		
Q.A		
MATERIAL: Cast aluminium and other materials		
WEIGHT: 157.4 g		

TITLE:
Motor with gear

DWG NO.
motor_with_gear

SCALE:1:2

SHEET 1 OF 1



DETAIL A
SCALE 4 : 1



UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MILLIMETERS
SURFACE FINISH:
TOLERANCES:
LINEAR:
ANGULAR:

PRODUCTION METHOD:
Soldering and assembly



DO NOT SCALE DRAWING DEBURR AND BREAK SHARP EDGES

OMEGA VERKSTED

NAME	SIGNATURE	DATE
DESIGNER Einar Henriksen		03.05.18
DRAWN Per Anton Ø. Olsen		
CHK'D Per Anton Ø. Olsen		
TESTED		
APP'VD		

POST-PRODUCTION REMARKS
remove flux from circuit board after soldering

TITLE:
Position sensor

OTHER COMMENTS:
Keep conducting surfaces away from top and bottom of circuit board after assembly. For detailed information about circuit, see "potmeter.PrjPcb"

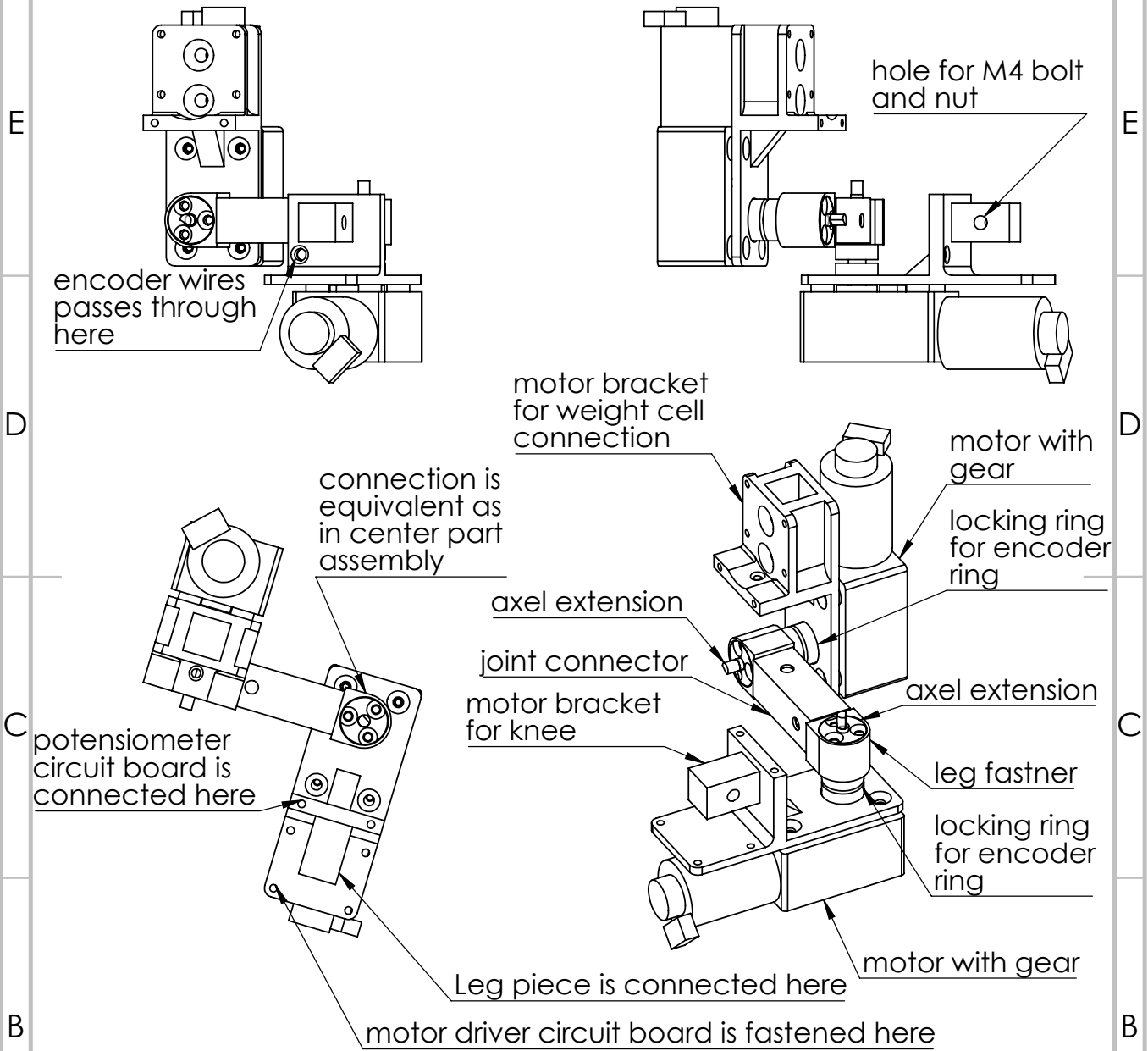
MATERIAL(S):
different materials

DWG NO./FILE NAME:
potmeter

Missing parts will be shown in other drawings:

- 1) encoders
- 2) motor driver circuit board
- 3) potentiometer circuit board

For detailed description on assembly of center parts, see "Assembly of center parts"



UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MILLIMETERS
SURFACE FINISH:
TOLERANCES:
LINEAR:
ANGULAR:

FINISH:

DEBURR AND
BREAK SHARP
EDGES

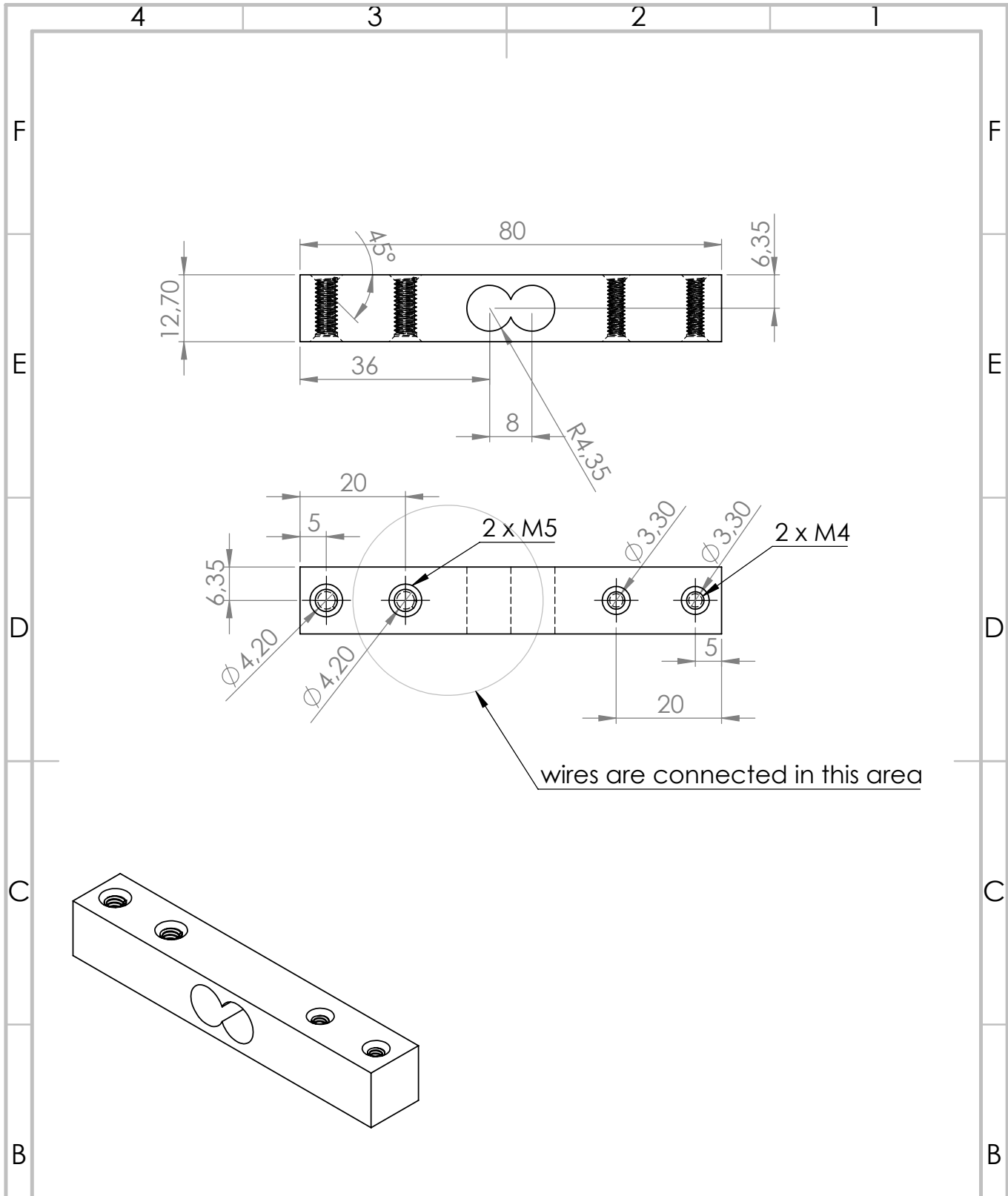
DO NOT SCALE DRAWING

REVISION 1

OMEGA VERKSTED

NAME	SIGNATURE	DATE
DRAWN Per Anton Ø. Olsen		30.04.18
CHK'D Johannes Wågen		
APP'VD Johannes Wågen		
MFG		
Q.A		

TITLE:	<h1>Assembly of shoulder joint</h1>
DWG. NO.	
MATERIAL:	<h2>shoulder_joint_assembly</h2>
WEIGHT:	
SCALE:1:5	A4
SHEET 1 OF 1	



UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MILLIMETERS
SURFACE FINISH:
TOLERANCES: NS-ISO 2768-1 M
LINEAR:
ANGULAR:

FINISH:
No finish

DEBURR AND
BREAK SHARP
EDGES

DO NOT SCALE DRAWING

REVISION 2

OMEGA VERKSTED

NAME	SIGNATURE	DATE
DRAWN Per Anton Ø. Olsen		14.04.18
CHK'D Johannes Wågen		
APPVD Johannes Wågen		
MFG		
Q.A		

TITLE: Weight cell
DWG NO. weight_cell_2
SCALE:1:1
SHEET 1 OF 1

MATERIAL:
Aluminium
and electric circuit

WEIGHT: 31.1g

A4

A.5 Annen mekanisk dokumentasjon

Disse to artiklene ble skrevet på Omega Verksteds interne wiki.

Tekniske tegninger

From OmegaV-wiki

Contents

- 1 Tekniske tegninger
- 2 Ansvarsfraskrivelse
- 3 Hva kan inngå i en teknisk tegning?
- 4 Generelt om tegninger og linjer
- 5 Generelt om målsetting
- 6 Teknisk Tegning i SolidWorks17 (3D -> 2D)

Tekniske tegninger

For å beskrive hvordan en form eller en maskindel ser ut, eller skal se ut bør standardiserte former for tekniske tegninger benyttes.

Ansvarsfraskrivelse

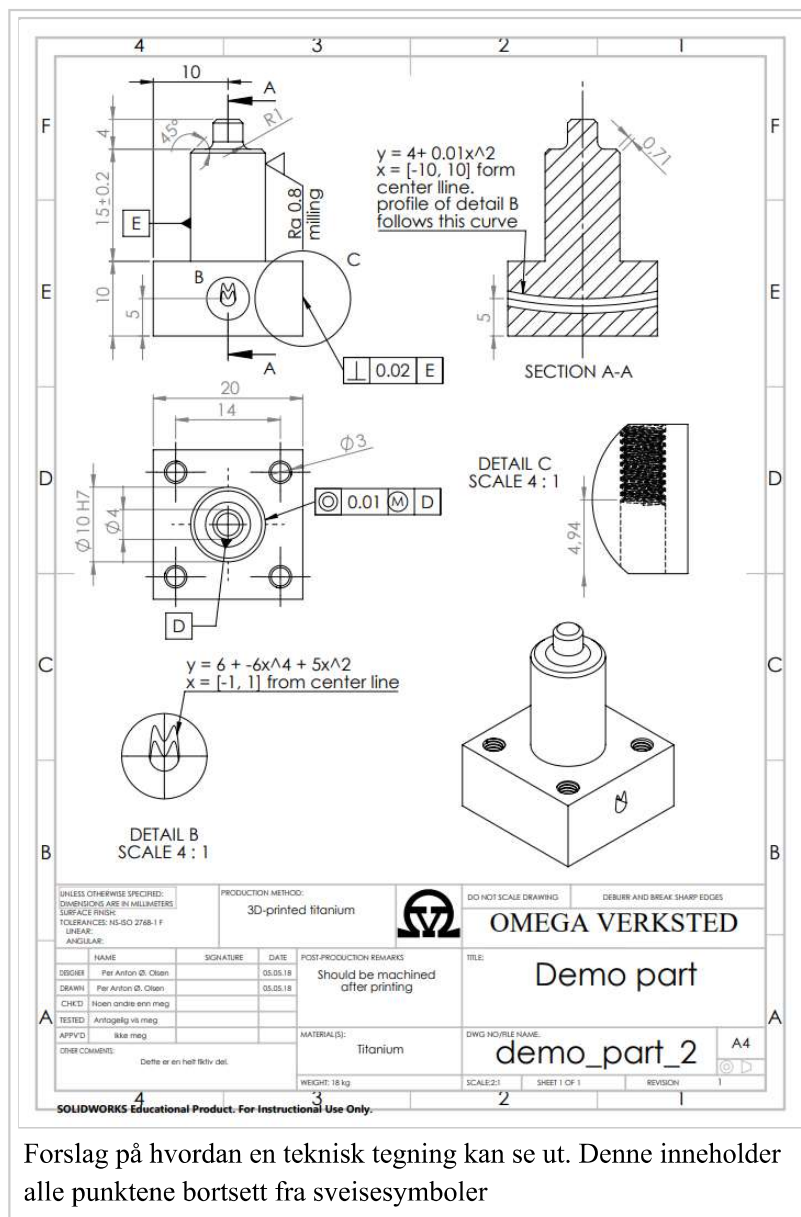
Retningslinjene som beskrives i dette notatet er ikke korrekt i forhold til reglene som tekniske tegnere må forholde seg til. Hvis man vil lære seg den korrekte måten å lage tekniske tegninger bør man bruke andre kilder. Denne artikkelen er til for at folk skal dokumentere de mekaniske tingene sine bedre. Informative tegninger er bedre enn ingen tegninger selv om de ikke følger strenge industrielle krav til punkt og prikke. For en mer detaljert beskrivelse kan man se på kapittel 2 i Verkstedhåndboka.

Hva kan inngå i en teknisk tegning?

- 1) Del(er) i tegnet fra forskjellige vinkler
- 2) Forskjellige typer målsetting på avstander vinkler osv.
- 3) Detaljer og snitt
- 4) Toleranser på mål
- 5) Overflatetoleranser
- 6) Geometriske toleranser
- 7) Kommentarer
- 8) Sveisesymboler og andre symboler
- 9) Infoboks

Generelt om tegninger og linjer

En teknisk tegning av en del eller en sammensetning av deler skal være beskrivende på en minimert måte. Dvs. prøv å unngå redundans der det ikke er nødvendig. Dersom delen er av en slik kompleksitet at den må tegnes fra flere sider for å beskrive formen, skal projeksjon benyttes. Delen kan også skaleres, og det må merkes hvilken skala delen er tegnet i. Linjer og kanter på delen som ligger bak eller inne delen som ikke vises fra den vinkelen som delen er tegnet fra kan tegnes inn med stiplede linjer. Dersom det er behov for å beskrive en detalj som kommer dårlig frem pga. liten skalering, kaotiske omgivelser eller andre grunner, kan det tegnes inn en avmerking som merkes med en unik bokstav (og eller tall hvis det blir veldig mange av disse) i tillegg til en oppskalert kopi av detaljen innenfor det avmerkede området der det passer seg. Denne detaljen skal merkes med samme bokstav som avmerkingen. Denne teknikken kan være hensiktsmessig dersom en del har kompleksitet inni som må beskrives, men som blir for kaotisk på hele delen. Stiplede linjer kan ofte bli veldig uoversiktlig. Andre måter å vise detaljer på kan være å gjøre et snitt. Da skal snittflaten skraveres med 45 graders linjer med passende avstand i mellom. På symmetriske deler kan en senterlinje tegnes inn som hjelpelinje. Senterlinjer skal tegner med lengre stipling enn vanlig stiplet linje (tenk $.. \cdot \cdot \cdot$). For å beskrive delens form på en opplysende men lite anvendelig måte kan delen tegnes i perspektiv eller isometrisk som en liten figur. Denne figuren er det ikke vanlig å sette mål på, men det kan benyttes og kan være nyttig for geometriske toleranser, overfaldetoleranser eller for å beskrive hva de ulike elementene/delene heter.



Forslag på hvordan en teknisk tegning kan se ut. Denne inneholder alle punktene bortsett fra sveisesymboler

Generelt om målsetting

Målsetning kan gjøre på mange måter, og noen måter er mer fornuftige enn andre. Generelt gjelder det å målsette slik at alle aspekter og detaljer er fullstendig beskrevet, enten gjennom et direkte eller et indirekte mål, uten at det blir for kaotisk (dette kan være vanskelig i seg selv). For at ting skal være mer konsistent finnes det noen standarder for hvordan mål skal se ut.

- 1) Prøv å benytte samme piltype på alle målepilene i en tegning (Det anbefales på det sterkeste å bruke spisse fylte piler)
- 2) Mål skal stå oppå streken til pilen, helst midt mellom pilspissene

- 3) Skriften skal alltid stå oppreist eller rotert men en vinkel inntil $\pi/2$, bortsett fra punkt 8 og 9
- 4) Prøv å benytte samme type mål og målkonvensjon over alt på tegningen
- 5) Redundante mål skal skrives i parenteser og skal ikke toleransesettes
- 6) Mål som ikke er relevante for funksjonalitet, men kun til opplysning kan merkes med NF
- 7) Mål som er relevante for funksjonalitet kan merkes med F
- 8) Mål på radier kan merkes med R <mål> og en pil som peker på radien med en vinkel. Her skal skriften stå oppå pilen og følge vinkelen til pilen
- 9) Diametermål på sirkler, (utvendig og innvendig) skal merkes med Ø <mål> og kan enten merkes på lik linje som horisontale og vertikale mål, eller som i 8, men med en avkappet pil som peker inn i sirkelen fra andre siden i tillegg med en strek som går fra spiss til spiss
- 10) På plassering av hull eller andre sirkler er det vanlig å tegne inn et rettvinklet kryss i midten av sirkelen og bruke dette som referansemål på plasseringen (ikke kanten av sirkelen)
- 11) Uvesentlige detaljer eller mål kan droppes
- 12) Toleranse skal stå etter eksakt referanse (selv om toleransen er dobbel minus eller dobbel pluss)
- 13) prøv å unngå at det kommer streker gjennom skrift eller at mål krysser andre mål. (den siste kan være umulig å unngå)

Teknisk Tegning i SolidWorks17 (3D -> 2D)

Dersom du har laget en "Part" eller en "Assembly" i SolidWorks17, kan det være veldig nyttig å lage en tegning av delen både som dokumentasjon for egen del og dersom det er noen andre som vil gjenskape eller forandre ditt design. Denne guiden viser deg hvordan du kan gå frem.

- 1) Du kan enten velge å bruke file->new->Drawing og deretter velge insert->drawing view->Model for å så velge hvilken del/sammenstilling du vil tegne, eller file->make drawing from part/assembly
- 2) Når du får opp en ny drawing vil SW spørre hva slags tegneformat du vil bruke. Du kan bruke hva du vil, men det finnes en fornuftig mal for A4 med OV-logoer og fornuftige felter på som ligger på OVsky->Brukere->noenoe.
- 3) Litt avhengig av hvilken fremgangsmåte du benyttet i punkt 1 vil du få opp en tom tegning, kanskje med en rubrikk på høyre side som kalles view palette med noen alternativer for hvilke vinkler du vil tegne delen fra.
- 4) Før du plasserer ut delen bør du høyreklikke på dokumentet på en tom plass trykke utvid-tegnet -> properties. Dersom det står under Type of projection <Third angle>, skift til <First angle>. Dette er forskjellen på amerikansk og Europeisk standard på hvordan du flipper delen din rundt. På OV-malen er det et symbol nederst i høyre hjørne med to konsentriske sirkler og et trapes. Dette symbolet beskriver at det er europeisk projeksjon på denne tegningen. Hvis det hadde vært amerikansk hadde trapesets spiss pekt mot sirkelen.

5) Nå kan du plassere ut de forskjellige projeksjonene dine. Hvis du vil vise både overside og underside av delen bør du starte med å sette ut en side. Hvis det ikke gir noen særlig gevinst å se delen i isometrisk eller perspektiv er det ikke essensielt å ha den slik.

6) Fordel de ulike vinklingene slik at du gfar plass til å sette på mål der det trengs. (de kan flyttes under veis også)

7) Hvis du har noen innvendige mål som ikke kommer godt frem kan du endre visningsmodus på en del ved å bytte fra symbolet av en boks uten stiplede linjer til en boks med stiplede linjer. Det kan hende du må stille de andre tilbake. Prøv å unngå å ha stiplede linjer på 3D-figuren.

8) Under samme meny som 7, kan du velge en annen skalering for figuren din ved å velge <use custom scale> Huska at dersom du endrer skalaen må du også sjekke at skalaen har blitt endret under <sheet format> (infoboksen nederst)-

9) Sett på mål, kommentarer, detaljer, geometriske og ordinære toleranser, sveisesymboler osv. Hvis det er aspekter du ønsker å få fram men som krever at du tegner på delen, ikke nøl med å legge til en sketch.

10) Hvis vannmerket blir i veien kan du slette det i neste punkt.

11) Når du er ferdig, eller tror du er ferdig, med å sette på mål og symboler på delen kan du redigere sheet format. Velg <sheet format> oppe til venstre og <edit sheet format>.

12) Hver rubrikk skal i prinsippet være selvforklarende. I rubrikken der det står om toleranser kan du velge toleranser som skal gjelde for hele delen, også der det ikke står noe. Eks: NS-ISO 2768-1 M (Toleranser for ikke-toleransesatte mål type Medium). Se verktøyhåndboka kapittel 2 evt. s. 19.

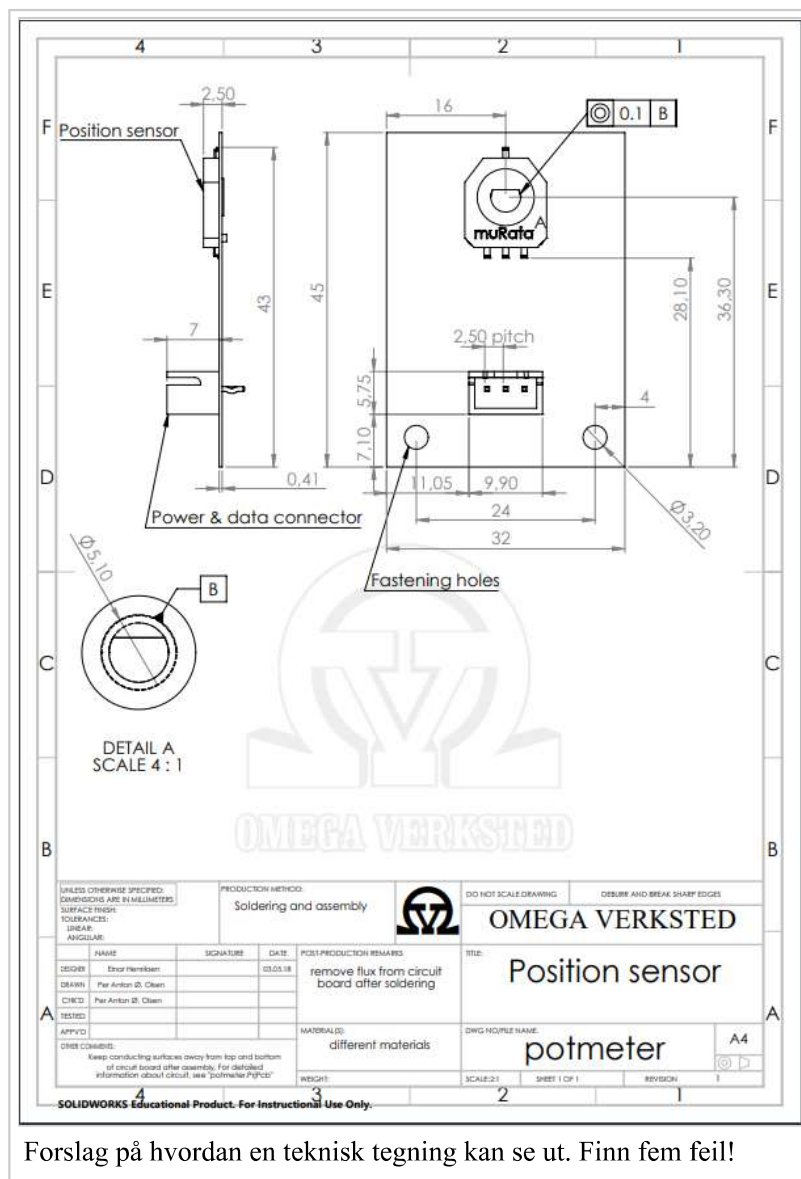
13) Husk å skrive in hvilken revisjon det er.

14) Lagre tegningen i SolidWorks sitt eget format i tilfelle noe må endres senere, og lagre som PDF i tillegg.

15) Hvis du skal skrive ut burde du åpne dokumentet i nettleser og printe derfra, for å sørge for at du ikke får krøll med papirtype.

16) Print ut et ekstra eksemplar og putt inn i permen med tekniske tegninger.

Retrieved from "https://wiki.omegav.no/index.php?title=Tekniske_tegninger&oldid=1935"



Forslag på hvordan en teknisk tegning kan se ut. Finn fem feil!

- This page was last modified on 5 May 2018, at 16:46.

3D-printing tolleranser

From OmegaV-wiki

Tolereanser ved 3D-printing på Omega Verksted

En toleranse er en beskrivelse av hvilken nøyaktighet en metrikk er innenfor. Dette kan være geometriske toleranser, som hvor eksentriske to hull er i forhold til hverandre, toleranser som beskriver ruheten på overflater eller hva som er maksimalt avvik tillat på en størrelse som vinkel eller lengdeenhet. I Verkstedhåndboka handler det andre kapittelet om toleranser. Hvis du er i tvil om hvilke toleranser som er rette for din generelle mekaniske del, anbefales det å ta en titt på NS-ISO 2768-1 (F,M,G,MG). Dette er generelle toleranser for ikke-toleransesatte mål, og mye brukt i industrien. Se s. 19 i verkstedhåndboka

Resinprinter

Resinprinterer på OV kan være veldig nøyaktig. Den har forskjellige nøyaktighetsinnstillinger, og høyere presisjon medfører lengre printetid. Det er også andre faktorer som spiller inn når det kommer til nøyaktighet.

- Hvordan man fjerner delen fra byggeplaten i printerer
- Hvor "supports" er festet på modellen
- Om det er innvendig mål eller utvendig mål
- Orienteringen til delen under printing

Tips og triks

På raskeste innstilling gjelder disse triksene:

- Ytre mål som skal passe inn i en annen del burde designes ca. 2 tiendeler mindre enn akkurat mål.
- Indre mål som skal ha en del på innsiden burde, dersom dette er over 5 mm, ha 4 tiendeler overmål (dvs. 2 tiendeler på radius), og dersom det skal være mindre kan det være lurt å designe delen slik at det er mulig å justere med for eksempel boring etter printen er ferdig.
- For indre mål som er tilstrekkelig store, for eksempel 12 mm i diameter kan det betraktes som likt ytre mål
- Dersom du skal fjerne din presisjonsdel fra byggeplaten er det fornuftig å forsiktig prøve å fjerne supports først, siden kreftene som skal til for å løsne modellen kan være nok til å deformere delen.
- Etter vanlig behandling med isopropanol og uv-herding kan support-bitene fjernes. Dersom du bruker en fin figursag eller skalpell unngår du at deler av overflaten brykker av. Sandpapir kan benyttes etterpå
- Husk også at trange hull må tømmes helt for resin før herding

Retrieved from "https://wiki.omegav.no/index.php?title=3D-printing_tolleranser&oldid=1981"

Category: Portet til Confluence

- This page was last modified on 22 May 2018, at 14:59.

A.6 Prosjektbeskrivelse

Prosjektbeskrivelse

Avansert designprosjekt for Omega verksted, TFE4205

Kunden til dette prosjektet er Omega verksted (OV). Dette er et studentverksted der alle kan bli medlem. OV selger et utvalg elektronikk komponenter til medlemmene. Hver uke blir det holdt et styremøte. Dette møtet har mandat til å godkjenne bruk av OV sine ressurser. For prosjektet vil det være ekvivalent med kundemøte.

Prosjektet har som formål å lage en tre beinet robot for OV. Denne roboten skal brukes til å skape interesse rundt robotikk på OV. For å tilrettelegge for dette skal så mye av arbeidet gjort i denne oppgave kunne brukes av andre på OV. Det vil derfor bli laget selvstendige moduler av roboten, i den grad det er mulig. For at disse modulene skal ha et kundegrunnlag må de kunne lages til en rimelig kostnad som en kan forvente at en student kan ha råd til.

Prosjektet skal dokumenteres med standardiserte maler fra OV. Der det ikke finnes maler må dette lages. Prosjektet ønsker også å utbedre kompetansen ved OV for gjennomføring av større prosjekter etter kravspesifikasjon.

Prosjektet har tre mentorer som er ferdig utdannede sivilingeniører. Disse har stilt seg disponibel til å veilede prosjektet og å utforme krav til prosjektet.

Krav:

1. Alle design filer til kretskort skal lages i det samme programmet slik at andre enkelt kan åpne det. Dette skal gjøres i:
 - a. Allium eller
 - b. Kicad
2. Komponentene skal i så stor grad som det er mulig velges fra det OV har på lager.
3. Skal kunne være kapabel til å gå. Programmering av gangen er ikke en del av oppgaven.
4. Roboten skal være selvstendig, dvs all prosessering på roboten.
5. Dokumentasjon skal følge dokumentasjonsmal fra OV. Der en mal ikke foreligger skal det lages.
6. Prosjektet skal underrette mentorer og styret i OV underveis i prosjektet.
 - a. Prosjektstyring skal gjøres med et verktøy som gjør det mulig å dele fremdrift med styret og mentorene.

Figurer

1.1	En tidlig konsepttegning fra en av de første møtene.	6
1.2	En overordnet tegning over hvordan robotens ledd er satt sammen, og hvor de generaliserte koordinatene er	7
1.3	V-modellen	8
1.4	Work breakdown diagram fra midten av prosjektet	10
2.1	en todimensjonal fremstilling av roboten, med beskrivelse av lengder, vinkler og masse-sentre på hvert ledd	14
2.2	Bokkdiagram over generell motor med girkasse-modell hentet fra [3]	21
2.3	Bokkdiagram over generell "Model Reference Adaptive Control"	22
2.4	Bevegelsesgeneratoren γ ved to-bein-stilling	23
2.5	Ønsket bevegelse av et bein fra posisjonen I_{k-1} til I_k	25
2.6	En mulig strategi for bevegelse av et ben om gangen for å oppnå fremdrift	26
3.1	Motoren som vi valgte å bruke i alle leddene	34
3.2	Motorbrakett 1; til montering på vektsensor	35
3.3	Motorbrakett 2; til montering på firkantrør	35
3.4	Motorbrakett 3; til mer generelle prosjekter	35

3.5	Benfeste for firkantrør	36
3.6	monteringsring for optisk skive for enkoder	36
3.7	Flens med akseltapp for å passe inn i potmeter	37
3.8	Fot for roboten	37
3.9	Vektsensorer som monteres ut fra midten på plattformen	38
3.10	Senterfeste til vektsensorer og plattform	38
4.1	IMU-modulen	41
4.2	IMU-modulen	42
4.3	Kretsdigram av HX711	43
4.4	Lastcelle-modulen	43
4.5	Lastcelle-modulen	44
4.6	Testoppsett for testing av motor	45
4.7	Plot av motor torque ved spenning	45
4.8	Blokkdiagram over Allegro A4950 motor driver	47
4.9	Motordriveren	47
4.10	Motordriveren	48
4.11	Potmeterkortet, etset på OV	48
4.12	Beagle Bone Black	49
4.13	Waveshare RS485 CAN Cape	50
6.1	Delvis sammensatt ben av roboten	55
A.1	Skjematikk av potmeterkort	60

A.2 Skjematikk av lastcelle-modul	61
A.3 Skjematikk av lastcelle-modul	62
A.4 Skjematikk av motordriver	63
A.5 Skjematikk av motordriver	64
A.6 Skjematikk av IMU-modul	65
A.7 Utlegg av lastcelle modul	66
A.8 Utlegg av IMU-modul	67
A.9 Utlegg av motordriver	68
A.10 Utlegg av potmeterkort	69
A.11 Utlegg av potmeterkort	69

Tabeller

2.1	Oversikt og forklaring av symboler som blir brukt i seksjonen	17
2.2	Denavit-Hartenberg-diagram fra det initielle aksekorset på overflaten til bein nr. i	18
3.1	Materialegenskaper for fotopolymeret FLGPCL03 [14]	31
3.2	Materialegenskaper for karbonfiber hentet fra [14]	32
3.3	Materialegenskaper for aluminium hentet fra [20] og [7]	32
4.1	Motor moment Spenning	46
A.1	BOM: motordriver	70
A.2	BOM: imu-modul	71
A.3	BOM: lastcelle-modul	72

Bibliografi

- [1] Brian Chen. *MPU-9250 SPI Library for Arduino/Teensy*. Mai 2017. URL: <https://github.com/brianc118/MPU9250>.
- [2] Christine Chevallereau mfl. «RABBIT: A Testbed for Advanced Control Theory». I: *IEEE Control Systems Magazine* (okt. 2003), s. 57–77.
- [3] Olav Egeland og Tommy Gravdahl. *Modeling and Simulation for Automatic Control*. Marine Cybernetics AS, 2002. ISBN: 82-92356-01-0. URL: <http://www.marinecybernetics.com>.
- [4] embeddedartistry.com. *embeddedartistry/embedded-resources*. Apr. 2017. URL: <https://github.com/embeddedartistry/embedded-resources>.
- [5] Leonid B. Freidovich mfl. «A Passive 2-DOF Walker: Hunting for Gaits Using Virtual Holonomic Constraints». I: *IEEE Transactions on Robotics, Vol 25, No.5* (nov. 2009), s. 1202–1207.
- [6] Harald Hareide. «Creation and Control of Gaits for the SemiQuad Robot». I: *NTNU Master Thesis* (mai 2014), s. 1–76.
- [7] Hartvig Hartvigsen mfl. *Verkstedhåndboka Utgave 6. -*. Gyldendal Undervisning, 2016. ISBN: 978-82-05-35707-5. URL: -.
- [8] Petros A. Ioannou og Jing Sun. *Robust Adaptive Control*. Dover Books on Engineering. Dover Publications, 2012. ISBN: 978-0-486-49817-1. URL: -.
- [9] Yun-Jung Lee og Shigeo Hirose. «Three-Legged Walking for Fault Tolerant Locomotion of a Quadruped Robot with Demining Mission». I: *Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems* (2000), s. 973–978.
- [10] Sylvan Martel mfl. «Three-Legged wireless Miniature Robots for Mass-scale Operations At the Sub-Atomic Scale». I: *Proceedings of the 2001 IEEE International Conference on Robotics & Automation* (mai 2001), s. 3423–3428.

- [11] Yoichi Masuda og Masato Ishikawa. «Simplified Triped Robot for Analysis of Three-Dimensional Gait Generation». I: *Journal of Robotics and Mechatronics Vol.29No.3* (feb. 2017), s. 528–534.
- [12] Sebastian O H Madgwick. «An efficient orientation filter for inertial and inertial/magnetic sensor arrays». I: *Her maa det puttes inn noe* (apr. 2010).
- [13] Per Anton Ø. Olsen. «Analysis and Design of Gaits for Three-Legged Walker». I: *Project report NTNU* (jun. 2018), s. 1–40.
- [14] Per Anton Ø. Olsen mfl. «Prosessrapport EiT». I: *NTNU EiT Report* (mai 2017), s. 1–27.
- [15] Peter VAN OVERSCHEE og Bart DE MOOR. «Subspace Identification of Purely Deterministic Systems». I: *KLUWER ACADEMIC PUBLISHERS* (jan. 1996), s. 1–257.
- [16] Anton Shiriaev, John W. Perram og Carlos Canudas-de-Wit. «Constructive Tool for Orbital Stabilization of Underactuated Nonlinear Systems: Virtual Constraints Approach». I: *IEEE Transactions On Automatic Control, Vol.50 No 8* (aug. 2005), s. 1164–1175.
- [17] Anton Shiriaev mfl. «Periodic motion planning for virtually constrained Euler-Lagrange systems». I: *Systems & Control Letters 55* (jun. 2006), s. 900–907.
- [18] Mark W. Spong, Seth Hutchinson og M. Vidyasagar. *Robot Modeling and Control*. John Wiley & Sons, Inc, 2006. ISBN: 978-0-471-64990-8.
- [19] Siddharth Talesra. *HX711*. Des. 2016. URL: <https://github.com/getsiddd/HX711>.
- [20] Various. *Aluminium*. Mai 2018. URL: <https://en.wikipedia.org/wiki/Aluminium>.
- [21] Various. *Main page*. Jun. 2018. URL: <https://formlabs.com/>.