CHALMERS



TARS - Modell av en gående fyrbent robot

Design, konstruktion och styrning av en nedskalad modell inspirerad av roboten TARS från filmen Interstellar

Kandidatarbete vid avdelningen för System- och Reglerteknik på institutionen för Elektroteknik

HAMPUS ANDERSSON
NARON BERISHA
STEPHIE DO
DIVYA KARA
HENRIK PETTERSON
IVAR SÖRQVIST

Institutionen för Elektroteknik
Chalmers Tekniska Högskola
Göteborg, Sverige 2019

Kandidatarbete 2019: EENX15-19-20

TARS - Modell av en gående fyrbent robot

Design, konstruktion och styrning av en nedskalad modell inspirerad av roboten TARS från filmen Interstellar

HAMPUS ANDERSSON
NARON BERISHA
STEPHIE DO
DIVYA KARA
HENRIK PETTERSSON
IVAR SÖRQVIST



CHALMERS

Institutionen för Elektroteknik
Avdelningen för System- och Reglerteknik
EENX15-19-20
Chalmers tekniska högskola
Göteborg, Sverige 2019

TARS - Modell av en gående fyrbent robot
Design, konstruktion och styrning av en nedskalad modell inspirerad av roboten TARS från filmen Interstellar
Hampus Andersson, Naron Berisha, Stephie Do, Divya Kara, Henrik Petterson, Ivar Sörqvist

© HAMPUS ANDERSSON, 2019.
© NARON BERISHA, 2019.
© STEPHIE DO, 2019.
© DIVYA KARA, 2019.
© HENRIK PETTERSON, 2019.
© IVAR SÖRQVIST, 2019.

Handledare: Maryam Razi, Postdoc, Elektroteknik
Examinator: Nikolce Murgovski, Docent, Elektroteknik

Kandidatarbete 2019: EENNX15-19-20
Institutionen för Elektroteknik
Avdelningen för System- och Reglerteknik
EENX15-19-20
Chalmers tekniska högskola
SE-412 96 Gothenburg
Telefon +46 31 772 1000

Omslag: Fysisk modell på projektets robot TARS.

Skriven i LaTeX
Göteborg, Sverige 2019

Förord

Rapporten är ett kandidatarbete som utförts på Chalmers tekniska högskola av sex civilingenjörsstudenter från sektionerna Elektroteknik, Maskinteknik samt Automation och Mekatronik. Utbildningens omfattning är fem år och arbetet genomfördes på våren under utbildningens tredje år.

Projektgruppen vill uppmärksamma särskilda personer som bidragit till arbetets utveckling. Bland annat projektgruppens handledare, Maryam Razi som har väglett och hjälpt till med beräkningar och struktur genom hela projektets gång. Nikolce Murgovski och kandidatgruppen EENX15-19-16, för de regelbundna råden de har givit. Rikard Karlsson som har haft hand om beställningen av alla komponenterna. Chalmers Robotförening och Prototyplabbet som har bidragit med lokal, material och utrustning för byggandet av den fysiska modellen. Till sist, Reine Nohlborg på prototyplabbet som har hjälpt till med rådgivning och vattenskärning av delarna.

Abstract

TARS is a robot that got attention through the movie *Interstellar*, directed by Christopher Nolan. It has a seemingly simple design with four rectangular blocks connected to each other which constitutes each of the robot's legs.

This thesis is written in Swedish and has been conducted at the division Signals and Systems, which is part of the department of Electrical Engineering at Chalmers University of Technology. The aim of the project was to construct and regulate a robot with similar properties and design as the robot TARS. Delimitations such as a controlled environment, budget and limited time period were set. The robot was constructed in a way that enables three different types of walking styles to be implemented whereof at least one of the walking styles was to be actualized through a physical model. A detailed digital model in the program Autodesk Inventor has also been made for which has been used as a base for the simulation in MATLAB Simulink. Furthermore, calculations of the construction and control have been performed and tests on the components have been conducted with the intent of validating functionality.

A model was constructed through the preparations, which was able to walk accordingly to two walking styles. The steady construction made it very sensitive to changes, but through repeated improvements of the control, the model was able to walk consistently without falling. The main goal of the project has been achieved although some minor changes can be made in order to optimize the product.

Keywords: TARS, Interstellar, construction, remote-controlled robot, walking robot, digital, physical modell, Arduino, control, Simulink.

Sammanfattning

TARS är en robot som har fått uppmärksamhet genom filmen *Interstellar*, regisserad av Christopher Nolan. Den har en till synes enkel design med fyra rektangulära block som är sammankopplade med varandra vilket utgör var och en av robotens ben.

Projektet har genomförts vid avdelningen för System- och Reglerteknik på institutionen för Elektroteknik på Chalmers tekniska högskola. Syftet med projektet var att konstruera och styra en robot med liknande egenskaper och design som TARS. Utifrån det ställdes krav på att minst tre gångstilar skulle vara implementerbara varav minst en skulle förverkligas. För projektet har även avgränsningar såsom en tidsram, budget samt att modellen ska testas i en kontrollerad miljö satts. En detaljerad digital modell i programmet Autodesk Inventor skapades för konstruktion men kunde även användas bland annat som bas för simulering i MATLAB Simulink. Vidare har beräkningar för konstruktion och styrning genomförts samt även utförliga tester för att säkerställa bland annat komponenters funktionalitet.

Till följd av förberedelserna konstruerades en fysisk modell som klarade av att gå enligt två olika gångstilar. På grund av den stadiga konstruktionen var systemet mycket känsligt för förändringar. Genom iterativ förbättring av robotens styrning lyckades den gå flera steg i följd utan att välta. Projektets huvudsakliga syfte uppnåddes men vissa detaljer skulle kunna förbättras för att optimera produkten ytterligare.

Nyckelord: TARS, Interstellar, konstruktion, fjärrstyrd robot, gående robot, digital, fysisk modell, Arduino, styrning, Simulink.

Innehåll

Förkortningar **vii**

1 Inledning **1**
1.1 Bakgrund . . . 1
1.2 Syfte . . . 2
1.3 Avgränsningar . . . 2
1.4 Frågeställningar . . . 2
1.5 Samhälleliga och etiska aspekter . . . 3
1.6 Rapportens disposition . . . 3

2 Framtagning av koncept **4**
2.1 Liknande robotar . . . 4
2.2 Inverterad dubbelpendel . . . 5
2.3 Materialegenskaper . . . 6
2.4 Valt koncept . . . 7
2.5 Tillvägagångssätt . . . 9

3 Test och beräkning av komponenter **10**
3.1 Kritiska vinklar . . . 10
3.2 Stegmotor axel . . . 12
3.3 Linjär aktuator . . . 14
3.3.1 Krav på längdskillnad . . . 15
3.3.2 Beräkningar av kulskruv . . . 15
3.3.3 Val av drivkort . . . 18
3.4 Accelerometer . . . 19
3.5 Batteri . . . 20
3.6 Blåtandskommunikation . . . 20

4 Design och konstruktion **22**
4.1 Modellering . . . 22
4.2 Stressanalys . . . 26
4.3 Tillverkning och montering . . . 27

5 Simulering och styrning **28**
5.1 Simulering . . . 28
5.2 Styrning . . . 30
5.2.1 Metod för styrning av roboten . . . 30
5.2.2 Programmering . . . 31

6 Resultat **35**
6.1 Modell . . . 35
6.2 Simulering och styrning . . . 37

7 Diskussion **42**
7.1 Konstruktion . . . 42

7.2 Komponenter 43
7.3 Simulering och styrning 45
7.4 Tester 46

8 Slutsats **47**

Referenser **48**

Bilagor **I**
A Kretsschema I
B Styrning III
C Budget IV
D Gångtabeller V
E Ritningar VII

Förkortningar

ABS	Acrylonitrile Butadiene Styrene
CAD	Computer Aided Design
CRF	Chalmers Robotförening
I/O	Input/Output
I2C	Inter Intergrated Circuit
PLA	Polylactic Acid
PWM	Pulse Width Modulation
RPM	Revolution Per Minute
TP	Tyngdpunkt

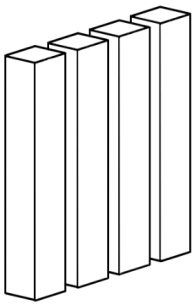
1

Inledning

För att ge en överblick över projektet kommer bakgrunden, syftesformuleringen och avgränsningar att behandlas nedan. Även en detaljerad problembeskrivning och en diskussion om samhälleliga- och etiska aspekter kring projektet att föras. Till sist kommer rapportens disposition att tas upp.

1.1 Bakgrund

TARS är en robot som presenteras i Christopher Nolans science fiction-film *Interstellar* som hade premiär år 2014 [1]. I filmen åker en grupp astronauter, tillsammans med robotarna TARS och CASE, ut i rymden för att hitta en ny planet där mänskligheten kan bosätta sig då jorden är hotad. TARS är tillsynes en enkelt uppbyggd robot av fyra rektangulära block som fungerar som ben, likt Figur 1.1. Trots den enkla visuella bilden TARS ger så är det en komplex robot att konstruera. Blocken kan röra på sig individuellt eller tillsammans med ett annat block som ett system. Med detta kan TARS gå på olika sätt vilket visas i filmen. I filmen kommunicerar TARS med en mänsklig röst och används bland annat vid räddningsuppdrag vid riskfyllda miljöer.



Figur 1.1: *Skiss av fyra rektangulära block som vardera representerar ett av robotens ben.*

Många robotar byggs för att efterlikna människan eller djur, exempelvis har Boston Dynamics [2], byggt den hundliknande roboten Spot som konstruerats med flertalet leder. I jämförelse med detta är TARS ett mer minimalistiskt alternativ. Färre rörliga delar kan ses som begränsande men leder också till en mindre komplex tillverkning.

Liknande kandidatarbeten kring att skapa en robot som efterliknar TARS har två gånger tidigare utförts på Chalmers tekniska högskola [3], [4]. På internet kan även flera försök till att återskapa TARS hittas, vissa mer lyckade än andra. En av de bättre lösningarna har en användare på Youtube

vid namn Akira B Higaki gjort [5]. Roboten är en nerskalad version som är byggt av bland annat 3D-utskrivna plastdelar och aluminium. På videon visas hur roboten kan ta sig fram genom att lyfta och sänka sina ben med hjälp av sex stycken servomotorer. Då dokumentation av tidigare arbeten är en värdefull tillgång studeras de vidare och problemen som tidigare uppkommit analyseras.

Automatiserade system är idag mer aktuella än någonsin. Dessa system möjliggör utförande av uppgifter som antingen underlättar eller som inte är genomförbara för människan. Robotar är ett perfekt exempel av detta och är därför ett intressant område att utforska. Roboten TARS som både är rörlig och dynamisk med dess anpassningsbara gångstilar, skulle kunna användas vid just denna typ av automatiserade system.

1.2 Syfte

Syftet med projektet är att ta fram en modell av en fjärrstyrd modell av TARS från filmen Interstellar. Roboten simuleras, konstrueras och styrs för att minst tre olika gångstilar skall vara applicerbara, varav minst en av gångstilarna realiseras med en fysisk presentation.

1.3 Avgränsningar

Roboten kommer endast att testas i miljöer utan yttre påverkan. En sådan miljö skulle kunna vara labbmiljö med plan yta utan externa krafter. Inga krav på att möjliggöra för roboten att svänga eller gå vid en viss hastighet kommer att ställas. Den sociala interaktionen likt filmen kommer inte heller att tas i beaktande. Roboten kommer dessutom att skalas ner jämfört med den som presenteras i filmen. Faktorer som en tidsplan på en termin och tidigare kunskap inom projektgruppen är begränsande. En budget på en maximal kostnad av 5000 sek har satts för projektet.

1.4 Frågeställningar

Utifrån syftet har följande frågeställningar skapats:

- Hur kan en robot designas och konstrueras så att den efterliknar TARS?
- Hur kan en simulering liknande de verkliga gångstilarna tas fram?
- Hur kan en robot styras för att efterlikna TARS gång?

1.5 Samhälleliga och etiska aspekter

Roboten som har byggts i projektet bedöms inte påverka samhället på ett betydande sätt då dess enda egentliga förmåga är att gå på ett eller flera sätt. Projektet har därför inte tagit i beaktande hur TARS påverkar till exempel arbetssituationer eller människors säkerhet på det sätt som robotar som är tänkta att användas i eller av samhället.

1.6 Rapportens disposition

I nästkommande kapitel presenteras en förstudie om liknande koncept och teorier som studerats och som eventuellt lett fram till det valda konceptet som beskrivs under samma kapitel. Vidare beskriver kapitel 3 komponenter som använts och tester som utförts på komponenterna. Kapitel 4 och 5 beskriver tillvägagångssätten och metoden som använts för designen och konstruktionen respektive simuleringen och styrningen. Resultatet av projektet presenteras i kapitel 6 och en diskussion kring resultatet och förbättringsmöjligheter förs i kapitel 7. Slutsatser kring projektet dras till sist i Kapitel 8.

2

Framtagning av koncept

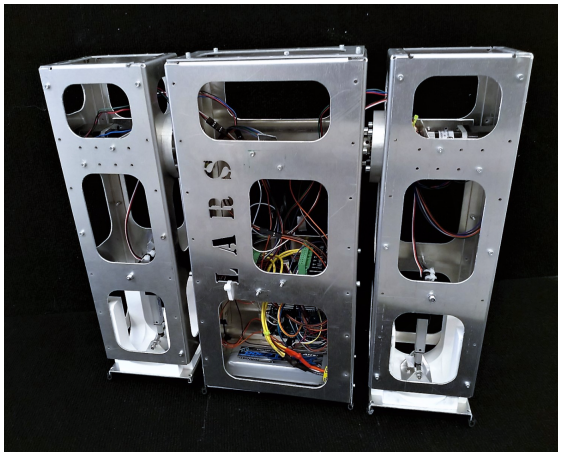
Här presenteras en förstudie på tidigare liknande koncept men även olika typer av material och modeller som ansågs vara relevanta för projektet. Detta analyserades i syfte att hitta fördelar och nackdelar i försök att optimera projektet. Sist i kapitlet ges en överblick över tillvägagångssättet som användes för att realisera konceptet.

2.1 Liknande robotar

En modell av TARS har två gånger tidigare konstruerats som kandidatarbete på Chalmers tekniska högskola. De båda robotarna TARS och LARS som skapades 2018 visas i Figur 2.2.



(a) *Kandidatarbetet TARS. Från [3]. Återgiven med tillstånd.*

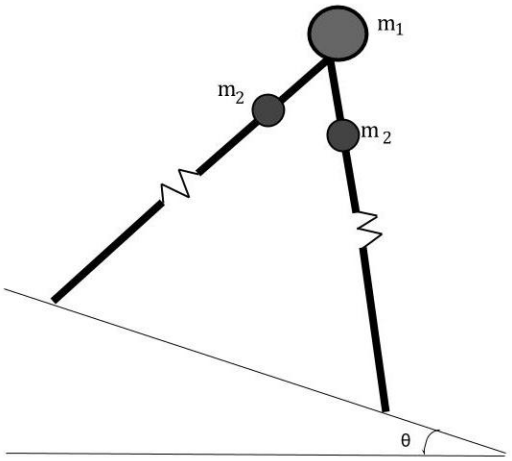


(b) *Kandidatarbetet LARS. Från [4]. Återgiven med tillstånd.*

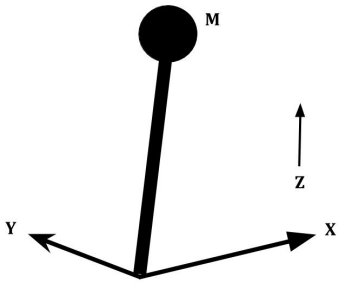
Figur 2.2: *Bilder från tidigare kandidatarbeten 2018.*

Robotarna tycktes ha problem med balans men enligt båda rapporternas slutsats hade detta kunnat lösas med hjälp av ett reglersystem [3], [4]. LARS hade även mekaniska problem som glapp i komponenter och att skruvar låste upp sig själva. TARS tog sig framåt genom att en axel förde upp och sedan ut benen medan LARS kunde skjuta ut och in sina yttre ben samtidigt som den roterade kring en axel. Från båda robotarna gavs möjligheten att använda komponenter för detta projekt.

Andra robotar som skulle kunna likna TARS är humanoida robotar som idag finns i större utsträckning. På 90-talet utvecklade företaget Honda en humanoid robot *Honda humanoid robot* som hade egenskapen att gå liknande en människa [6]. Idag finns det många olika varianter av dessa typer av robotar, bland annat tävlade cirka 40 robotar från olika plattformar i fotboll 2015 [7]. Dessa robotar kan beskrivas som bipedala robotar med en aktiv styrning. Passiva bipedala robotar går även att finna exempel på, en i mängden är *Dexter Mk III* skapad vid University of Manitoba [8]. Denna går med hjälp av gravitationen på ett lutande plan, en illustrerande bild har skapats för att visualisera detta, se Figur 2.3. Genom att detta plan bibehåller samma lutning klarar den att med en jämn steglängd gå 1500 steg utan att ramla.



Figur 2.3: *Bild på passiv bipedal robot på ett lutande plan.*

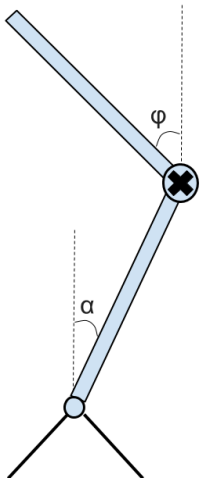


Figur 2.4: *Illustration av en inverterad dubbelpendel i 3D.*

Två vanliga sätt att välja att kontrollera och modellera humanoid bipedal gång är antingen att göra det genom att känna till alla dess dynamiska parametrar som massor och längder eller genom att endast modellera dess masscentrum och försöka reglera systemet utifrån det [9]. Från det andra tillvägagångsättet är det då möjligt att först modellera varje ben som en inverterad pendel i ett 3-dimensionellt rum, se Figur 2.4. Den tidigare nämnda passiva bipedala roboten går att på liknande sätt modellera som en inverterad dubbelpendel, se Figur 2.3 [10].

2.2 Inverterad dubbelpendel

Tidigare nämnda bipedala robotar modelleras oftast som inverterade dubbelpendlar. Detta ger en bra approximativ modell för dess rörelse och är ett enkelt sätt att abstrahera det komplexa systemet [11]. En dubbelpendel består av två pendlar varav ena är fäst vid den andras ände, se Figur 2.5.



Figur 2.5: *Illustration av en inverterad dubbelpendel.*

Tillsammans bildar de ett dynamiskt system vars ändring i initialtillstånd ger stora skillnader i systemets bana [12]. En dubbelpendel hänger från en punkt medan en inverterad dubbelpendel står upprätt, se Figur 2.5. För att balansera en sådan pendel kan olika metoder användas, en är att utföra en plan rörelse vid fästpunkten mot basen och en annan är istället att tillföra ett vridmoment på samma punkt. Det går även att applicera ett vridmoment på punkten mellan de två pendlarna. En inverterad dubbelpendel har flera jämviktslägen som den kan dras mot och är ett ostabilt system vilket innebär att minsta yttre påverkan kan göra att den lämnar ett jämviktstillstånd. Denna dynamiska egenskap gör systemet intressant att studera inom ett reglersammanhang. En variant av den inverterade dubbelpendel är Acrobot [13]. Den har en styrande del i leden mellan pendlarna och är fri i leden som är fäst vid basen. En sådan modell kan klara av att stabilisera sig själv med hjälp av reglering i den styrande leden. Detta är en underaktuerad modell vilket betyder att antalet frihetsgrader är fler än de kontrollerade lederna [14]. Paralleller mellan detta system och roboten TARS skulle kunna dras.

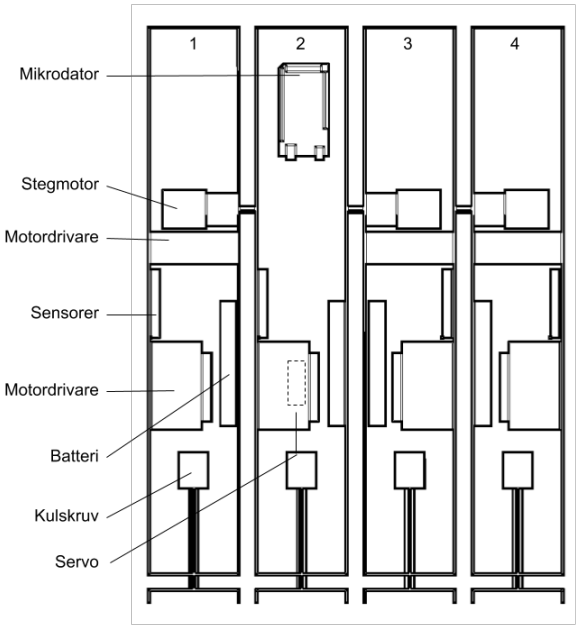
2.3 Materialegenskaper

För val av material till chassit har stål och aluminium valts att undersökas då dessa är starka och lätt formbara material. Stål är ett material med hög densitet och klarar av väldigt höga krafter innan det deformeras eller går sönder. Jämförelsevis är aluminium ett mer duktilt material som kan klara av att deformeras mer och absorbera mer energi innan det går sönder. Till följd av duktiliteten deformeras aluminium därmed tidigare än stål samt är ett mycket lättare material. Aluminium har dock ett mycket bättre förhållande mellan vikt och hållfasthet än stål [15].

Modellen kommer att kräva flera små fästanordningar för olika komponenter men kan även användas för strukturellt viktiga delar som del av fötterna. Vid 3D-skrivning kan PLA (*Polyactic Acid*) eller ABS (*Acrylonitrile Butadiene Styrene*) användas. Av dessa är ABS ett mycket mer duktilt material

som klarar av högre krafter jämförelsevist med PLA. PLA är ett svagare material men lättare att kontrollera på grund av dess låga smältpunkt, vilket gör det väldigt bra till att snabbt tillverka prototyper. PLA är ett material som inte kan användas inom områden med höga temperaturer gentemot ABS. Detta gör att ABS är mycket populärt inom industrin medan PLA är mer populärt vid hobbyverksamhet [16]. Utöver detta är PLA ett biologiskt nedbrytningsbart material vilket är mer miljövänligt [17].

2.4 Valt koncept



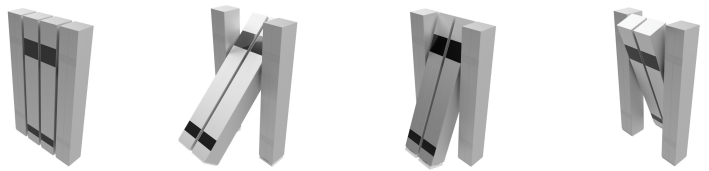
Figur 2.6: *Skiss av insidan på det valda konceptet med utlistade komponenter.*

Den design som valts för att realisera modellen av TARS från filmen Interstellar är en konstruktion bestående av fyra ben med en fot i varje ben. Det fungerar således att fötterna på de fyra benen går att skjuta in och ut med hjälp av en linjär aktuator. I lederna mellan varje ben finns en motor som kan rotera benet en viss vinkel relativt det intilliggande benet. Med dessa funktioner kan olika gångstilar förverkligas, se skiss i Figur 2.6. Styrenheten i modellen var en mikrodator som valdes så att den försåg projektet med en stor mängd digitala *input/output*-pinnar (I/O). Det finns också krav på PWM (*Pulse Width Modulation*) för att kunna styra ett servo som skulle kunna användas för att fästa de två mellersta benen. Mikrodatorns syfte var att kommunicera med en yttre styrenhet som agerade som en extern kontroll med interna sensorer för att mäta vinklar och styra interna komponenter. En översikt över de huvudsakliga funktionerna och deras valda lösning kan ses i Tabell 2.1.

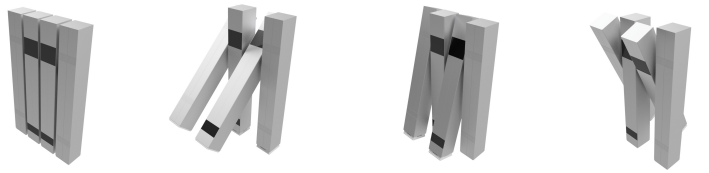
Tabell 2.1: *Komponenter som använts för att uppfylla olika funktioner för det valda konceptet.*

Funktion	**Komponent**
Rotera led runt axel	Stegmotor med växellåda
Förlänga/förkorta benen	Stegmotor med ledskruv
Låsa mittenbenen	Servomotor
Kommunikation	Blåtand
Mäta vinkel	Accelerometer
Styrning	Arduino Mega 2560
Strömförsörjning	Litium polymer batterier
Chassi	Aluminiumplåt

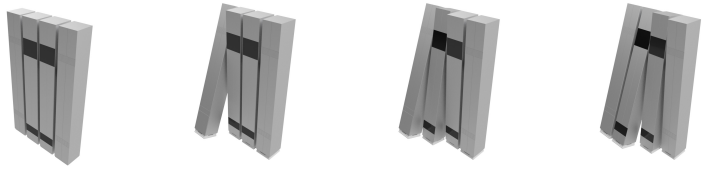
Förutom de namngivna komponenterna i Tabell 2.1 tillkom drivkort till vardera motor, säkringar, spänningsomvandlare, vippbrytare och ändlägesgivare för att realisera produkten. Modellen av TARS gjordes ungefär i skalan 1:3. Utifrån bedömningen att TARS var 1,80 m hög. Från detta valdes då följande mått på modellen: höjd: 61,70 cm, bredd: 44,35 cm och djup: 10,00 cm. Skalan valdes utifrån material, komponenternas behov av utrymme men även utifrån budgetens perspektiv. Exempelvis var mikrodatorn utav en viss bestämd storlek och behövde därför ett bestämt utrymme i innanmätet. Utöver detta var komponenternas placering viktig för att styra positionen av robotens masscentrum. Modellens chassi valdes att tillverkas av aluminium för att vara stabil och lätt.



(a) *Gångstil 1*



(b) *Gångstil 2*



(c) *Gångstil 3*

Figur 2.7: *Presentation av de tre olika gångstilarna, siffrorna representerar även prioriteten som satts för att förverkliga gångstilarna.*

För att uppfylla syftet togs tre olika gångstilar fram, vilket kan ses i Figur 2.7. Den första gångstilen enligt Figur 2.7a, fungerar som så att de två mellersta benen roterar fram och sedan dras de två yttersta benen fram. Enligt andra gångstilen, se Figur 2.7b, går modellen av TARS med vartannat ben synkroniserat och under det tredje och sista gångstilen går modellen med ett ben i taget, se Figur 2.7c. Gångstil 1 är den gångstilen som prioriterats som först att realisera följt av gångstil 2 och 3. Fortsättningsvis kommer benämningarna gångstil 1, 2 och 3 att syfta på dessa gångstilar.

2.5 Tillvägagångssätt

För att förverkliga konceptet från idé till modell beräknades vilka vinklar som krävdes för att föra ut benen i syfte att få roboten att tippa över. Parallellt med detta beräknades hur långt fötterna behövde förlängas samt förkortas för att de andra benen inte skulle vara i kontakt med marken vid rotation. Samt effektbehovet från komponenter beräknades för att välja batterier och för motorerna beräknades krävda vridmoment som de måste leverera. Andra komponenter valdes för att passa in i funktion, storlek och budget. Under projektets gång utfördes även fysiska tester för att se till att komponenterna klarade av de ställda kraven. En del tester genomfördes genom att införskaffa en komponent och testa om den klarade av kraven innan fler köptes in, andra för att testa redan tillhandahållna komponenter. Parallellt med detta modellerades alla delarna upp i modelleringsprogrammet Autodesk Inventor för att skapa ritningar och för att kunna få värden att utföra beräkningar, som vridmoment och stress. Modellen som skapats sattes in i programmet Simulink, MATLAB för att se om konceptet fungerade före implementationen av en fysisk produkt. När detta säkerställts skapades ett detaljerat kretsschema för att smidigt visualisera kopplingar mellan komponenter. Slutligen genomfördes den fysiska konstruktionen av modellen tillsammans med kabeldragning och montering av komponenter samt tillämpning av det designade styrprogrammet.

3

Test och beräkning av komponenter

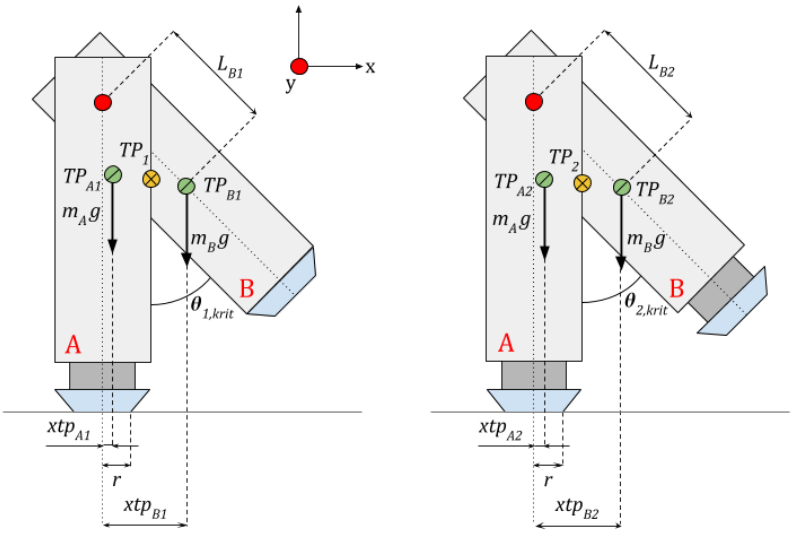
I detta kapitel presenteras beräkningar som gjorts på designen och för att bland annat välja komponenter. Även tester som genomförts på kritiska komponenter kommer att beskrivas. Resultaten av testerna på komponenter presenteras i respektive tests delkapitel.

3.1 Kritiska vinklar

Av de tre presenterade gångstilarna i Figur 2.7 krävdes det för gångstil 1 och 2 att roboten tippar framåt när den för fram sina ben, se Figurerna 2.7a och 2.7b. Eftersom roboten hade möjligheten att aktuera sina ben ut och in behövdes två fall beräknas. Dessa två fall illustreras i Figur 3.1. I båda fallen är fötterna till benen som är i kontakt med marken utskjutna tillräckligt mycket för att de andra benen ska kunna rotera fritt kring sin axel. Vid utåtvridet läge har det första fallet sina utfällda ben helt inne och i det andra fallet är de utfällda benen utskjutna till maximal längd.

Det som avgör när roboten tippar över är när den totala tyngdpunktens lodräta linje hamnar till höger om den punkt där det stående benet slutar ha kontakt med marken, det vill säga avståndet r från origo i x-led. Origo illustreras i Figur 3.1 som en röd ifylld punkt och de totala masscentrumet, även kallat tyngdpunkten (TP), som en gul punkt med ett kryss i. Vidare kommer det att skrivas om två benpar, detta för det förutsätts att TARS-modellens gångstil enligt Figurerna 2.7a och 2.7b rör sig symmetriskt. Detta innebär att ben 1 och 3 samt 2 och 4 rör sig synkroniserat. Benparen visas som ben A och B i Figur 3.1.

Massan för vardera ben i benparet betecknas som m_A och m_B i figur Figur 3.1, vilket innefattar massan av benet, foten och komponenterna. I samma figur illustreras tyngdpunkten i varje ben som TP_{Aj} och TP_{Bj} med streckmarkerade gröna punkter, där j symboliserar antingen fall 1 eller 2. Avståndet i x-led från origo till tyngdpunkterna är xtp_{Aj} respektive xtp_{Bj}. Benens tyngdpunkter är inte centrerade i mitten av benens bredd, dock så pass nära mitten i x-led att det framöver i beräkningarna antas att xtp_{Bj} följer benet med en rak linje längs origo eftersom avståndet mellan det verkliga och det antagna masscentrumet endast skiljer sig med några få millimeter. Se de gröna streckade punkterna på de utfällda benen, B, i Figur 3.1.



(a) *Fall 1, där foten är inskjuten.* **(b)** *Fall 2, där foten är utskjuten.*

Figur 3.1: *De två fallen för när roboten tippar framåt.*

Längden L_{Bj} kommer därför vara avståndet från axeln ut till benets masscentrum och avståndet från origo till tyngdpunkten xtp_{Bj} i x-led för benet som lyfts kan räknas ut enligt

$$xtp_{Bj} = L_{Bj} \cdot sin(\theta_{j,krit}). \tag{1}$$

Den totala tyngdpunkten i x-led för benparen där $k = 2$, vilket beskriver antalet ben, fås av 2. Hädanefter är $A = 1$ och $B = 2$ i ekvationerna.

$$x_{TPj} = \frac{\sum_{i=1}^{k} m_{ij} \cdot xtp_{ij}}{\sum_{i=1}^{k} m_{ij}} = \frac{m_{1j}xtp_{1j} + m_{2j}xtp_{2j}}{m_{1j} + m_{2j}}. \tag{2}$$

Då roboten har fötter med en liten avrundning, se Figur 3.1, för att enklare falla över kommer $x_{TPj} = r$ där r är avståndet från origo till stödytans kant för foten i x-led. Genom insättning av (1) i (2) och utbrytning av den kritiska vinkeln $\theta_{j,krit}$ erhålls

$$\theta_{j,krit} = \sin^{-1}\left(\frac{r(m_{1j} + m_{2j}) - m_{1j}xtp_{1j}}{m_{2j}L_{2j}}\right). \tag{3}$$

Från kommande Kapitel 6.1 om CAD som genomfördes parallellt med beräkningarna, kunde parametrarna som används i (3) fås från CAD-modellen. Resultatet för de olika kombinationernas

kritiska vinkel då de slutgiltiga massorna för CAD-modellen som visas i Tabell 6.1 presenteras i Tabell 3.1.

Tabell 3.1: *Resultat för de kritiska vinklarna i olika kombinationer för när ben B antingen har foten inskjuten, fall 1, eller foten utskjuten, fall 2.*

Ben A	**Ben B**	**Fall 1,** $\theta_{1,krit}$	**Fall 2,** $\theta_{2,krit}$
1	2	24,79°	24,55°
2	1	25,70°	25,46°
3	4	27,48°	27,22°
4	3	27,46°	27,20°

Det som gör att de kritiska vinklarna inte är densamma för alla olika kombinationer är på grund av att benens massor skiljer sig åt då varje ben innefattar olika komponenter, vikterna kan ses i Tabell 6.1. Ett tyngre ben som lyfts gör att den totala tyngdpunkten lättare kommer över stödytan som gör att roboten faller framåt och tvärtom gäller om ett lättare ben lyfts.

För gångstil 1 enligt Figur 2.7a rör ben 1 och 4 samt 2 och 3 tillsammans. På samma sätt som den kritiska vinkeln för benparen beräknades kan då den kritiska vinkeln för alla fyra ben beräknas enligt (4), där $k = 4$. Här är symboliseras ben A av index 1 och 4 samt ben B av index 2 och 3.

$$\theta = \sin^{-1}\left(\frac{r(m_{1j} + m_{2j} + m_{3j} + m_{4j}) - m_{1j}xtp_{1j} - m_{4j}xtp_{4j}}{m_{2j}L_{2j} + m_{3j}L_{3j}}\right) \tag{4}$$

Då vinklarna skiljer sig något från de olika benparen på grund av de olika massorna beräknas även de kritiska vinklarna då alla ben används. För gångstil 1 där ben 1 och ben 4 samt ben 2 och 3 rör sig tillsammans blir resultaten enligt Tabell 3.2. Enligt beräkningar bör således en vinkel på minst 27,48° sättas som referensvinkel för gångstil 1 för att det i alla fall ska garanteras att roboten faller framåt.

Tabell 3.2: *Kritiska vinklar för gångstil 1.*

Ben A	**Ben B**	**Fall 1,** $\theta_{1,krit}$	**Fall 2,** $\theta_{2,krit}$
1 & 4	2 & 3	26,12°	25,87°
2 & 3	1 & 4	26,59°	26,33°

De kritiska vinklarna som visas i Tabellerna 3.1 och 3.2 gäller för ett statisk tillstånd. I verkligheten när roboten lyfter benen kommer ett moment från den dynamiska rörelsen när benen lyfts göra att en mindre vinkel kommer att krävas.

3.2 Stegmotor axel

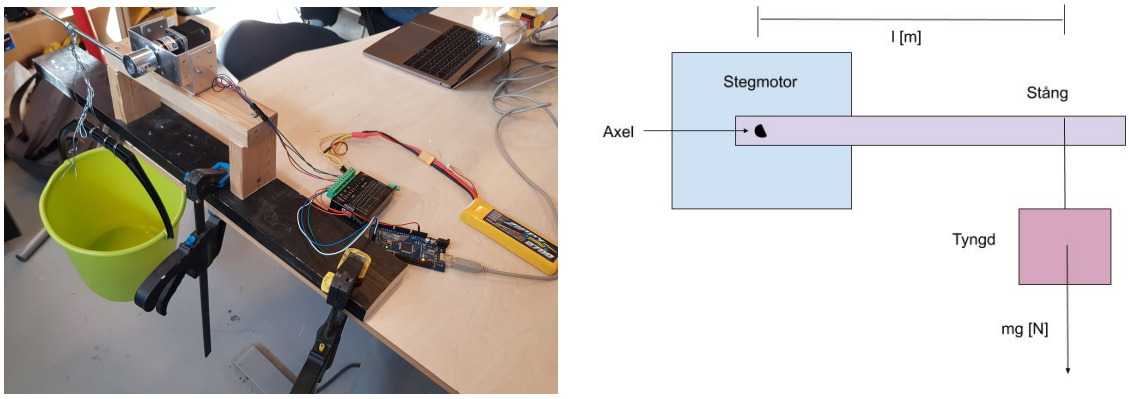
Med hjälp av värden för de vinklar som behövdes uppnås för att roboten skulle tippa över kunde beräkning av vilket vridmoment stegmotorn i axeln behöver ge ut för att komma upp i denna vinkel

göras. Detta gjordes med hjälp av följande ekvation

$$T = L \cdot \sin\theta \cdot mg. \tag{5}$$

Där g är gravitationskonstanten, θ vinkeln mellan benen, L är längden från axeln till masscentrum för det tyngsta benet och m är det tyngsta benets massa baserat på dåvarande CAD-modell. Genom att använda den maximala vinkeln som tidigare beräknats till 27,48 ° från Tabell 3.1, massan 2,7 kg och längden 0,13 m. Räknades det ut att motorn var tvungen att leverera 1,59 Nm.

Från LARS, ett av de tidigare kandidatarbeterna [4] kunde två stycken Nema-17 växlade stegmotorer användas till detta arbete [18]. Då den hade ett hållmoment på 0,30 Nm skulle den teoretiskt sett inte klara av att hålla uppe benet vid den kritiska vinkeln. För att undersöka vidare vilket vridmoment motorn klarade av att leverera i en kontinuerlig rörelse utformades ett test. Uppställningen för testet gjordes enligt Figur 3.2, ett detaljerat kretsschema kan ses i Bilaga A.



(a) *Bild tagen på uppställningen för testet.* **(b)** *Grafisk beskrivning av uppställningen.*

Figur 3.2: *Uppställning utav test av axelns stegmotor.*

På motorns axel monterades en stång vinkelrätt, 15 cm ut på denna axel hängde vikter för att se vid vilken tyngd motorn inte längre klarade av att lyfta vikterna. På grund av begränsning av vikter i CRFs lokaler användes blytyngder med en massa på cirka 250 g styck. För att motorn skulle klara av en viss vikt behövde den rotera förbi 90°. Detta gjordes genom att den fick börja 10° nedanför, rotera 20° förbi och sedan tillbaka till ursprungsläget. Första mätningen av maximala lasten gjordes vid den rotationshastighet drivkortet ”TB6600” kunde driva motorn vid ingen last, även det taget från förra årets kandidatarbete [4]. Därefter dubblades hastigheten för varje mätpunkt och samma test som tidigare nämnts, gjordes. Detta test utfördes på fyra mätpunkter för varje mikrosteg.

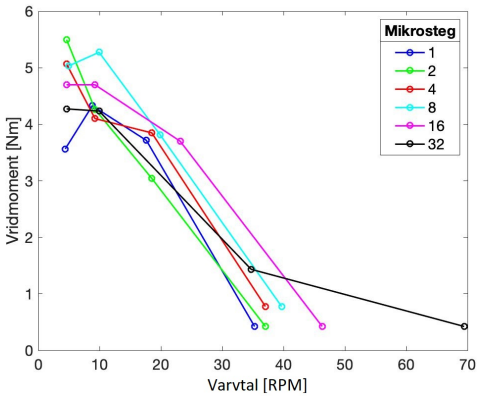
De totala vridmomenten vid varje hastighet kunde sedan beräknas genom

$$T = l \cdot m \cdot g, \tag{6}$$

där l är längden mellan stegmotorn och viktens tyngd på axeln, m är massan på den maximala vikten samt g, gravitationskonstanten. Eftersom hastigheten styrdes genom att ändra tiden mellan varje styrpuls från mikrodatorn (Arduino mega 2560) [19] kunde hastigheten räknas ut i varv per minut (RPM, från *Revolution Per Minute*) enligt

$$n = \frac{60/t}{c \cdot u \cdot s}. \tag{7}$$

Där t är tiden mellan två steg, c är antal mikrosteg, u är utväxlingen på växellådan och s antal steg som krävs för att snurra motorn ett varv. Resultatet från testet presenteras i Figur 3.3.



Figur 3.3: *Test av maximalt vridmoment vid olika hastigheter.*

Från grafen går det att särskilja att motorn var starkare vid lägre varvtal vilket är i linje med hur stegmotorers kraftutveckling går till [20]. Motorn uppvisade inga problem att vrida ut 90° med varje bens vikt om det skulle begäras, momentet som behöver övervinnas för att komma till denna vinkel är samma som i (5), men med $\theta = 90°$. Dock kunde den inte göra det snabbare än cirka 30 rpm.

3.3 Linjär aktuator

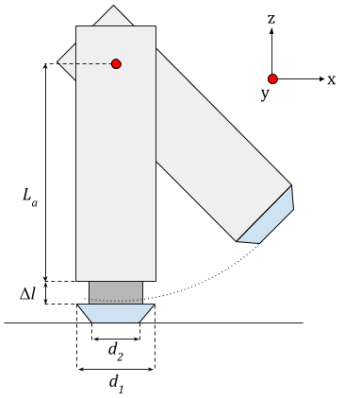
Detta kapitel beskriver först hur mycket en fot behövde lyftas för att kunna svänga runt utan att slå emot mark, sedan beräknades det på vilken motor som var stark nog och slutligen valdes motordrivare.

3.3.1 Krav på längdskillnad

Enligt konceptet i Kapitel 2.4 skulle roboten byggas så att varje ben kunde förlängas eller förkortas med hjälp av en linjär aktuator, bestående av en stegmotor i kombination med kulskruv. För att få en uppfattning av hur långt fötterna behöver skjutas ut för att de vridande benen ska kunna vinklas ut utan att ha kontakt med marken beräknas det först genom följande ekvation

$$\Delta l = \sqrt{L_a^2 + \left(\frac{d}{2}\right)^2} - L_a. \tag{8}$$

Där L_a är avståndet från marken till axeln, d är djupet av foten och Δl är fotens förlängning. För L_a =0,4 m och d = 0,1 m blir längden som foten behöver fällas ut minst cirka 3,11 mm.



Figur 3.4: *Visar hur foten skjuts ut för att tillåta de andra benet att vridas fram utan att slå i marken.*

För att minska vinkeln som modellen faller över kan hörnen på foten kapas och därmed minska djupet på foten, om djupet på foten minskar till 0,066 m minskar höjden som foten behöver lyftas till 1,45 mm.

3.3.2 Beräkningar av kulskruv

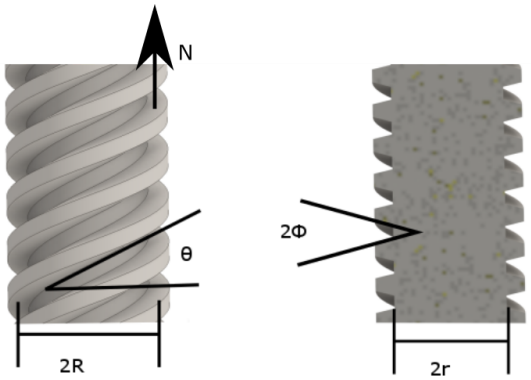
För att veta vilket vridmoment som krävdes för att hålla roboten uppe, dra in foten och trycka ut foten har följande beräkningar av hur kraft translateras från motor genom kulskruv gjorts med ledning från [21].

Först beräknades vridmomentet T_{upp} som krävdes för att lyfta foten, detta gjordes genom att jämviktsekvationer ställdes upp enligt:

$$\uparrow : -mg + N\cos(\phi)\cos(\theta) - \mu_1 N\sin(\theta) = 0 \tag{9}$$

$$\overset{\circlearrowleft}{M_z} : T_{upp} - mg\mu_2 R - rN\cos(\phi)\cos(\theta) - rN\mu_1\cos(\theta) = 0 \tag{10}$$

Där m är fotens massa, N är normalkraften på kulskruvens gängor, ϕ är halva vinkeln mellan över- och undersida på en gänga. θ är stigvinkeln för gängorna, se Figur 3.5.



Figur 3.5: *Visar vart variablerna Θ, Φ, r, R och N i (9) och (14) är lokaliserade på kulskruven. Värt att notera är att R endast går till halva gängdjupet.*

μ_1 är friktionskoefficienten mellan gängorna. μ_2 är friktionskoefficienten mellan kulskruven och foten. Genom att lösa ut N och kombinera med (10) fås

$$T_{upp} = mg\mu_2 R + mgr\Big(\frac{\cos(\phi)\sin(\theta) + \mu_1\cos(\theta)}{\cos(\phi)\cos(\theta) - \mu_1\sin(\theta)}\Big). \tag{11}$$

För att beräkna vridmomentet som krävs för att trycka foten ned T_{ner} och på så sätt lyfta roboten är förfarandet mycket det samma som för T_{upp}.

$$\downarrow : \frac{Mg}{4} - N\cos(\phi)\sin(\theta) + \mu_1 N\sin(\theta) = 0 \tag{12}$$

$$\overset{\circlearrowleft}{M_z} : -T_{ner} + \frac{Mg}{4}\mu_2 R + rN\cos(\phi)\cos(\theta) + rN\mu_1\cos(\theta) = 0 \tag{13}$$

Här är M den total vikten som roboten beräknades ha enligt dåvarande CAD. Tillsammans ger (12)

och (13)

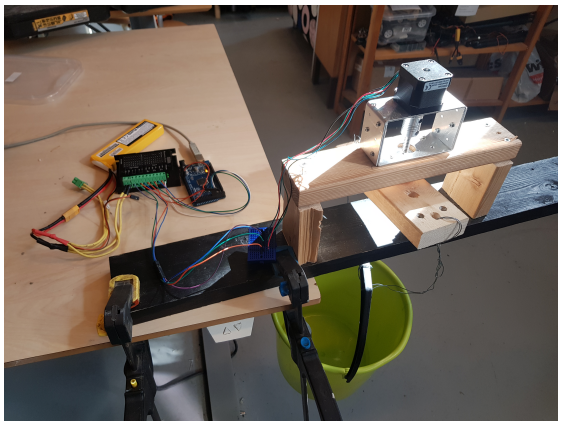
$$T_{ner} = \frac{Mg\mu_2 R}{2} + \frac{Mgr}{2}\Big(\frac{-\cos(\phi)\sin(\theta) + \mu_1\cos(\theta)}{\cos(\phi)\cos(\theta) + \mu_1\sin(\theta)}\Big). \tag{14}$$

Då uträkningar var för fallet vid jämvikt krävs ett vridmoment över T_{upp} och T_{ner} för att röra foten upp eller trycka benet upp. Genom att använda värden presenterade i Tabell 3.3 för de framtagna ekvationerna kunde vridmomenten beräknas till $T_{upp} = 0{,}16\,\text{Nm}$ och $T_{ner} = 0{,}28\,\text{Nm}$. Friktionskoefficienterna μ_1 och μ_2 är baserade på friktionen mellan plast och stål [22].

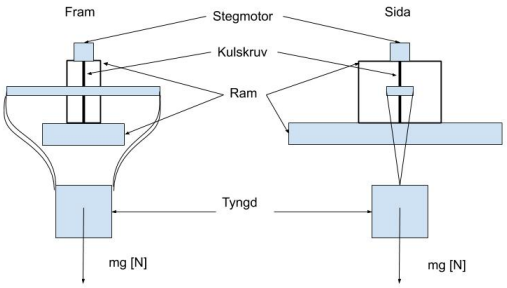
Tabell 3.3: *Värden på de variabler som använts i* (11) *och* (14)*, friktionskoefficienterna är hämtade från StructX [22]. Vikten är baserad på dåvarande CAD-modell.*

Variabel	**Uppskattat mätetal**
M	$10\,\text{kg}$
g	$9{,}82\,\text{kgm/s}^2$
μ_1	0,3
μ_2	0,3
r	$0{,}02\,\text{m}$
R	$0{,}04\,\text{m}$
m	$0{,}5\,\text{kg}$
θ	$17{,}66\,^\circ$
ϕ	$14{,}5\,^\circ$

Utifrån detta resultatet beställdes stegmotorn QSH-4218-35-10-027 med hållmoment på $0{,}35\,\text{Nm}$ [23]. Att endast en valdes att köpas in först trots att fyra krävdes var för att utföra tester och på så sätt inte gå utanför budgeten. För att den skulle vara applicerbar krävdes det att den i kombination med kulskruven klarade att trycka upp halva robotens vikt. Det krävdes för att gångstilarna i Figurerna 2.7a och 2.7b skulle vara applicerbara. För detta utformades ett test enligt Figur 3.6, ett detaljerat kretsschema kan ses i Bilaga A. Med hjälp av uppställningen testades det sedan att motorn klarade av att lyfta halva robotens vikt genom att lasta hinken med tyngder.



(a) *Bild tagen på uppställningen.*



(b) *Grafisk beskrivning av uppställningen.*

Figur 3.6: *Uppställning av test för linjär aktuator.*

För testet av stegmotornerna som används i kombination med kulskruv klarade de av att lyfta halva robotens vikt enligt dåvarande CAD-modell, det vill säga 5 kg vid 3 rpm enligt (7) med 0,0037 s.

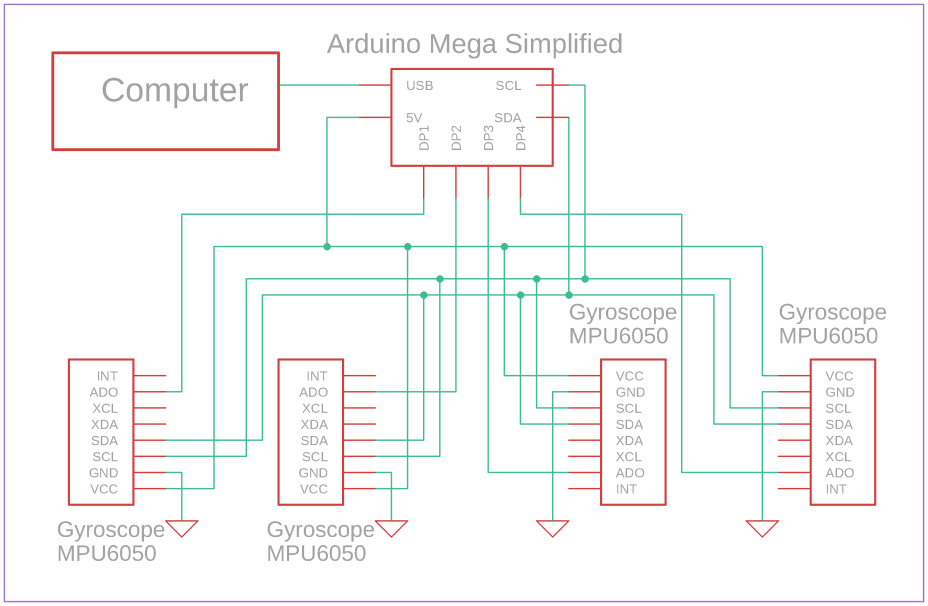
3.3.3 Val av drivkort

För att driva stegmotorn till den linjära aktuatorn krävdes ett drivkort. Först beställdes motordrivaren EVALSP820-XS som enligt dess datablad skulle klara av kontinuerlig ström på 1,5 A [24]. Jämfört med stegmotorn QSH-4218 -35-10-027 som hade rekommenderade spolström på 1,0 A så borde detta drivkort klarat av att driva motorn [23]. Men om motorn tappade steg vid testet för den linjära aktuatorn,det vill säga när motorn inte klarade av att utföra förändring till en ny position, gick drivkortet sönder och blev därmed ej brukbart. Detta testades på två drivkort av typ EVALSP820-XS innan slutsatsen drogs att de inte klarade av att ge motorn den effekt som krävdes för att lyfta halva robotens vikt samt att de inte klarade av att tappa steg.

Att stegmotordrivarna gick sönder kan bero på flera olika själ, varav några beskrivs i ett dokument [25]. Det som troligtvis förstörde motordrivarna var att när motorn inte orkade ta ett steg uppstod det en hög spänningsspik som skadade drivkortet. Dokumentet redovisar även vad som bör undvikas för att inte skada drivkortet. Detta var bland annat att inte skruva på motorn med drivkortet inkopplat på grund av att detta kan skapa ström som matas in i drivkortet. Drivkort kan även ta skada om det används med alltför hög frekvens, om man drar ur eller sätter i sladdar när strömmen är på och om drivkortet blir för varmt. För att kringgå dessa problem valdes det att använda drivkorten TB6600 (samma som för axelns stegmotor) [26]. Denna hade visat sig inte vara känslig för dessa problem vid testet av axeln stegmotor Kapitel 3.2. Därför valdes det att använda även TB6600 för den linjära aktuatorn.

3.4 Accelerometer

För att mäta de vinklar som benen har i förhållande till marken och varandra används fyra stycken accelerometrar/gyroskop. Från förra årets projekt ärvdes två stycken GY-521 [27] som baseras på chippet MPU6050 [28]. Denna typ av sensor kan mäta acceleration, vinkelhastighet och temperatur[29]. GY-521 använder sig utav I2C för kommunikation men har endast två olika adresser som det går att läsa från 0x68 och 0x69, vilken adress som sensorn befinner sig på bestäms av en adresspinne, AD0. För att komma runt begränsningen med för få I2C-adresser kopplades adresspinnen(AD0) från de fyra sensorerna till varsin I/O pinne på Arduinon. När det var dags för en sensor att bli avläst dras adresspinnen hög för att skifta adressen från 0x68 till 0x69, Arduinon läser sedan endast av från adress 0x69. I Figur 3.7 visas hur accelerometrarna kopplades in i Arduino Mega.



Figur 3.7: *Uppkoppling av fyra GY-521 till Arduino Mega.*

Vinkeln räknades ut genom att accelerometerdatan avlästes och genom att använda trigonometri

$$\theta = tan^{-1}\left(\frac{x_{acc}}{y_{acc}}\right)\frac{180}{\pi}. \tag{15}$$

x_{acc} och y_{acc} är accelerationen uppmätt i respektive led. Vinkeln som fås är accelerometerns vinkel i förhållande till jordens gravitationsriktning.

3.5 Batteri

Vid dimensionering av batterier behövdes den maximala effekten som lasten drar undersökas. Motorerna i roboten ansågs vara det som drog störst effekt, därför försummades övriga komponenter vid beräkning och endast motorerna användes för dimensionering av batterierna. Enligt Kapitel 2.4 finns det som mest två stegmotorer i varje ben. Båda av modell Nema 17 men olika egenskaper [18], [23]. Utifrån dess datablad kan det avläsas att motorerna har en spolresistans på 1,7 Ω för motorn i axeln och 4,3 Ω för den linjära aktuatorns motor. Den maximala effekten per motor kan då räknas ut genom

$$P_{motor,max} = I_m \cdot U_v = \frac{U_v^2}{R_s}. \tag{16}$$

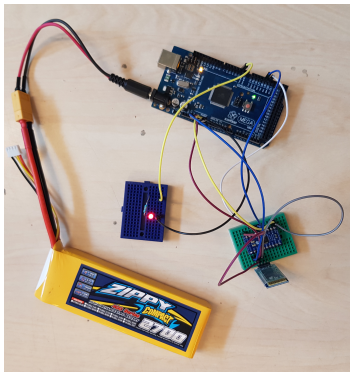
Där U_v är batteriets spänning, R_s motorn spolresistans och I_m strömmen genom motorn. Att motorn senare skulle komma styras genom att pulsmodulera spänningen (PWM-signaler) resulterar detta i att strömmen även den inte hålls konstant vilket dessa beräkningar antar. Vidare leder det till att effekten in i motorn varieras. Men eftersom detta endast var en uppskattning för maximal effekt antogs det att strömmen var konstant genom motorerna. Från ett av förra årets kandidatarbeten kunde erhölls fyra 11,1 V batterier [3], [30]. För att bedöma om dessa kunde användas beräknades först den totala effektutvecklingen per ben genom dess spänning, de båda motorernas tidigare nämnda spolresistanser och (16) kunde lastens maxeffekt per ben räknas ut till 101,1 W. Genom den maximala effekt och spänningen kunde (16) åter användas och den maximala strömmen som kan krävas till 9,1 A.

För att veta om batteriet kan leverera denna effekt läses dess C-klassificiering av, 40 C och lagringskapacitet 2700 mAh [30], [31]. En C-klassificiering på 1 C innebär att ett batteri med 1 Ah kan laddas ur med 1 A under 1 h. En C-klassificiering på 2 C innebär då istället att den kan laddas ur med 2 A under 30 min. Med klassificiering på 40 C och 2700 mAh kan dessa batterier laddas ur med 40 A, vilket skulle ta 4 min. Med den klassificeringen skulle de givna batterierna klara av att driva var ben. Med antagandet att den maximala strömmen 9,1 A skulle dras kontinuerligt hade den totala batteritiden istället vart 18 min. Det bör dock sägas att dessa beräkningar är med antagande att batteriets urladdning sker idealt och konstant vilket inte är fallet enligt [31]. Att använda ett batteri i varje ben förlänger inte endast batteritiden utan gör även roboten mer symmetrisk och minskar kabeldragnings arbetet.

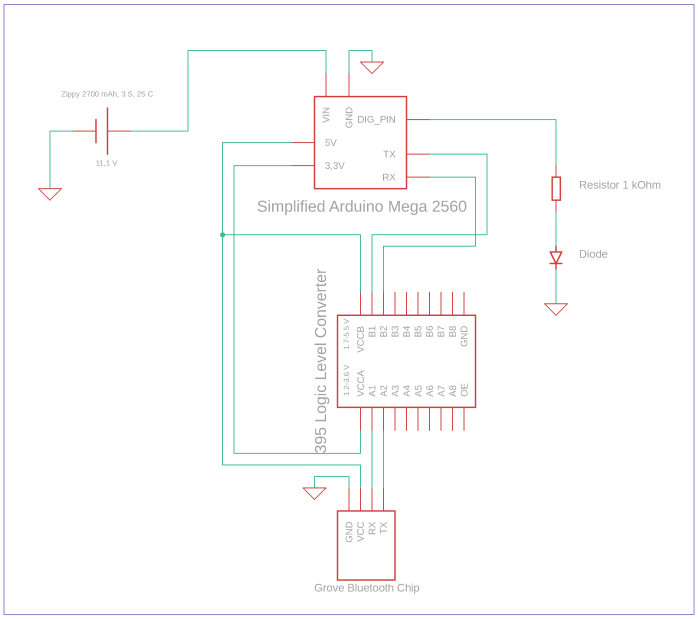
3.6 Blåtandskommunikation

För att testa om blåtandskommunikationen med det valda blåtandschipet, HC-06, fungerade designades en krets enligt Figur 3.8. Kommunikationen mellan HC-06 och Arduinon sker med seriell kommunikation. Arduino Mega har flera stycken seriella portar att välja bland men port 0 är samma som USB till Arduino använder, därför väljs serieport 1 att användas. En LED-lampa tillsammans

med applikationen *Arduino bluetooth controller* användes i detta testet för att tydligt se om signaler skickades och togs emot från applikationen på mobiltelefonen på ett korrekt sätt Figur 3.8a.



(a) *Bild av uppställning för test av blåtand-kommunikation.*



(b) *Kretsschema av uppställningen.*

Figur 3.8: *Uppställning för test av blåtand-kommunikation.*

4

Design och konstruktion

Detta kapitel beskriver konstruktionen och designen som grundar sig i dimensionerna och de valda komponenterna från Kapitel 2.4. Även fysiska och digitala tester som utförts på komponenterna och CAD-modellen kommer att redovisas.

4.1 Modellering

För att utvärdera olika möjligheter att konstruera roboten skapades en CAD-modell med Autodesk Inventor 2019. Modellen skapades för att kunna upptäcka problem, undersöka viktfördelningen och utrymme samt för en ökad förståelse av systemet. Modellen användes även vid simulering i MATLAB Simulink för att skapa en så realistisk simulering som möjligt. Utifrån CAD-modellen skapades skalet, fästen och fötterna för att sedan kunna monteras.

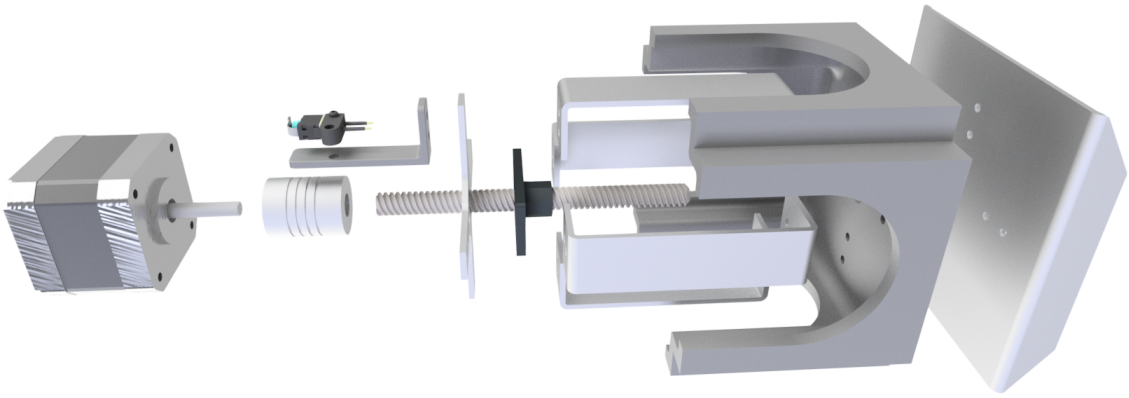
Alla komponenter modellerades för att kunna ge en så exakt modell som möjligt. Modellerna för Arduino Mega, motordrivkortet, spänningsomvandlare och ändlägesgivaren hämtades från tidigare gjorda CAD-modeller [32]–[35]. Arduino Mega och motordrivkortet fick modifieras då dimensionerna inte överensstämde med den verkliga produkten och ändlägesgivaren saknade detaljer som fick tilläggas. Utifrån komponenter från tidigare arbeten och komponenter som fanns tillgängliga på CRF skapades modellerna för axlarnas stegmotorer, batterierna, blåtands-modulen HC-06, accelerometerna, säkringarna och spänningsomvandlare [3], [4]. Dessa tillsammans med resterande komponenter skapade från tillgängliga ritningar kan ses i Bilaga E.



Figur 4.1: *De modellerade komponenterna som användes i modellen. Arduino mega, motordrivkort, buck konverter och ändlägesgivare. Från [32]–[35]. Omarbetade med tillstånd.*

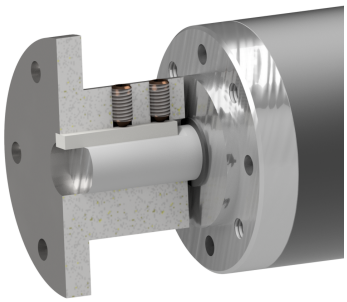
Varje ben designades som ett långt block bestående av fyra sidor, en överdel och en fot. Sidorna skapades med aluminiumplåt och böjdes för att skapa kontaktytor mellan de olika sidorna. Delarna designades med utskurna hål för att minska den totala vikten, så kallade lätthål. Benen designades för att vara så lika som möjligt, detta för att skapa en jämn viktfördelning. Foten som var ett system i sig skapades av bockat aluminium och en 3D-skriven form för att hålla foten rak när den sträcks ut och hindras från att roteras. Formen agerade även som ett fäste för att koppla ihop foten och kulskruven samt stegmotorn med kulskruven, se Figur 4.2. Foten fästes med benet genom att montera fast kulskruvsmotorn på en platå som var monterad på insidan av varje ben, se Bilaga E.

Kulskruven som skapade den linjära rörelsen var fäst på en anordning som i sin tur var fäst på foten. För att kunna identifiera var fotens position i z-led befinner sig monterades även en ändlägesgivare på foten. Figur 4.2 visar fotens uppbyggnad och kopplingen mellan stegmotor och kulskruv.

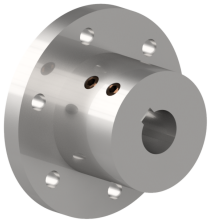


Figur 4.2: *Sprängskiss av foten.*

För att fästa motorn till det intilliggande benet skapades ett axelfäste som designades för att få ett så litet mellanrum som möjligt mellan benen. Axelfästet valdes att tillverkas av ett starkt material som klarar av höga påfrestningar. Ett problem som uppmärksammats i tidigare projekt med liknande motorer var att skruvarna i axelfästet skruvar loss sig själva [4]. För att förhindra detta har en kil lagts in som lägger ett jämnt tryck över hela motorns axel istället för två punkter som skruvarna endast ger, se Figur 4.3a. Den slutgiltiga modellen av axeln kan ses i Figur 4.3b.



(a) *En genomskärning av axeln, observera kilen som trycks fast av de två skruvarna mot motorns axel.*



(b) *Modell av axelfäste.*

Figur 4.3: *Överblick och genomskärning av axelfästet*

Två delar av CAD-modellen bedömdes potentiellt kunna försvagas på grund av påfrestningar från att hålla uppe delar av roboten, därför var det viktigt att analysera dessa. Den ena var fästet i foten där kulskruven monterades, denna del behövde klara av att hålla uppe halva robotens vikt. Den andra var motorfästet vid axeln som behövde klara av att hålla uppe minst halva motorns vikt utan att motorfästet vinklades. Dessa delarna analyserades i Autodesk Inventors Stress Analysis Environment för att upptäcka eventuella kritiska områden.

I samband med designen i Autodesk Inventor modellerades hållare till batterier och sensorer som skulle användas, se Figur 4.4a. Även delar för att omvandla servons rotation till en linjärrörelse i syfte att kunna låsa benen modellerades, se Figur 4.4b. Utöver detta modellerades sensorhållarna, dessa valdes att göras som ett skelett för använda så lite plast som möjligt samt för en snabb utskrift.

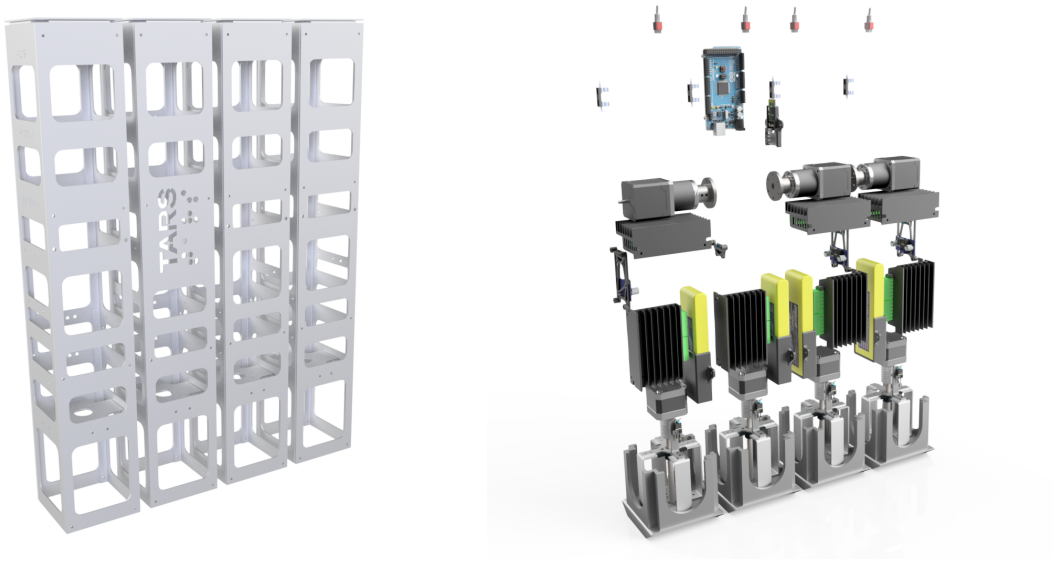


(a) *Hållare till sensorer och batterier.*

(b) *Anordning till servot för att kunna låsa fast benen.*

Figur 4.4: *Modellerade delar som har 3D-utskrivits.*

Vidare har CAD-modellen använts som ett verktyg för beräkning av masscentrum och testa olika placeringsmöjligheter för komponenter. Det alternativ med den lägsta kritiska vinkeln valdes och därefter skapades lätthålen, detta för att alla komponenter skulle ha en fästyta. Många komponenter byttes ut eller togs bort vilket gjorde att detta steg upprepades ett flertal gånger. Chassit kan ses i Figur 4.5a, de olika komponenterna och sensorhållarnas placering i de olika benen kan ses i Figur 4.5b.

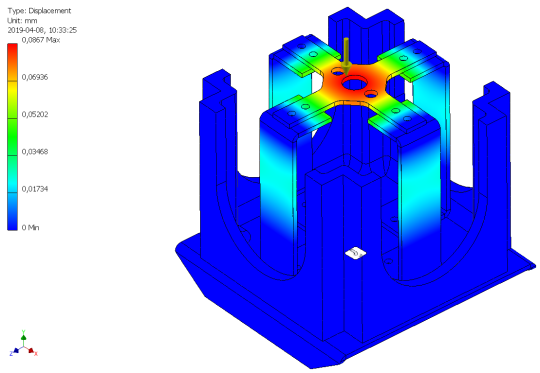


(a) *Robotens chassi.* **(b)** *Överblick över insidan.*

Figur 4.5: *Skal och innanmäte av den slutgiltiga CAD-modellen*

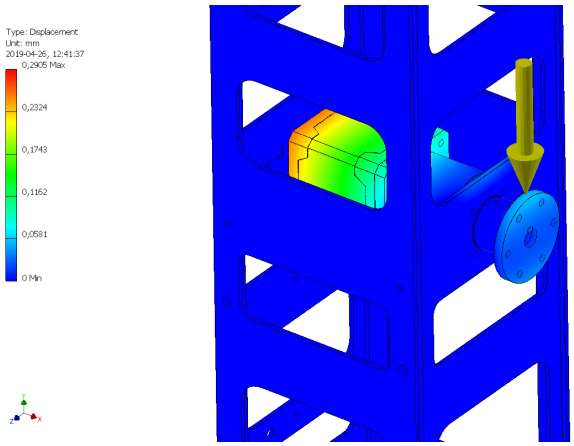
4.2 Stressanalys

Som tidigare nämnts undersöktes spänningarna på foten och axelfästet i Autodesk Inventors Stress Analysis Environment. Detta gjordes genom att en kraft på 51 N, representerande dåvarande CAD modellens vikt, lades på kulskruvsfästet och analyseringen gav resultatet att hållaren deformerades som mest i centrum med cirka 0,087 mm. Sidorna till fästet böjdes utåt med cirka 0,018 mm, se Figur 4.6. Dessa deformationer ansågs vara försumbara i detta fall och kommer därför att kunna hålla roboten uppe utan komplikationer.



Figur 4.6: *Deformering med en pålagd kraft på* 51 N *tryckandes på kulskruvsfästet.*

För nästa analys lades en kraft på 98,1 N längst ut på axelfästet och analysen gav resultatet att motorfästet deformerades med cirka 0,06 mm i ytterkanten, se Figur 4.7. 91,8 N lades på för att kompensera för vridmoment och det är den högsta radiella kraften motorns axel klarar av [18]. Denna deformation ansågs vara försumbar i detta fall och skulle därför inte påverka robotens funktion.



Figur 4.7: *Deformering med 98,1N tryckandes på axeln.*

4.3 Tillverkning och montering

Inför konstruktionen skapades ett detaljerat kretsschema för roboten, se Bilaga A. Genom att studera datablad för varje komponent, läsa relevanta forumstrådar, erfarenhet från testerna gjorda i Kapitel 3 samt utföra enklare tester kunde det bestämmas hur kopplingen skulle ske [19], [23], [26], [27], [33], [36]–[38].

Vidare påbörjades konstruktionen med att de fyra sidorna, platån till nedre stegmotorn och överdelen på var ben skars ut med en vattenskärare. Efter detta handbockades platåns fram- och baksida. Vattenskärningen och bockningen skedde enligt ritningar skapade från CAD-modellen, se Bilaga E. Platån och överdelen bockades samt borrades i efterhand för att passa till sidorna på grund av bockningens oprecisa resultat. Axelfästena svarvades ut, borrades, gängades och spaltar skapades med hjälp av ett sågblad och en hydraulisk press. Sensorfästen, batterihållare och anordningen till servomotorn 3D-skrevs ut i PLA. Robotens skal sammanlänkades sedan med skruvar i kombination med låsmuttrar. Vidare monterades alla komponenter och kabeldragningen gjordes enligt kretsschemat som skapats.

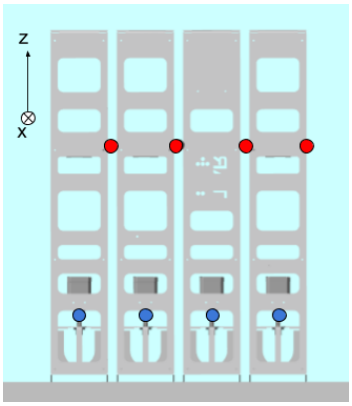
5

Simulering och styrning

Detta kapitel behandlar simuleringen som utförts för att testa konceptet. Styrningen som använts, dess tillhörande program för att styra roboten och hur den konstruerades kommer även att behandlas nedan.

5.1 Simulering

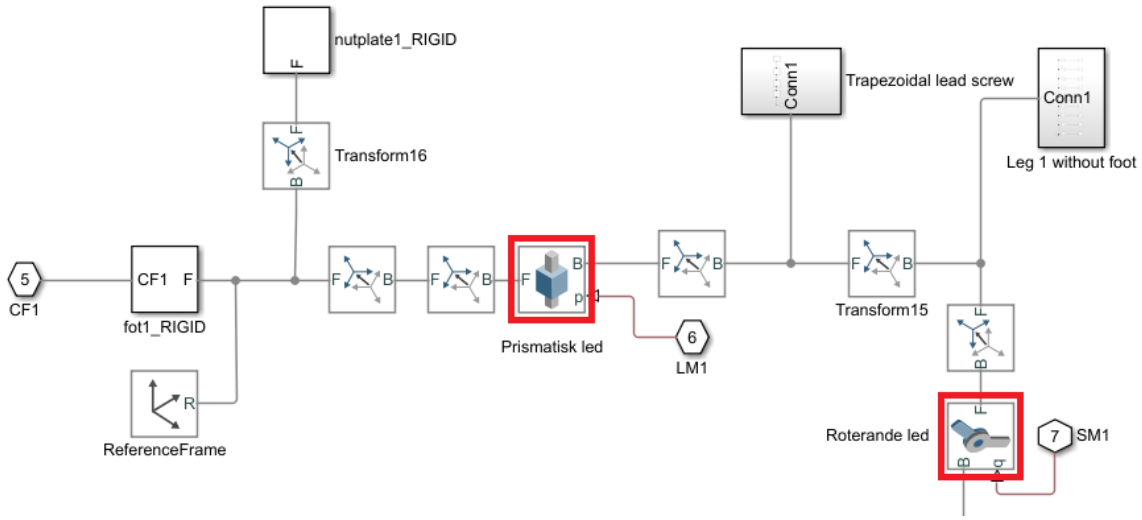
För att enkelt testa konceptets olika rörelser och dess begränsningar simulerades det. Detta gjordes med programmet Simulink i MATLAB. Modellen som gjordes i Autodesk Inventor, se Kapitel 4.1, exporterades till Simulink med hjälp av Simscape Multibody Link plug-in, simulerings modellen kan ses i Figur 5.1. Med denna modell skapades först miljön som roboten kunde röra sig i, det vill säga en plan yta att utföra tester på. För att få roboten att stå upp och interagera med marken användes Simulinks Contact Forces Library [39]. Detta gjordes även på grund av att roboten kan liknas vid en dubbelpendel som rör sig i ett oregelbundet rörelsemönster. Det innebar att den till synes sporadiskt studsade mot den plana ytan, vilket krävde en impulsbaserad kontaktmodellering som biblioteket erbjuder.



Figur 5.1: *Röda cirklar visar positionen av de roterande blocken och blåa cirklarna visar vart den linjära rörelseaktuatorn ansattes.*

Kontaktpunkter lades under simuleringsmodellens fötter vilket möjliggjorde för modellen att komma vara i kontakt med den plana ytan. Efter korrekt modellering av de kontaktkrafter som gjordes

med ytan, normalkrafter och friktionskrafter, bestod följande procedur av att ge rörelse till robotens leder runtom robotens axlar. För detta anbringades roterande leder mellan benen. Efter att axelns rotationsrörelse hade implementerats, skapades fyra ytterligare leder mellan fotens hållare och ledskruvarna. För detta användes prismatiska leder vilket möjliggjorde en linjär rörelse vinkelrätt mot den nedre ytan på stegmotorerna som driver ledskruvarna, se Figur 5.2 för simuleringsblocken, där lederna är fetmarkerade i rött. I Figur 5.1 visas de roterande lederna i rött och de prismatiska lederna i blått. Med kontaktmodellen och lederna inställda tilläts korrekt rörelse, var nästa steg att simulera avvikelser i robotens masspunkt under rörelse.



Figur 5.2: *Del av simuleringsmodellens uppbyggnad i MATLAB Simulink. Fet markering visar vart prismatiska och roterande led per ben placerats.*

För att beräkna masspunktens rörelse under robotens gång användes Simulinks General Mass Variable Block [40]. Den sätter en masspunkt med angiven vikt vid önskad position och ger ytterligare funktionalitet genom att välja en tidsberoende tyngdpunkt, som tar in en vektor i tre dimensioner som motsvarar förflyttningen. Utifrån rörelsen som de olika lederna tillgivits kunde masspunktens förflyttning, radianer per sekund för de roterade lederna i axlarna mellan benen och millimeter per sekund för de prismatiska lederna i fötterna beräknas, samtliga i realtid. Masspunkten ansattes vid axeln kring mitten av roboten och när den stod i startposition fanns det endast en förflyttning i z-led. Genom att använda längden mellan robotens rotationsaxel och masscentrum för varje ben som togs fram via modellering i CAD för att sedan ansättas i simuleringsmodellen, se Kapitel 4.1, kunde masscentrumets förflyttning beräknas i simuleringen genom att multiplicera alla längder med dess korresponderande trigonometriska funktion, där α står för vinkeln som benet har skjutits ut med.

$$\begin{aligned} \mathbf{Z} &= \sum_{n=1}^{4} L_n \cos(\alpha), \\ \mathbf{X} &= \sum_{n=1}^{4} L_n \sin(\alpha). \end{aligned} \tag{17}$$

När roboten stod i grundläge skedde endast en förflyttning i z-led då axelvinkeln var noll vilket gav ett sinus-värde på noll. Denna ekvation gav två värden per ben vilket sedan summerade alla cosinus- och sinusvärden för sig då de motsvarar förflyttning i sin respektive dimension. Dessa summor divideras slutligen med för att ge ett medelvärde av ett bens massa. De användes sedan som indata i förflyttningsvektorn. För att modellera fötternas påverkan av varje bens masspunkt vid utskjutning användes ritningarna som togs fram med Autodesk Inventor då foten var utskjuten med 30 mm, se bilaga E. Genom att dividera förflyttningen av masscentrumet Δh med den nämnda längden ΔL fås konstanten K_n som motsvarar förflyttning av tyngdpunkten per millimeter, följande ekvationer fås

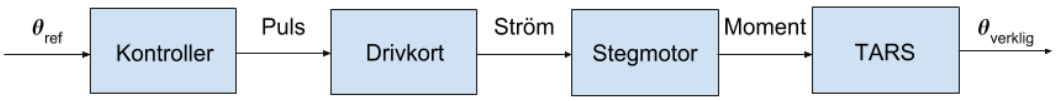
$$\begin{aligned} \mathbf{Z} &= \sum_{n=1}^{4} K_n \cdot \Delta h \cdot \cos(\alpha) \\ \mathbf{X} &= \sum_{n=1}^{4} K_n \cdot \Delta h \cdot \sin(\alpha). \end{aligned} \tag{18}$$

5.2 Styrning

Syftet med styrningen var att få robotens komponenter att samverka och på så sätt möjliggöra de olika gångstilarna. Styrningen har baserats på de valda stegmotorerna och modellen skapad i MATLAB Simulink har använts för att testa styrningen och säkerställa dess funktionalitet före implementation. Förutom metoden för styrningen kommer även en beskrivning av det konstruerade programmet att presenteras i detta kapitlet.

5.2.1 Metod för styrning av roboten

Stegmotorn saknar återkoppling och kan därför inte detektera fel som uppkommer vid störningar måste en kontroller och dekoder användas om ett återkopplat system ska skapas [41]. Istället kan en öppen styrning användas som även är lättare att implementera. Ett öppet system har därför konstruerats enligt Figur 5.3.



Figur 5.3: *Blockschema för systemets styrning som sker med hjälp av ett öppet system.*

Då antalet steg motorn tar per varv var känt till 200 skapades ett program där en referensvinkel matades in för att få ett specifikt bens axel att rotera till den givna vinkeln [18], [23]. Genom

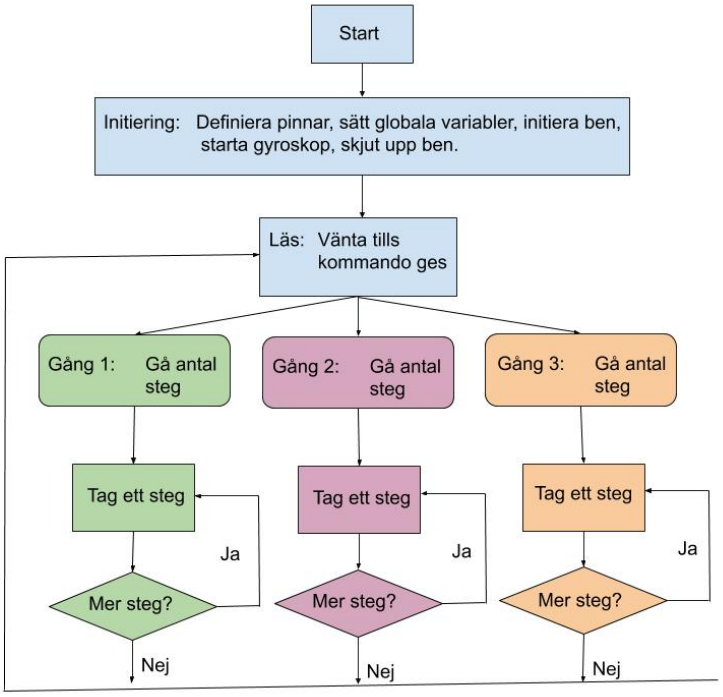
tidigare nämnd simulering i Kapitel 5.1 tillverkades övriga nödvändiga styrsignaler. Programmet fungerade som så att gångstilen och antalet steg som önskades matades in i en blåtandsapplikation som sedan överförde signalerna till roboten. Allt annat fördefinierade och de interna styrsignaler togs fram genom en iterativ optimeringsprocess som utgick ifrån de beräknade referensvinklarna och simuleringen.

5.2.2 Programmering

För att roboten skulle kunna utföra önskade rörelser och samverka med blåtandskommunikationen, motorerna samt sensorerna programmerades den. Detta gjordes i Arduinos egna programmeringsspråk vilket är en blandning av C och C++ [42].

Koden delades upp i tre större delar: kommunikation, styrning av motorer och avläsning av sensorer. Applikationen Arduino bluetooth controller användes för att kommunicera med roboten, där insignaler skickades in genom applikationens terminal [43]. Programmet designades för att skriva meddelanden på applikationens terminal som bland annat bad användaren skriva in kommandon. Detta kommando lästes sedan av och en skapad funktion för det givna kommandot tillkallades. Roboten styrde hur fötterna förflyttade sig och hur benen roterade i en fördefinierad loop som kördes lika många gånger som de valda antal stegen.

Figur 5.4 visar ett flödesschema över programmet som skapats. Programmet startade upp och började med att initiera mikrodatorns pinnar för olika signaler enligt Bilaga B. Därefter avvaktade programmet till dess att ett kommando gavs gällande om gångstil 1, 2 eller 3 skulle utföras samt antalet steg. Därefter loopades programmet tills givet antal steg var utfört. Därefter väntade det återigen på ett nytt kommando.



Figur 5.4: *Flödesschema över programmet som använts för att styra modellen av TARS.*

I den skrivna koden definierades en egen klass som kallades Leg och initierades med de I/O-pinnarna som kopplades till korresponderande bens motorer och sensor. Den fullständiga listan med variabler som användes vid initieringen finns i Tabell 5.1.

Tabell 5.1: *Variablerna som användes för att skapa en instans av klassen Leg och deras betydelse.*

Inparameter	Innebörd
PUL_rot	För att motorn ska ta ett steg, reagerar på positiv flank.
DIR_rot	Sätter riktningen på motorn.
ENA_rot	Sätter på motorn.
PUL_lead	För att motorn ska ta ett steg, reagerar på positiv flank.
DIR_lead	Sätter riktning på motorn.
ENA_lead	Sätter på motorn.
endSwitch	Ändlägesgivare, låg när ledskruven nått sitt slut.
microsec_screw	Tiden mellan två steg för ledskruvsmotorn.
microsec_axle	Tiden mellan två steg för axelmotorn.

Värt att notera är att i ben 2 fanns ingen motor i axeln, initieringen skedde då genom att använda en I/O-pinne som inte användes av övriga komponenter. Klassen Leg innehöll grundläggande funktioner

för benet som gjorde det enkelt att implementera olika gångstilar, ett visas i Tabell 5.2, två andra gångstilar finns i Bilaga D.

Tabell 5.2: *Beskriver ordningen som de olika funktionerna anropas för gångstil 1.*

Gångstil 1		
Instruktion	**Anrop**	**Förändring**
1	Leg1.pushOut();	4 mm
	Leg4.pushOut();	4 mm
2	Leg1.rotateForward();	17 °
	Leg4.rotateBackward();	17 °
3	delay(1700);	
4	Leg1.pushOut();	18 mm
	Leg4.pushOut();	18 mm
5	Leg1.pushIn();	22 mm
	Leg4.pushIn();	22 mm
	Leg2.pushOut();	≈8 mm
	Leg3.pushOut();	≈8 mm
	Leg1.rotateBackward();	17 °
	Leg4.rotateForward();	17 °
6	Leg1.rotateBackward();	17 °
	Leg4.rotateForward();	17 °
7	delay(1700);	
8	Leg2.pushout();	18 mm
	Leg3.pushOut();	18 mm
9	Leg2.pushIn();	22 mm
	Leg3.pushIn();	22 mm
	Leg1.pushOut();	≈8 mm
	Leg4.pushOut();	≈8 mm
	Leg1.rotateForward();	17 °
	Leg4.rotateBackward();	17 °
10	Leg1.pushIn();	≈8 mm
	Leg4.pushIn();	≈8 mm

Den första gångstilen var implementerad enligt Tabell 5.2. Anrop inom samma instruktion anropades samtidigt och *delay();* anropades för att tillåta roboten sluta gunga efter att den fallit över. För att använda de olika anropen flera gånger och på så sätt bestämma hur mycket som antingen de linjära aktuatorerna eller vinkeln mellan axlarna skulle förändras gjordes detta som en loop. Trots att det i Kapitel 3.3.2 beräknades att foten endast skulle behöva lyftas knappt 1,5 mm över marken lyfter modellen upp foten 4 mm för att minska risken att till exempel en ojämnhet skulle fälla modellen när den tar ett steg. De funktioner som användes vid gångstil 1 i Tabell 5.2 beskrivs i Tabell 5.3.

Tabell 5.3: *De olika funktonerna som finns för klassen Leg.*

Funktion	Resultat
pushOut()	Trycker ut benet ett motorsteg motsvarande $\approx 0{,}042\,\text{mm}$
pushIn()	Drar in benet ett motorsteg motsvarande $\approx 0{,}042\,\text{mm}$
rotateForward()	Roterar axelns led $1{,}8\,^\circ$
rotateBackward()	Roterar axelns led $-1{,}8\,^\circ$
resetLegLocation()	Drar in alla ben tills det att ändlägesgivaren blir låg

De implementerade gångstilarna använde inte sensorerna som installerats. Detta berodde på att när sensorerna och motorerna initierades samtidigt klarade programmet inte av att köras. När sensorerna kördes utan att motorerna var inkopplade kunde de mäta vinklar som verkade skilja sig $\approx \pm 1\,^\circ$ från verkligheten. En möjlig anledning till att sensorerna och motorerna inte fungerade tillsammans var att för stor ström drogs från Arduinon. Enligt databladet [44] kan motordrivaren TB6600 kräva upp till 15 mA. Detta skulle innebära att den totala strömmen som alla drivkort tillsammans maximalt kunde kräva var

$$I_{tot} = 15 \cdot 7 \cdot 3 = 315\,\text{mA}. \tag{19}$$

Där 7 är antalet drivkort och 3 antalet sladdar som kommunicerar med varje drivkort. Detta är över de 200 mA som Arduino Mega 2560s processor ATmega2560 kan leverera [45]. Då det uträknade värdet från (19) är ett extremfall är det inte troligt att 315 mA någon gång skulle krävts. Dock nämner även [44] att drivkortet minst drar 8 mA vid drift, det skulle innebära att med samma uträkning som tidigare att 168 mA drogs kontinuerligt. Vilket bedömdes vara väldigt nära gränsen på 200 mA. Under kontinuerlig drift var det mest troligt att motordrivarna drog någonstans mittemellan.

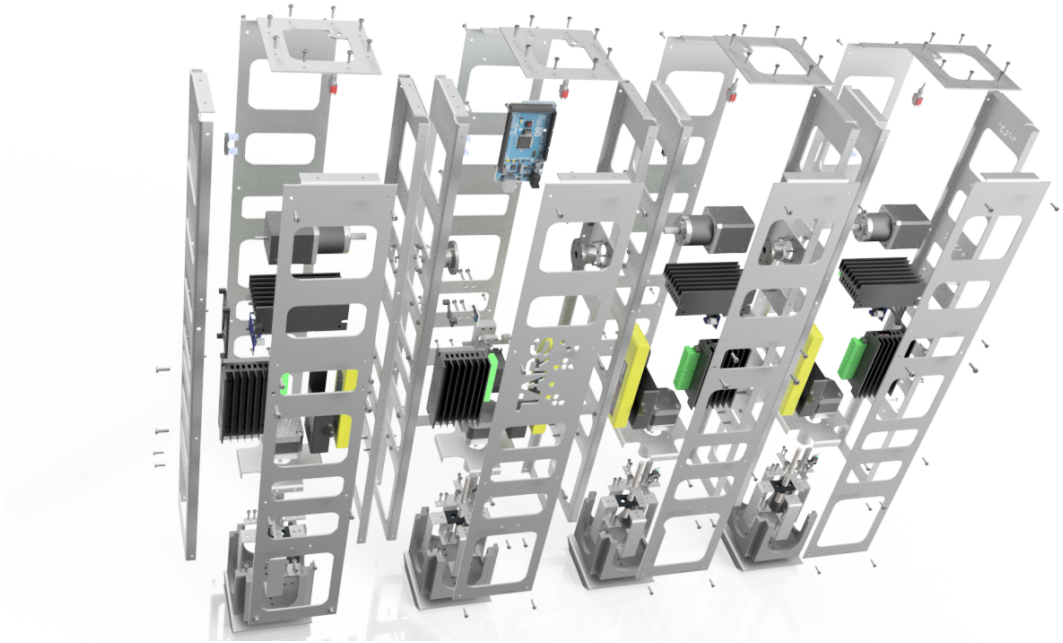
6

Resultat

Projektet resulterade i en robot som kunde gå med varierande kvalité på två olika sätt och styras trådlöst med hjälp av blåtand via applikationen Arduino bluetooth controller [43]. Projektet höll sig inom given budget på 5000 sek, se Bilaga C. Nedan presenteras det slutgiltiga resultatet av projektet i form av modell, simulering och styrning.

6.1 Modell

Den slutgiltiga placeringen av komponenterna och hur det var tänkt att roboten skulle assembleras kan avläsas i Figur 6.1 och Figur 6.2a. Vikten på hela modellen och varje enskilt ben kan finnas i Tabell 6.1.



Figur 6.1: *Sprängskiss av den slutgiltiga modellen.*

Tabell 6.1: *CAD-modellens totala vikt respektive robotens totala vikt fördelad på de fyra benen.*

Ben	Vikt CAD-modell	Vikt fysisk modell
1	2,82 kg	2,66 kg
2	2,39 kg	2,35 kg
3	2,90 kg	2,78 kg
4	2,81 kg	2,70 kg
Totalt	10,91 kg	10,48 kg

Från modellen kan det ses att tre ben var liknande, ben 1 var spegelvänt jämfört med ben 3 och 4. Ben 3 hade även ett axelfäste utöver de andra komponenterna, se Figur 6.1. Ben 2 skiljer sig från resterande då det saknar en motor i axeln men även eftersom den innehåller en mikrodator och en servomotor. Denna modell gav det bästa masscentrumet, det vill säga det masscentrum som gav lägsta möjliga kritiska vinkel.



(a) *Slutgiltiga CAD-modellen av TARS.* **(b)** *Fysiska modellen.*

Figur 6.2: *Jämförelse mellan CAD-modellen och den slutgiltiga fysiska modellen*

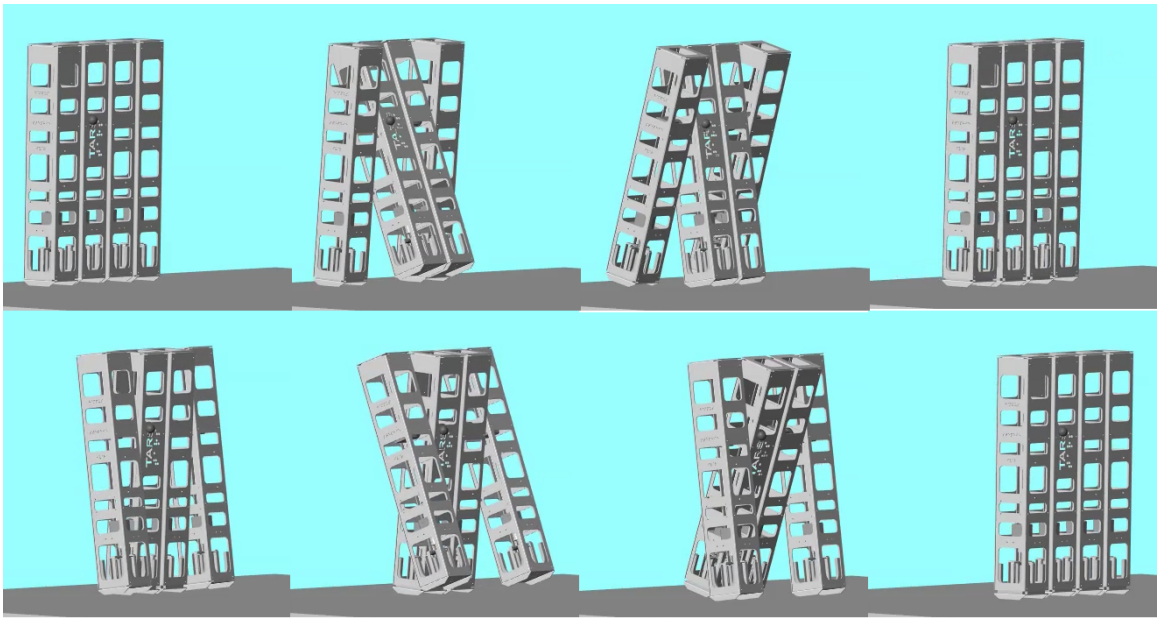
CAD-modellen och den färdigkonstruerade roboten utseende och dimensioner kan ses i Figur 6.2 och Tabell 6.2.

Tabell 6.2: *CAD-modellens respektive den fysiska-modellens dimensioner*

Dimension	Cad-Modell	Fysisk-modell
Höjd	61,70 cm	61,70 cm
Bredd	44,35 cm	45,70 cm
Djup	10,00 cm	10,00 cm

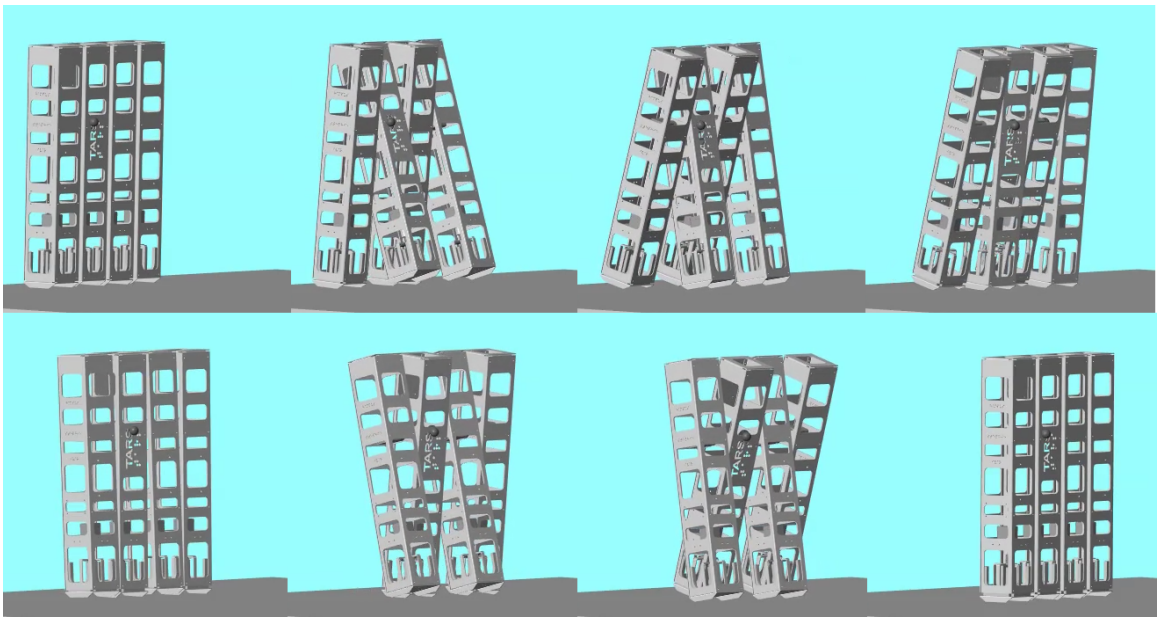
6.2 Simulering och styrning

Utifrån simuleringen erhölls det att gångstil 1 och 2 var enklare att realisera än gångstil 3. Rörelsemönstrets bana för gångstil 1 var rak medan för gångstil 2 var det en svängande rörelse i sidled då benen som skjutits ut drogs mot golvet vid inskjutning vilket ledde till en rotation kring masspunkten. Gångstil 1 kan ses i Figur 6.3 simuleringsmodellen uppvisade en kritisk vinkel på 25 ° då de mellersta benens fötter skjutits ut 4 mm.



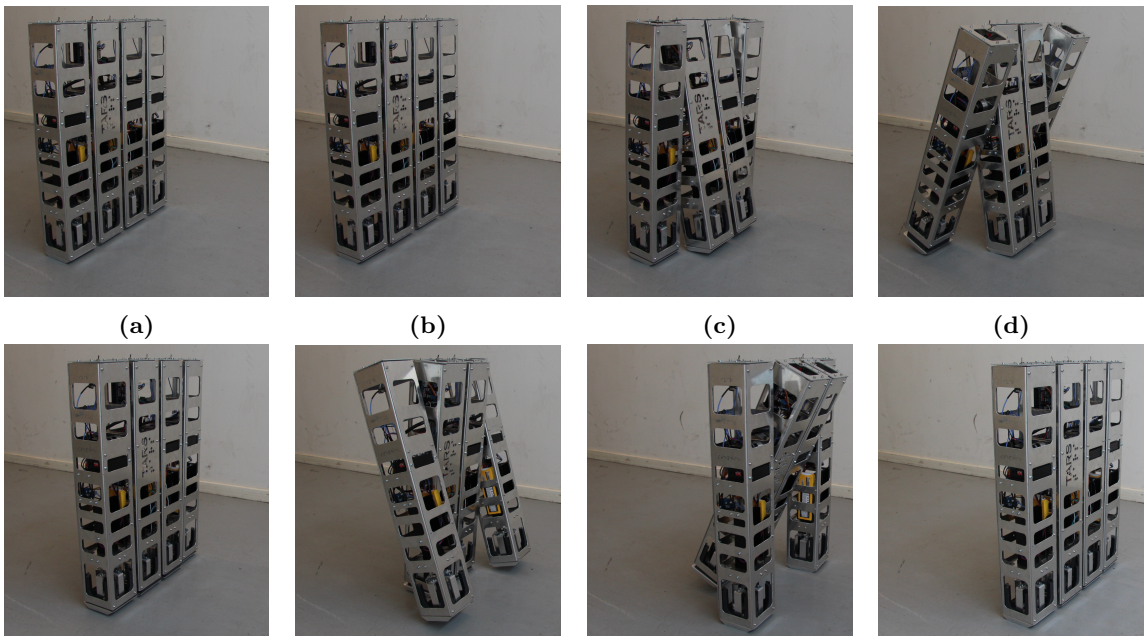
Figur 6.3: *Ett steg enligt gångstil 1 i simulationen.*

För gångstil 3 lyckades inte en fungerande simulering skapas. Svårigheten uppkom i gångstilens dynamik som gjorde det svårt för roboten att balansera. Simulering för gångstil 2 kan ses i Figur 6.4.



Figur 6.4: *Två steg enligt gångstil 2 i simulationen.*

För den fysiska modellen visas gångstil 1 i Figur 6.5 och gångstil 2 i Figur 6.6. Gångstil 1 kan delas in i två delar, första delen visas i Figurerna 6.5a - 6.5e där ben 2 och 3 roterades ut först med vinkeln 17 °. I andra delen som visas i Figurerna 6.5f - 6.5h roterades sedan 1 och 4 ut. Figur 6.5a var robotens initiala position, Figur 6.5b efter instruktion 1, se Tabell 5.2 där fötterna skjutits ut och i Figur 6.5c lyftes benen framåt enligt instruktion 2. Fördröjningen i instruktion 3 gav roboten tid att landa och stabilisera sig på golvet innan instruktion 4 sköt ut de bakre benen för att förflytta tyngdpunkten framåt, vilket visas i Figur 6.5d. I Figurerna 6.5e - 6.5h visas den andra delen vilket var densamma som första men där ben 1 och 4 istället roterades fram. Efter instruktion 6 visas slutpositionen för roboten i Figur 6.5h, vilket var densamma som startpositionen.



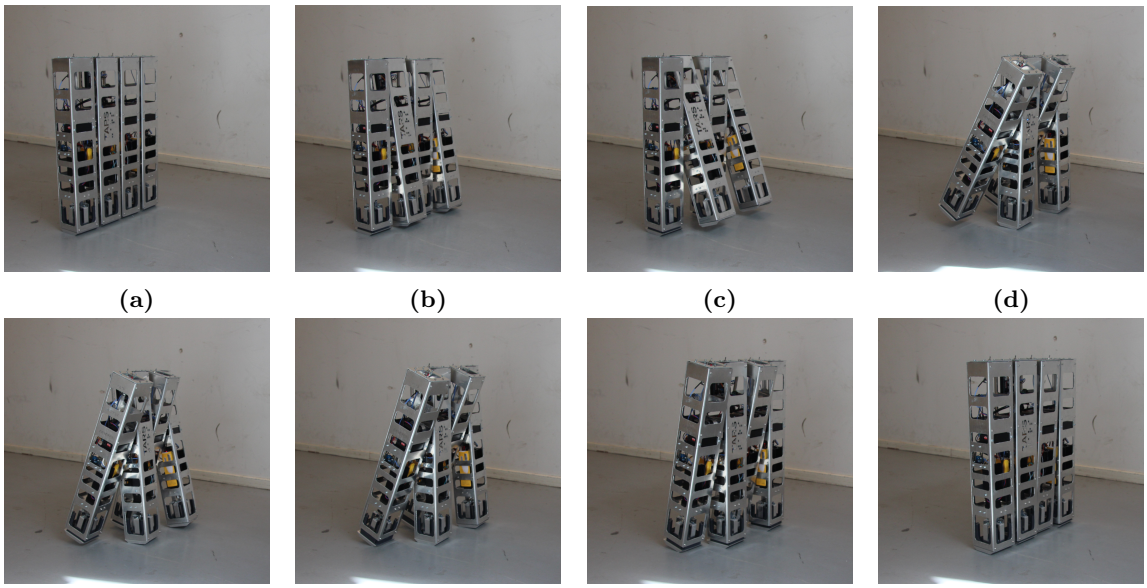
Figur 6.5: *Bildserie av ett steg enligt gångstil 1.*

Modellen av TARS tog ett steg på 35 cm enligt Figur 6.5 på 23 sekunder. För att avgöra om en gångstil var lyckad skickades kommandot att ta fem steg till roboten, sedan upprepades försöket tio gånger. Även då modellen hade ett fysiskt fel som åtgärdades fortsatte testerna för att kunna presentera ett sanningsenligt resultat vilket visas i Tabell 6.3. Gångstilen ansågs lyckad då majoriteten av försöken lyckades ta tre eller fler steg.

Tabell 6.3: *Resultat av testerna för gångstil 1.*

Försök	Resultat	Kommentar
1	0	Ben 3 lyftes inte upp
2	2	Ben 3 lyftes inte upp ordentligt
3	1	Ben 4 roterar upp för sent
4	1	Föll över mellan första halvan och andra halvan
Axeln mellan ben 3 och 4 var lös - skruvades åt		
5	5	Ben 4 hamnade lite efter
6	4	Allt gick fel på sista steget
7	4	Föll över
8	3	Föll över
9	5	Ben 3 lyfts inte ordentligt, vi vet hur vi kan svänga
10	5	Gick bra

I Figur 6.6 tar den fysiska modellen ett steg enligt gångstil 2. För denna gångstil roterades först ben 2 och 4 ut, vilket kan ses i Figurerna 6.6a till 6.6c. Därefter, likt gångstil 1 ges det tid för roboten att falla fram och stabiliseras innan den sköt fram sig på ben 2 och 4 till positionen visat i Figur 6.6d. Slutligen roterades ben 1 och 3 in och roboten avslutades i startposition, se Figur 6.6h. Nästa steg skedde sedan spegelvänt med ben 1 och 3 som roterades ut.



(a) **(b)** **(c)** **(d)**

(e) **(f)** **(g)** **(h)**

Figur 6.6: *Bildserie av ett steg enligt gångstil 2.*

Tester gjordes även för gångstil 2 och visas i Tabell 6.4. På grund av att de första sex försöken resulterade i färre än tre steg avbröts testet då detta ansågs vara ett misslyckande. Detta på grund av att ben 2 hamnar efter ben 4 då den tappar steg under gången. Ben 2 roteras av två stegmotorer och när dessa inte roteras exakt samtidigt skapar en motor en motkraft och stegmotorerna tappar steg. Detta resulterade i att benen blev osynkroniserade vilket ledde till att benet inte rörde sig fram till rätt vinkel för att ta emot fallet med foten.

Tabell 6.4: *Resultat av testerna för gångstil 2.*

Försök	Resultat	Kommentar
1	2	Bra första steg, sedan släpar ben 2 efter
2	2	Bra första steg, sedan släpar ben 2 efter
3	2	Bra första steg, sedan släpar ben 2 efter
4	1	Roboten faller över efter steg 1
5	2	Återgår till tidigare rörelse där ben 2 släpar efter
6	2	Bra första steg, sedan släpar ben 2 efter

Ett problem som uppkom när flera av robotens motorer kördes samtidigt var att tiden mellan stegen som togs blev annorlunda jämfört med den som uppmättes under testerna när endast en

motor testades åt gången. Koden för att motorerna ska vrida sig ett steg skrevs och optimerades i teststadiet för en motor med hjälp av den inbyggda delayMicros(x)-funktion, i Arduinos IDE. När denna koden sätts in i en loop med flera motorer blir inte tiden mellan stegen den optimerade utan

$$t_{mellansteg} \approx t_{optimerad} \cdot n, \tag{20}$$

där n är antalet motorer som körs samtidigt.

Det använda styrprogrammets fullständiga kod går att finna på `https://github.com/IvarSorqvist/TARS`.

7

Diskussion

Den slutgiltiga produkten uppfyllde projektets syfte, den klarade av att gå minst en gångstil, även om kvalitén på gångstilen varierade. Den kunde även styras med hjälp av blåtand. Dock har flertalet förändringar i de specifika lösningarna från projektets början skett. Trots det höll projektet sig inom den giva budgeten och tidsbegränsningarna som satts. Nedan följer en grundlig diskussion av projektets utförande, resultat och förbättringsmöjligheter.

7.1 Konstruktion

Konstruktionens största utmaning var att lyckas få masscentrumet till främre nedre delen av roboten. Den tyngsta delen, stegmotorn till axeln, var låst till den övre delen av benet för att efterlikna TARS, se Figur 2.6. De övriga tunga delarna var även de största vilket skapade komplikationer vid placeringen eftersom dessa komponenter inte kunde placeras i närheten av varandra. Nedre delen av vardera ben var ett begränsat utrymme vilket gjorde det svårt att placera ut komponenter utan att behöva ändra robotens dimensioner.

Ett annat problem som förekom i konstruktionen var att roboten gungade fram och tillbaka när den tippat över och benen slagit i marken. Detta påverkade inte balansen märkvärt men det gjorde att gångstilen blev hackig. Detta försökte åtgärdas genom att sätta en svart gummilist som formats med metall på baksidan av foten samt filt på framsidan för att jämna ut höjden och för att förhindra att roboten skrapade direkt i marken. Gungandet dämpades då lite men kunde inte åtgärdas helt. Om en bättre förstudie kring problem som uppkommit kring liknande konstruktioner hade gjorts, skulle detta möjligtvis inte vara ett problem. Listen togs sedan bort då den inte tillförde till robotens stabilitet.

Lätthålen var en väsentlig del skalets design. Det kunde tidigt ses i CAD-modellen att vikten utan lätthålen skulle bli mycket högre än vad det beräknats att stegmotorerna skulle klara av, se Kapitel 3.2. Ett problem som uppstod var däremot att ändringar i designen blev svåra att utföra efter att sidorna med lätthålen vattenskurits ut. Då delar av modellen ändrades i ett väldigt sent stadie på grund av bland annat ändringar av komponenter, fick de placeras mindre optimalt vilket påverkade masscentrumet. Detta gjordes exempelvis för de linjära aktuatorernas drivkort som i Kapitel 3.3.3 beskrivs att de fick bytas då de ej var kapabla att driva stegmotorn. Även batteriet fick förflyttas längre ner i benet, detta löstes genom att modellera om dess 3D-utskrivna hållare. Dessa förändringar försvårade insättningen av batteriet. Det gick inte efter dessa förändringar att sätta in batteriet utan att behöva montera ner delar av roboten. Detta gjorde att ett av lätthålen på baksidan fick förstoras för att underlätta tillgängligheten. Om testerna av komponenter hade varit utförligare innan

sidorna skars ut hade detta kunnat undvikas men eftersom CAD-modellen skapades parallellt med utförandet utav testerna var detta fallet. För att helt undvika det hade testerna kunnat ha utförts innan designen i CAD.

Sidorna bockades för hand och det gjorde att benen inte blev exakt 100 x 100 mm i bredd och djup vilket de hade modelleras att vara, utan 102 x 98 mm. I sin tur behövde då hållarna för linjäraktuatorn, toppen och den 3D-utskrivna foten anpassas utefter det här. Detta gjorde även roboten bredare än CAD-modellen. Utöver detta skapades då ett glapp mellan axelfäste och axelmotor som missades i CAD-modellen medverkade i den slutgiltiga breddskillnaden i Tabell 6.2.

Axelfästet fick ändra design efter att sidorna skars ut, fästet fick en större diameter än tidigare tänkt. Detta gjorde att axelfästet täckte delar av de tänkta hålen för kabeldragning på sidorna. Det löstes istället med att kablarna drogs genom de stora lätthålen ovanför axeln. Att kablarna drogs genom dessa hål gav både en mindre belastning på kablarna och en större frihet att arbeta med kablarna än om de dragits genom de tänkta hålen. Axelfästena designades för att få en större fästyta på motorns axel vilket blev en förbättring, men när roboten förflyttades och användes tappade dessa fäste efter ett tag. Problem med att axelfästen för samma motorer hade redan påpekats av kandidatarbetet [4]. De hade dock en simplare fästning än den skapad för denna robot, se Figur 4.3. För att förbättra fästet hade det behövts fästas permanent med motorns axel. Detta hade kunnat ske genom svetsning, försöka byta ut eller modifiera motorns axel.

Vid 3D-utskrift användes PLA eftersom det var det material som var lättast att tillgå, det möjliggjorde även att snabbt skriva ut flera olika prototyper, se diskussionen i 2.3. Detta sparade mycket tid i konstruktionsfasen och trots att PLA är ett svagare material än ABS räckte det till för de ändamål det använts för i projektet.

Viktskillnaden i CAD-modellen och den fysiska roboten kan ha berott på ett flertal faktorer. Den fysiska roboten hade slipats på alla dess kanter vilket inte hade tagits hänsyn till i CAD-modellen. Vidare hade motordrivarna av samma modell som beställts omgångar varierande storlekar och vikter. I CAD-modellen var alla motordrivarna av den större och tyngre varianten. CAD-modellen tog heller inte hänsyn till alla sladdar som användes i den fysiska modellen. Vidare användes flera muttrar och skruvar för att sätta ihop modellen. Dessa antogs väga 1 g i CAD-modellen som var ett förinställt värde. Det stämde dock dåligt överens med verkligheten. Viktfelet för skruvarna kompenserade lite för avsaknaden av sladdar men den största faktorn var de olika motordrivarna. I Tabell 6.1 kan det ses att ben 2 var det ben vars konstruerade vikt som stämde bäst överens med den modellerade. En anledning till detta kan vara att motordrivare som användes var av den större varianten och stämde därför bättre överens med den som används i CAD-modellen

7.2 Komponenter

Val och tester av komponenter utfördes kontinuerligt under förstudie och konstruktion. Flera av de först tänkta komponenterna byttes på grund av svårighet att inhandla, otillräcklig prestanda och misstag vid tester.

Valet att utföra tester på komponenter som kan ses i Kapitel 3 gav inte bara ett klart resultat om komponenten skulle klara av det tänkta arbetet utom fungerade samtidigt som ett sätt att bekanta sig hur de kunde styras och avläsas. Eftersom denna kunskap om dessa komponenter skaffades tidigt kunde även styrprogrammet börja designas redan innan roboten var färdigställd. Detta gav en bättre helhetsbild utav projektet och problem som skulle kunna uppkomma i mjukvaran på grund av hårdvara kunde smidigt åtgärdas.

Ett problem som ibland uppstod var att ett ben hamnade efter det andra benet när de svängde fram vilket ofta resulterade med att roboten föll. Detta hade kunnat lösas genom att implementera de installerade accelerometrarna. Benen som skulle flyttas ut till en given referensvinkel hade då kunnat genom vinkelåterkoppling från accelerometrarna fortsätta rotera tills de uppnått önskat värde. Detta kan jämföras med det implementerade som antog att den inmatade vinkeln skulle uppnås. Som tidigare nämnts i Kapitel 5.2.2 klarade de inte av att köras samtidigt som motorerna, detta kan berott på att strömmen som drogs från Arduinon var för hög. Ett möjligt sätt att minska strömmen som går genom Arduinon hade kunnat vara att inte styra *enable*-pinnen med Arduinon utan låta motorerna starta med strömbrytarna istället för ett kommando. Detta då *enable* endast krävdes att vara hög (5 V) under tiden då roboten var igång.

För att slippa problemet med att accelerometrarna endast hade två I2C-adresser diskuterades flera olika lösningar som inkluderade både förändringar i mjukvara och hårdvara. Det hade kunnat var möjligt att byta ut två av accelerometrar till annan variant som använder sig av *Serial Peripherial Interface* (SPI) istället för I2C. En annan hårdvarurelaterad lösning vore att använda sig utav en I2C multiplexer. Om problemet skulle lösas med mjukvara fanns alternativet att skapa en virtuell I2C-buss. Den lösning som till slut användes var att endast läsa data från adressen 0x69 och skifta adressen på den sensor som data lästes ifrån. Denna lösning valdes för att den till synes var enklast att implementera och krävde minst förändring av hårdvaran. Detta fungerade mycket väl i testerna av accelerometrarna, se Kapitel 3.4, men när sensorerna kördes tillsammans med motorerna lyckades roboten endast mäta några gånger innan programmet havererade. Till slut gjordes valet att inte fortsätta arbetet med sensorerna på grund av tidsbrist. Roboten klarade fortfarande att gå men helt utan återkoppling. Ett vanligt fel som gjorde att modellen föll var att benen hamnade ur synk och inte hade samma vinkel. Detta felet hade kunnat avhjälpas med återkoppling av nuvarande vinkel.

En kritisk begränsning av projektet var dess satta budget på 5000 sek, detta gjorde att valet av att hitta passande och prisvärda komponenter blev allt mer viktigt. Detta stödjer utförandet av en allt mer grundlig förundersökning och tester av de bedömda kritiska komponenter såsom motorer, drivkort, sensorer och batterier. För att spara på plats och pengar användes först små stegmotordrivare men efter att ett par stycken havererat bytte projektet till att använda de nuvarande stegmotordrivarna med inbyggt skydd mot flera av de möjliga anledningarna till haveriet. Om en allt större ekonomisk frihet hade givits hade antagligen inte lika stort fokus behövts läggas på att testa komponenter utan genom grav överdimensionering hade möjliga svaga komponenter kunnat undvikas.

7.3 Simulering och styrning

Simuleringen av TARS-modellen gav en god insikt om hur de olika gångstilarna skulle kunnat se ut och vilka potentiella problem som skulle kunna uppkomma. Ett av dessa problem var robotens tendens att falla över efter ett steg då rörelsemängden var hög. Detta medförde att en hastighet framåt gjorde att den enkelt ramlade. För att utöka simuleringens robusthet i jämförelse med verkligheten kunde fler mätningar i dess miljö ha gjorts. En mer robust simulation hade möjliggjort mer kvalitativa tester av gångstilarna.

Ett annat intressant område att studera med simulationen hade varit inom domänen av maskininlärning. Med hjälp av bibliotek i MATLABs Simulink kan rörelsen avgränsas och ange approximativa anvisningar på hur roboten ska ta ett steg. Med denna metod kan programmet själv testa sig fram och efter flera iterationer, ta fram de parametrar som uppfyller målet på bästa sätt. Denna typ av metodik valdes dock inte att undersökas vidare eftersom det bedömdes att syftet kunde uppfyllas enklare genom att själva hitta bäst parametervärden för styrning. Trots brister ansågs simulationen vara tillräcklig för dess syfte då det syntes tydligt i verkligheten att den stämde bra överens, se Figurerna 6.3 - 6.6.

Den styrning som slutligen implementerades var långt från felfri. När flera motorer kördes samtidigt uppstod ett oljud på grund utav att tiden mellan två steg inte var optimal, se 20. För att lösa detta skulle nya basfunktioner kunna skrivas som använde sig av en avbrottsbaserad lösning för att säkerställa att tiden mellan två steg hölls konstant. Ett möjligt sätt att förhindra detta och göra styrningen smidigare hade kunnat var att implementera en Arduino Nano i varje ben. Var ben hade då kunnat styra individuellt samtidigt som var bens mikrodator kommunicerar med varandra för att genomföra en synkroniserad gång. Förslagsvis hade denna kommunikation kunnat göras med hjälp av Nanons I2C-buss.

Styrningen som implementerades på den färdiga roboten hade kunnat göras genom att importera redan skapade styrsignaler från simuleringen. Om detta hade valts att göra hade en detaljerad styrning kunnat designas redan innan roboten blivit färdigställd. Sättet som valdes var istället att genom kunskap från simuleringens styrsignaler skapa nya i styrprogrammet. Visserligen var inte simuleringen från ett styrperspektiv utan framgång men simuleringsverktyget hade kunnat utnyttjas mer.

Den simulerade och den fysiska modellen jämfördes och till skillnad från simuleringen, som vred upp benen 25 °, vred den fysiska modellen endast upp benen 17 °. Detta kunde bero på flera faktorer. Som tidigare nämnts överensstämde inte vikterna för den fysiska modellens ben med CAD-modellens och därmed stämde inte heller vars ben masscentra överens med det tänkta. I den fysiska modellen tycktes tyngdpunkten befinnas längre fram än i CAD-modellen som användes vid simuleringen vilket ledde till att en lägre vinkel krävdes för att få roboten att tippa framåt. En annan faktor är avrundningen av foten som gör det möjligt för roboten att enklare falla framåt. Denna hade skurits ut för hand vilket kan ha lett till att avståndet r i (3) både kan ha varierat från ben till ben men också skiljt sig från det som modellerats. Ett mindre r ger en mindre kritisk vinkel vilket kan vara fallet för den fysiska modellen.

I förstudien, Kapitel 2, togs det upp att roboten kunde modelleras som en dubbelpendel. Denna

kunskap hade varit väsentlig om ett återkopplat reglersystem hade valts att implementeras. Roboten hade då behövts modelleras med en matematisk modell men på grund av att bland annat stegmotorer valdes var detta inte nödvändigt. Om exempelvis DC-motorer hade använts så beskriver dubbelpendel robotens dynamik på ett likartat sätt. Att stegmotorer kan bedömas vara enklare i ett öppet system är deras förmåga att ta förutbestämda steg. Genom att dimensionera valet av motor kan det tänkas att antagandet att motorerna alltid lyckas ta antalet bestämda steg vara rimligt. Systemet blir då diskret och mycket enkelt att implementera. Med en DC-motor hade systemets istället vart kontinuerligt. I och med att stegmotorerna beter sig diskret bedömdes att ett reglersystem skulle vara svårare att implementera än fallet med DC-motorer.

7.4 Tester

Vid tester av de framtagna gångstilarna visade det sig att resultatet var starkt beroende av underlaget. Vid tester i olika lokaler klarade roboten att gå med varierande resultat. Detta kan ha berott på bland annat att friktionen på golven var annorlunda och att det fanns ojämnheter i golvet. Interna faktorer kan också ha spelat roll såsom batterispänning, motortemperatur och hur väl motorer samt andra rörliga delar var fastskruvade. Resultaten som redovisas i Tabellerna 6.3 och 6.4 korrelerar mot hur mycket tid som spenderades på att få gångstilarna att fungera. I gångstil 2 användes alla motorer i axlarna, två av dessa som var kopplade till ben 2 behövde roteras samtidigt annars uppstod ett motstånd av den ena motorn och stegmotorerna tappade steg. Detta hade kunnat förhindras genom att minska tidsfördröjningarna då de beter sig som (20).

Gången uppvisade även en viss rotation kring dess z-axel under stigtiden, detta berodde på axeln konstruktion. Axlarnas infäste kunde motstå vikten av två ben och gav en viss oscillerande rörelse vilket ledde till en ökad instabilitet. Vid infall, se Figur 6.6d, roterade roboten även vid indragning av fötterna då de inte är symmetriska kring tyngdpunkten. Ena benet befann sig längre bort från tyngdpunkten än det andra vid den här gångstilen och därmed tillfördes ett vridmoment som ledde till en liten rotation. Då gångstil 2 alternerade benet som befann sig längre bort, tar denna rotation ut sig mer eller mindre vid jämna steg.

8

Slutsats

Syftet med projektet var att designa och konstruera en modell av roboten TARS från Interstellar som fjärrstyrt skulle kunna gå minst tre olika gångstilar, varav minst en skulle implementeras. Projektet anses uppfylla detta då två gångstilar både simulerades och implementeras i den fysiska roboten, dock med varierande kvalité. Om fler gångstilar önskas kan de enkelt implementeras genom att lägga till kod som följer strukturen på det skapade styrprogrammet.

Problem som uppstod och påverkade resultatet var bland annat en följd av de valda komponenterna och konstruktionen. Ett av dessa var att roboten gungade fram och tillbaka när benen under gång fällts ut och roboten trillat över. Om större kraft hade lagts på de inbyggda men ej fungerande sensorerna hade detta problem antagligen kunnat lösas. Men på grund av begränsad tidsram hanns detta inte med.

En tredje gångstil lyckades varken simuleras eller skapas med den färdigställda roboten. En anledning till detta var gångstilens komplicerade dynamik som beror på att när endast ett ben i taget flyttas, förflyttas även masspunkten så att roboten inte längre kan hållas upprätt. Om större tidsram givits hade denna gångstil kunnat undersökas ytterligare.

Även om projektet bedöms ha uppnått sitt syfte så är den modell av TARS som byggts en bit ifrån den TARS som presenteras i filmen Interstellar, skillnader kan bland annat ses i dess design, storlek och rörelsedynamik.

Om vidare utveckling väljes att göras på den skapade roboten skulle det kunna fokusera på bristande delar såsom axlar, implementering av sensorer och det skapade styrprogrammet. Genom dessa förbättringar bör de redan existerande gångstilarna kunna förbättras samtidigt som fler gångstilar skulle kunna skapas.

Referenser

[1] Ahmetkozan, "Interstellar," 2017. [Online]. Tillgänglig: https://www.imdb.com/title/tt0816692/, hämtad; 2019-02-08.

[2] Boston Dynamics, "Spot Takes a Kicking and Keeps on Ticking," 2018. [Online]. Tillgänglig: https://www.bostondynamics.com/spot-classic, hämtad: 2019-02-13.

[3] C. Ansin, K.-O. Gunnarson, J. Hejderup, H. Månson, E. Nehmé och N. Tran Luu, "TARS: - Robot från Interstellare," examensarbete för kandidatexamen, Instutionen för Elektroteknik, Chalmers tekniska högskola, Göteborg, Sverige, 2018, [Online]. Tillgänglig: http://studentarbeten.chalmers.se/publication/255869-tars-roboten-fran-interstellar, hämtad: 2019-05-16.

[4] L. Crnic, S. Hafström, S. Johansson, D. Olsson, E. Säterskog och H. Winblad, "LARS - Gestaltning och konstruktion av en robot inspirerad av TARS från filmen Interstellar," examensarbete för kandidatexamen, Institutionen för Elektroteknik, Chalmers tekniska högskola, Göteborg, Sverige, 2018, [Online]. Tillgänglig: http://studentarbeten.chalmers.se/publication/255881-lars-gestaltning-och-konstruktion-av-en-robot-inspirerad-av-tars
-fran-filmen-interstellar, hämtad: 2019-05-16.

[5] A.B Higaki, "TARS from Interstellar: A Walking Robot Invention (Award-Winning Engineering Project)," 2017. [Online]. Tillgänglig: https://www.youtube.com/watch?v=IdvzREqSIA4, hämtad: 2019-02-10.

[6] K. Hirai, M. Hirose, Y. Haikawa och T. Takenaka, "The development of Honda humanoid robot," i *Proceedings. 1998 IEEE International Conference on Robotics and Automation*, Leuven, Belgium, 1998, vol.2, ss. 1321-1326. [Online].
Tillgänglig: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=677288&isnumber=14899, hämtad: 2019-05-16.

[7] R. Gerndt, D. Seifert, J. H. Baltes, S. Sadeghnejad och S. Behnke, "Humanoid Robots in Soccer: Robots Versus Humans in RoboCup 2050," *IEEE Robotics & Automation Magazine*, vol. 22, nr. 3, ss. 147-154, sept. 2015. doi: 10.1109/MRA.2015.2448811, [Online]. Tillgänglig: https://ieeexplore.ieee.org/document/7254325, hämtad: 2019-05-16.

[8] K. Rushdi, D.Koop och C.Q. Wu, "Experimental studies on passive dynamic bipedal walking," *Robotics and Autonomous Systems*, vol. 62, nr. 4, ss. 446-455, April, 2014. doi:https://doi.org/10.1016/j.robot.2013.12.002, [Online].
Tillgänglig: https://www.sciencedirect.com/science/article/pii/S0921889013002303,
hämtad: 2019-05-16.

[9] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi och H. Hirukawa, "The 3D linear inverted pendulum mode: a simple modeling for a biped walking pattern generation," i *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium*, Maui, HI, USA, 2001, ss. 239-246 vol.1. Tillgänglig: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=973365&isnumber=20976, hämtad: 2019-05-16.

[10] A. Goswami, B. Espiau och A. Keramane, "Limit cycles and their stability in a passive bipedal gait," i *Proceedings of IEEE International Conference on Robotics and Automation*, Minneapolis, MN, USA, 1996, ss. 246-251 vol.1. doi: 10.1109/ROBOT.1996.503785. Tillgänglig: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=503785&isnumber=10815, hämtad: 2019-05-16.

[11] M. Vukobratovic, B. Borovac, D. Surla och D. Stokic, *Scientific Fundamentals of Robotics 7: Biped Locomotion*, 1 uppl, Berlin, Tyskland: Springer Verlag, 2013. [Online]. Tillgänglig: https://link.springer.com/book/10.1007%2F978-3-642-83006-8, hämtad: 2019-05-16.

[12] T. Shinbrot, C. Grebogi och A. keramane, "Chaos in a double pendulum," *American Journal of Physics 60*, nr. 6, ss. 491-599, dec. 1992. doi: 10.1119/1.16860.
illgänglig: https://aapt.scitation.org/doi/10.1119/1.16860, hämtad: 2019-05-10.

[13] O. Ravn, "DTU Acrobot," 2010. [Online]. Tillgänglig:
https://www.youtube.com/watch?v=sMZRnE3q72c, hämtad: 2019-02-10.

[14] Y. Xiaom, "Acrobot Swing Up with MATLAB," *arXiv e-prints*, jan. 2019. [Online]. Tillgänglig: https://ui.adsabs.harvard.edu/abs/2019arXiv190103952X, hämtad: 2019-05-16.

[15] M. Ashby, H. Shercliff och D. Cebon, *Materials: Engineering, Science, Processing and Design*, uppl. 3, Storbrittanien: Butterworth-Heinemann, 2014.

[16] K. Giang, "PLA vs. ABS: What's the difference?," 2017. [Online].
Tillgänglig: https://www.3dhubs.com/knowledge-base/pla-vs-abs-whats-difference,
hämtad: 2019-03-22.

[17] Naturskyddsföreningen, "De olika plasterna," 2019. [Online].
Tillgänglig: https://www.naturskyddsforeningen.se/node/35087/#H, hämtad: 2019-05-12.

[18] Robotshop inc, Quebec, Canada. "12V, 1.7A, 416 oz-in Geared Bipolar Stepper Motor,"
u.d. [Online]. Tillgänglig: https://www.robotshop.com/eu/en/12v-17a-416oz-geared-bipolar-stepper-motor.html, hämtad: 2019-04-26.

[19] Arduino, "ARDUINO MEGA 2560 REV3," 2019. [Online].
Tillgänglig: https://store.arduino.cc/mega-2560-r3, hämtad: 2019-04-24.

[20] X. Hu, “An Introduction to Stepper Motors,” u.d. [Online].
Tillgänglig: https://wp.optics.arizona.edu/optomech/wp-content/uploads/sites/53/2016/10/Tutorial_Xinda-Hu.pdf, hämtad: 2019-05-01.

[21] J.A. Collins, *Mechanical design of machine elements and machines : a failure prevention perspective*, NY, USA: Wiley cop, 2003.

[22] StructX, “Static Friction Coefficients,” 2014. [Online].
Tillgänglig: https://structx.com/Material_Properties_005a.html, hämtad: 2019-05-08.

[23] *QMOT QSH4218 MANUAL*, Hamburg, Tyskland: TRINAMIC Motion Control GmbH & Co. KG, 2011. [Online]. Tillgänglig:
https://www.trinamic.com/fileadmin/assets/Products/Motors_Documents/QSH4218_manual.pdf, hämtad: 2019-04-26.

[24] *EVALSP820-XS: Compact evaluation board for STSPIN820 stepper motor driver*: STMicroelectronics NV, 2018. [Online]. Tillgänglig: http://www.farnell.com/datasheets/2600096.pdf?_ga=2.4943664.284985375.1557995344-1194436739.1557323363, hämtad: 2019-04-26.

[25] *General Safety Considerations: General Safety Considerations for Stepper Motor Applications* California, USA: Anaheim Automation, u.d. [Online].
Tillgänglig: https://www.anaheimautomation.com/manuals/forms/Common%20Causes%20for%20Step%20Motor-Driver%20Problems.pdf, hämtad 2019-05-16.

[26] *TB6600 Stepper Motor Driver User Guide*: DFRobot, u.d. [Online]. Tillgänglig: https://forum.arduino.cc/index.php?action=dlattach;topic=558523.0;attach=266385, hämtad: 2019-04-26.

[27] N. Nicholas, “How to use the accelerometer- gyroscope GY-521,” 2013. [Online]. Tillgänglig: https://create.arduino.cc/projecthub/Nicholas_N/how-to-use-the-accelerometer-gyroscope-gy-521-6dfc19, hämtad: 2019-05-08.

[28] *MPU-6000 and MPU-6050 Product Specification Revision 3.4*, Sunnyvale, USA: InvenSense Inc., 2013. [Online]. Tillgänglig: https://store.invensense.com/datasheets/invensense/MPU-6050_DataSheet_V3%204.pdf, hämtad: 2019-05-08.

[29] *MPU-6000 and MPU-6050 Register Map and Descriptions Revision 4.2*, Sunnyvale, USA: InvenSense Inc., 2013. [Online]. Tillgänglig: https://www.invensense.com/wp-content/uploads/2015/02/MPU-6000-Register-Map1.pdf, hämtad: 2019-05-16.

[30] HobbyKing, “ZIPPY Compact 2700mAh 3S 25C Lipo Pack,” 2019. [Online]. Tillgänglig: https://hobbyking.com/en_us/zippy-compact-2700mah-3s-25c-lipo-pack.html, hämtad: 2019-05-03.

[31] Battery University, “Learn About Batteries: BU-402: What Is C-rate?,” 2017. [Online]. Tillgänglig: https://batteryuniversity.com/learn/article/what_is_the_c_rate, hämtad 2019-05-16.

[32] Augustine Aelevanthara, “Arduino Mega 2560,” 2012. [Online]. Tillgänlig: https://grabcad.com/library/arduino-mega-2560–1, hämtad: 2019-05-16.

[33] Panasonic, “ASQM17428 | Turquoise Stroke Mini Switches (ASQM1) Download,” u.d. [Online]. Tillgänglig: https://www3.panasonic.biz/ac/e/dl/cad/index.jsp?series_cd=3066&part_no=ASQM17428, hämtad 2019-05-16.

[34] HAS, “SMAKN TB6600 STEPPER MOTOR DRIVER,” 2017. [Online]. Tillgänlig: https://grabcad.com/library/smakn-tb6600-stepper-motor-driver-1, hämtad: 2019-05-16.

[35] denshi kousaku, “XL6009,” 2017. [Online]. Tillgänglig: https://grabcad.com/library/xl6009-1, hämtad: 2019-05-16.

[36] ELECTRONOOBS, “We have 3 types of non-insulated DC to DC convertes: Buck, Boost and Buck-Boost configurations.,” 2019. [Online]. Tillgänglig: http://www.electronoobs.com/eng_circuitos_tut10.php, hämtad: 2019-05-16.

[37] *Product Data sheet*, Guangzhou, Kina: Guangzhou HC Information Technology Co., u.d. [Online]. Tillgänglig: https://www.olimex.com/Products/Components/RF/BLUETOOTH-SERIAL-HC-06/resources/hc06.pdf, hämtad: 2019-05-08.

[38] *8-BIT BIDIRECTIONAL VOLTAGE-LEVEL TRANSLATOR WITH AUTO-DIRECTION SENSING AND ±15-kV ESD PROTECTION* , Dallas, USA: Texas Instruments, u.d. [Online]. Tillgänglig: https://cdn-shop.adafruit.com/datasheets/txb0108.pdf, hämtad: 2019-05-08.

[39] S. Miller, “Simscape Multibody Contact Forces Library,” 2019. [Online]. Tillgänglig: https://se.mathworks.com/matlabcentral/fileexchange/47417-simscape-multibody-contact-forces-library, hämtad: 2019-02-04.

[40] MathWorks, “General Variable Mass,” 2019. [Online]. Tillgänglig: https://se.mathworks.com/help/physmod/sm/ref/generalvariablemass.html, hämtad: 2019-03-10.

[41] A. Morar, “Stepper Motor Model for Dynamic Simulation,” 2003. [Online]. Tillgänglig: https://pdfs.semanticscholar.org/55c3/e1287d30e3a2d9d93760dd04f1f7ebb00c93.pdf, hämtad: 2019-05-16.

[42] Arduino, “Frequently Asked Questions,” 2019. [Online]. Tillgänglig: https://www.arduino.cc/en/Main/FAQ, hämtad: 2019-05-16.

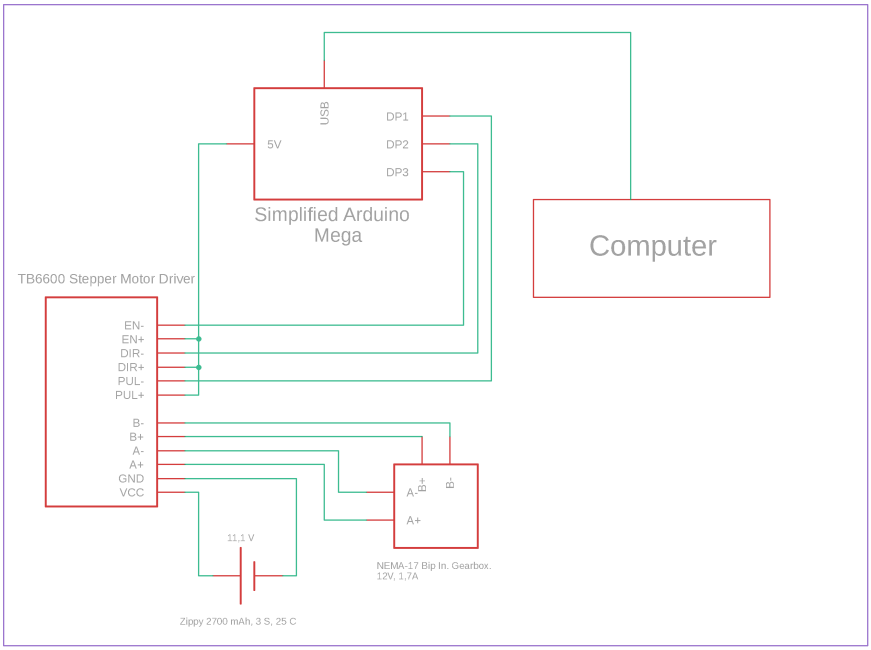
[43] Giumig Apps, “Arduino bluetooth controller,” 2016. [Onliine].
Tillgänglig: https://play.google.com/store/apps/details?id=com.giumig.apps.
bluetoothserialmonitor, hämtad: 2019-05-16.

[44] *TB6600: Stepper Motor Driver*, 2018. [Online].
Tillgänglig: https://webseite.sorotec.de/download/anleitungen/lowcoast
_schrittmotor_endstufen/TB6600_data_sheet_en_Soro_180122.pdf, hämtad: 2019-05-13.

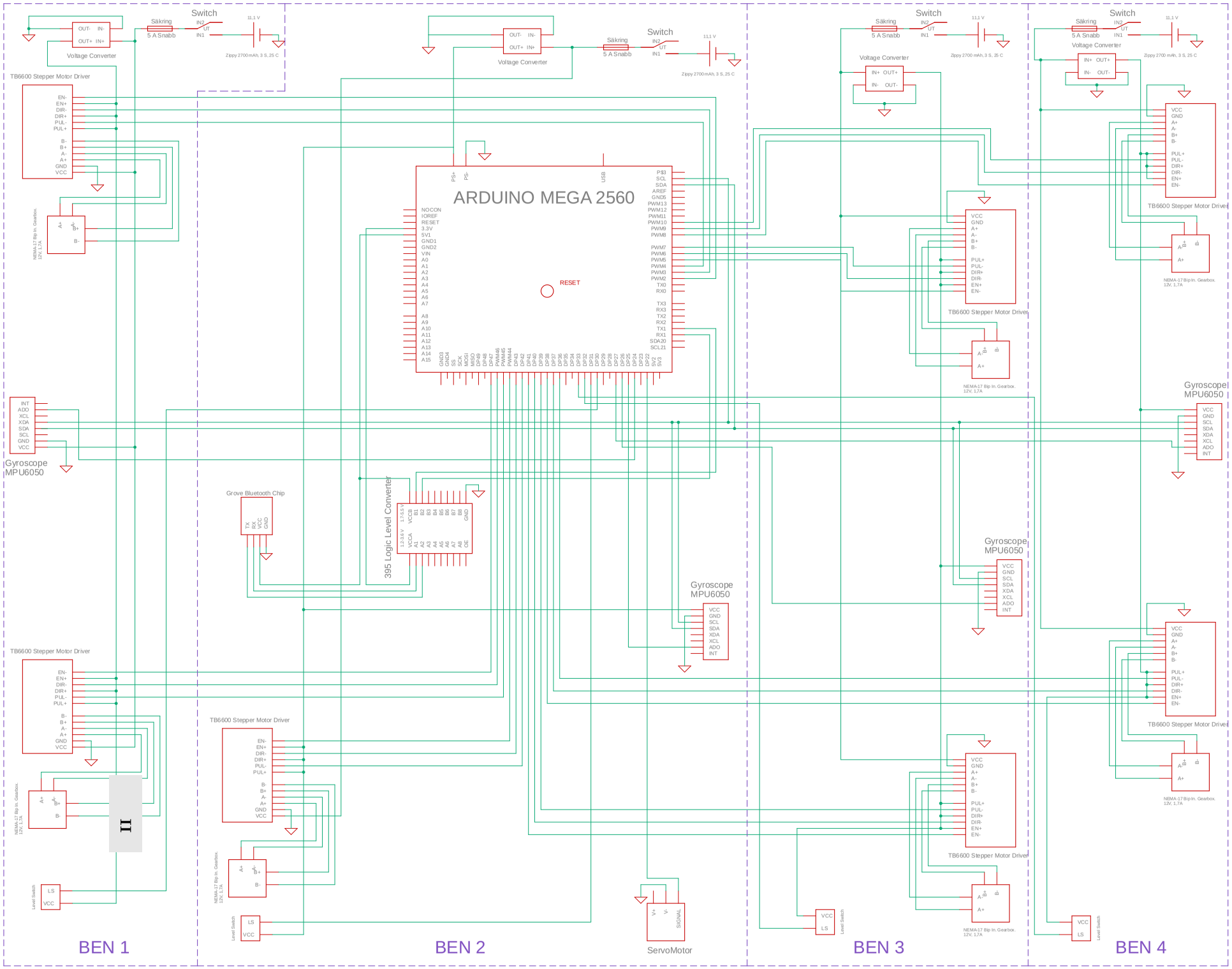
[45] *Atmel ATmega640/V-1280/V-1281/V-2560/V-2561/V: 8-bit Atmel Microcontroller
with 16/32/64KB In-System Programmable Flash*, San Jose, USA: Atmel, 2014. [Online].
Tillgänglig: https://ww1.microchip.com/downloads/en/devicedoc/atmel-2549-8-bit-
avr-microcontroller-atmega640-1280-1281-2560-2561_datasheet.pdf, hämtad: 2019-05-16.

Bilagor

A Kretsschema



Figur A.1: *Kretsschema för kretsen som användes vid test av de båda stegmotorerna.*


ARDUINO MEGA 2560
RESET
Voltage Converter
Switch
Säkring
5 A Snabb
11,1 V
Zippy 2700 mAh, 3 S, 25 C
TB6600 Stepper Motor Driver
NEMA-17 Bip In. Gearbox. 12V, 1,7A
Gyroscope MPU6050
Grove Bluetooth Chip
395 Logic Level Converter
Level Switch
ServoMotor
BEN 1
BEN 2
BEN 3
BEN 4

B Styrning

Tabell B.1: *Pinnar använda på Arduino Mega samt dess förkortningar i det skrivna styrporgrammet.*

Pinne	Akronym	Beskrivning
3	PUL_L4	Puls-signal drivkort motor axel ben 4
4	DIR_L4	Riktnings-signal drivkort motor axel ben 4
5	ENA_L4	Aktiverings-signal drivkort motor axel ben 4
6	PUL_L3	Puls-signal drivkort motor axel ben 3
5	DIR_L3	Riktnings-signal drivkort motor axel ben 3
8	ENA_L3	Aktiverings-signal drivkort motor axel ben 3
9	PUL_L1	Puls-signal drivkort motor axel ben 1
10	DIR_L1	Riktnings-signal drivkort motor axel ben 1
11	ENA_L1	Aktiverings-signal drivkort motor axel ben 1
14	SERVO	Styrsignal till servomotor i ben 2
18	BTout	Blåtand RX, utsignal till blåtand-modul
19	BTin	Blåtand TX, insignal från blåtand-modul
24	AD0_1	Signal för att läsa från accelerometer i ben 1
25	AD0_2	Signal för att läsa från accelerometer i ben 2
26	AD0_3	Signal för att läsa från accelerometer i ben 3
27	AD0_4	Signal för att läsa från accelerometer i ben 4
30	endSens1	Signal från ändlägesgivare i ben 1
31	endSens2	Signal från ändlägesgivare i ben 2
32	endSens3	Signal från ändlägesgivare i ben 3
33	endSens4	Signal från ändlägesgivare i ben 4
36	PUL_LS4	Puls-signal drivkort motor kulskruv ben 4
37	DIR_LS4	Riktnings-signal drivkort motor kulskruv ben 4
38	ENA_LS4	Aktiverings-signal drivkort motor kulskruv ben 4
39	PUL_LS3	Puls-signal drivkort motor kulskruv ben 3
40	DIR_LS3	Riktnings-signal drivkort motor kulskruv ben 3
41	ENA_LS3	Aktiverings-signal drivkort motor kulskruv ben 3
42	PUL_LS2	Puls-signal drivkort motor kulskruv ben 2
43	DIR_LS2	Riktnings-signal drivkort motor kulskruv ben 2
44	ENA_LS2	Aktiverings-signal drivkort motor kulskruv ben 2
45	PUL_LS1	Puls-signal drivkort motor kulskruv ben 1
46	DIR_LS1	Riktnings-signal drivkort motor kulskruv ben 1
47	ENA_LS1	Aktiverings-signal drivkort motor kulskruv ben 1
SCL	SCL	SCL-kommunikation till accelerometer
SDA	SDA	SDA-kommunikation till accelerometer

C Budget

Tabell C.1: *En sammanställning av totala kostnaden för roboten, observera att kvantiteterna av de olika komponenterna inte alltid överensstämmer med konceptet. Detta beror på att vissa komponenter kunde tas från tidigare arbeten samt att vissa valdes att inte att använda för det slutgiltiga produkten.*

Komponent	**Antal [st.]**	**Total kostnad [sek.]**
Motor axel	1	520
Motor linjär aktuator	4	1236
Drivkort	5	567
Kulskruv med Nutplate och Cooupling	4	750
Drivkort kulskruv 1 (ej fungerande)	5	400
Drivkort kulskruv 2 (ej fungerande)	1	121
Servo	1	154
Skruvar och muttrar	100 par	61
Säkringar (inkluderat säkringshållare	-	55
Brytare	4	72
Gyroskop	1	81
Från Chalmersrobotförening	-	500
Totalt	-	4517

D Gångtabeller

Tabell D.1: *Beskriver i vilken ordning som de olika funktionerna anropas, anrop inom samma ordning anropas samtidigt. delay(); anropas för att låta roboten sluta att gunga efter att den fallit över.*

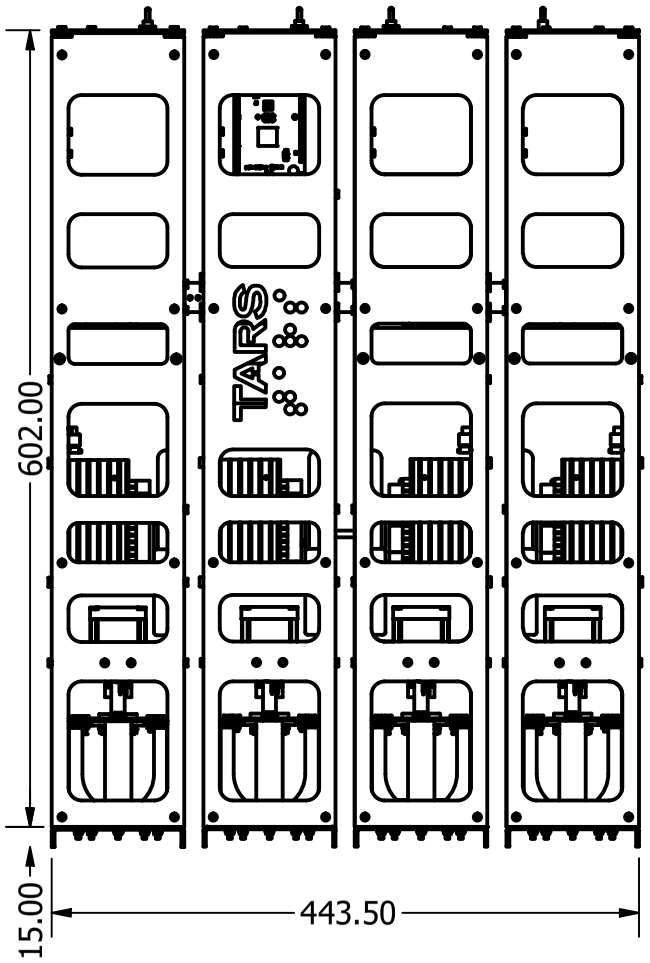
Gångstil 0.5		
Ordning	**Anrop**	**Förändring**
1	Leg1.pushOut();	4 mm
	Leg4.pushOut();	4 mm
2	Leg1.rotateForward();	14 °
	Leg4.rotateBackward();	14 °
3	delay(1700);	
4	Leg1.pushOut();	18 mm
	Leg4.pushOut();	18 mm
5	Leg1.pushIn();	22 mm
	Leg4.pushIn();	22 mm
	Leg2.pushOut();	≈10 mm
	Leg3.pushOut();	≈10 mm
	Leg1.rotateBackward();	14 °
	Leg4.rotateForward();	14 °
6	Leg2.pushIn();	≈ 10 mm
	Leg3.pushIn();	≈ 10 mm

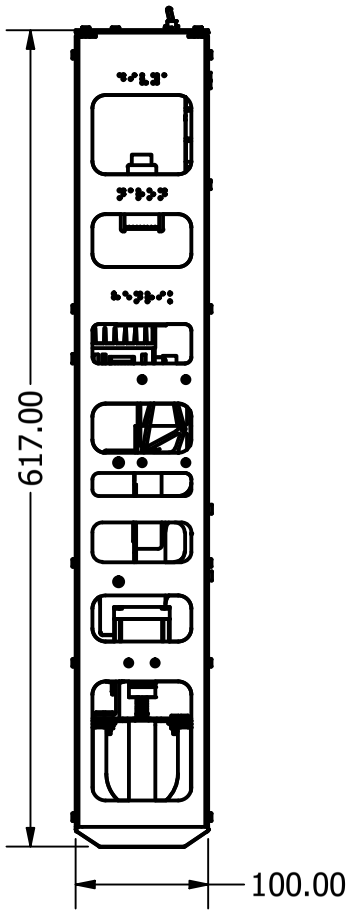
Tabell D.2: *Beskriver i vilken ordning som de olika funktionerna anropas för gångstil 1, anrop inom samma instruktion anropas samtidigt. delay(); anropas för att låta roboten sluta att gunga efter att den fallit över.*

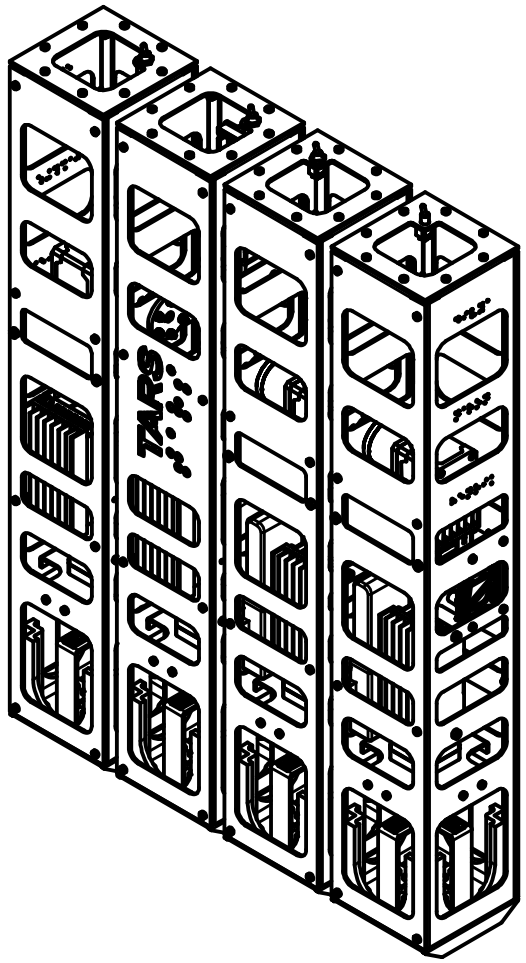
Gångstil 2		
Instruktion	**Anrop**	**Förändring**
1	Leg2.pushOut();	6 mm
	Leg4.pushOut();	6 mm
2	Leg1.rotateBackward();	14 °
	Leg3.rotateForward();	14 °
	Leg4.rotateBackward();	14 °
3	delay(1700);	
4	Leg2.pushOut();	14 mm
	Leg4.pushOut();	14 mm
5	Leg2.pushIn();	20 mm
	Leg4.pushIn();	20 mm
	Leg1.pushOut();	≈ 18 mm
	Leg3.pushOut();	≈ 18 mm
	Leg1.rotateForward();	28 °
	Leg3.rotateBackward();	28 °
	Leg4.rotateForward();	28 °
6	delay(1700);	
7	Leg1.pushIn();	18 mm
	Leg3.pushIn();	18 mm

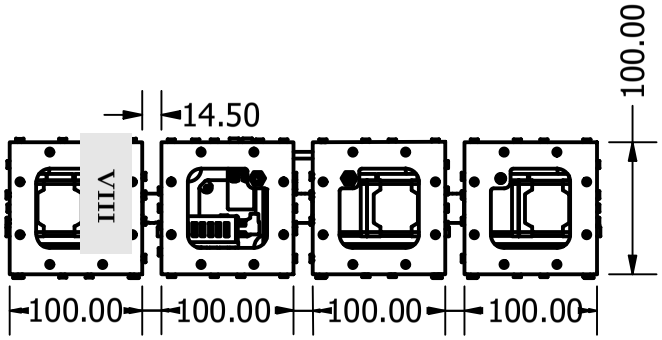
E Ritningar

Följande ritningar är skapade med Autodesk Inventor och är om utskrivet på A4 skalbara enligt respektives beskrivning.

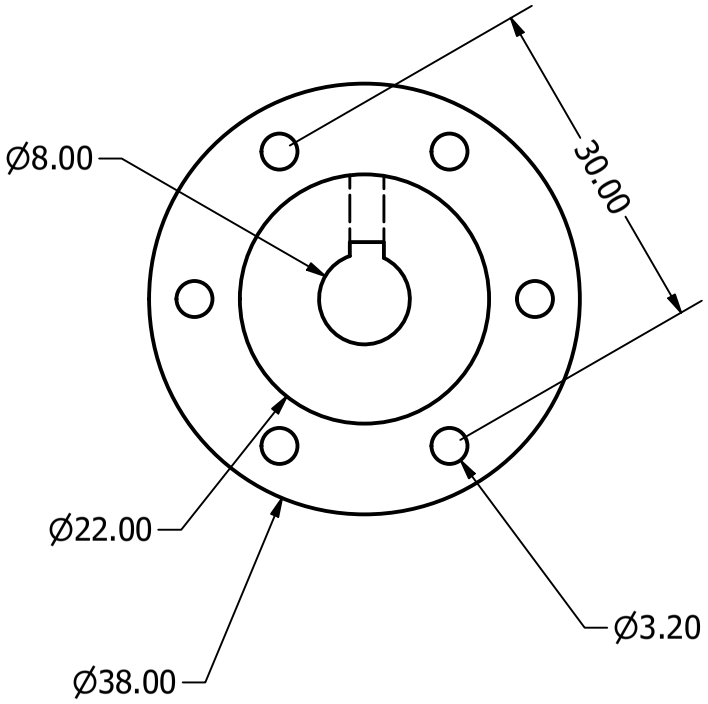


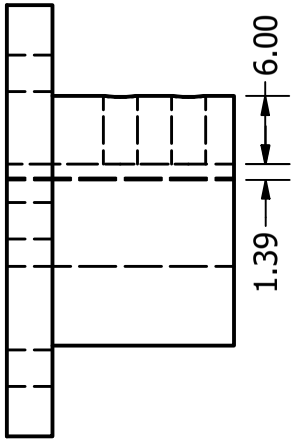


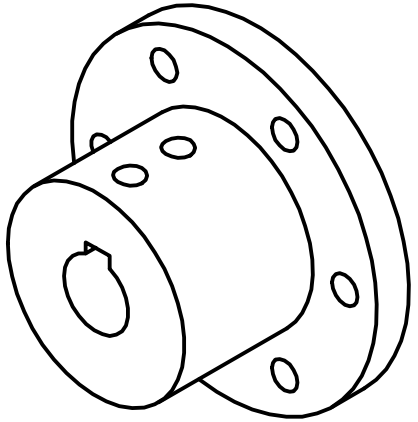


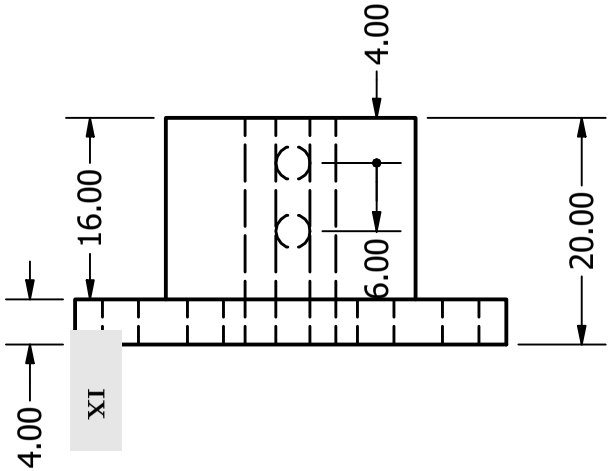


DRAWN Hampus	2019-03-03	EENX15-19-20		
CHECKED				
QA		TITLE		
MFG		Tars		
APPROVED				
		SIZE A4	DWG NO Tars complete	REV
		SCALE 0,175 : 1	[mm]	SHEET 1 OF 1

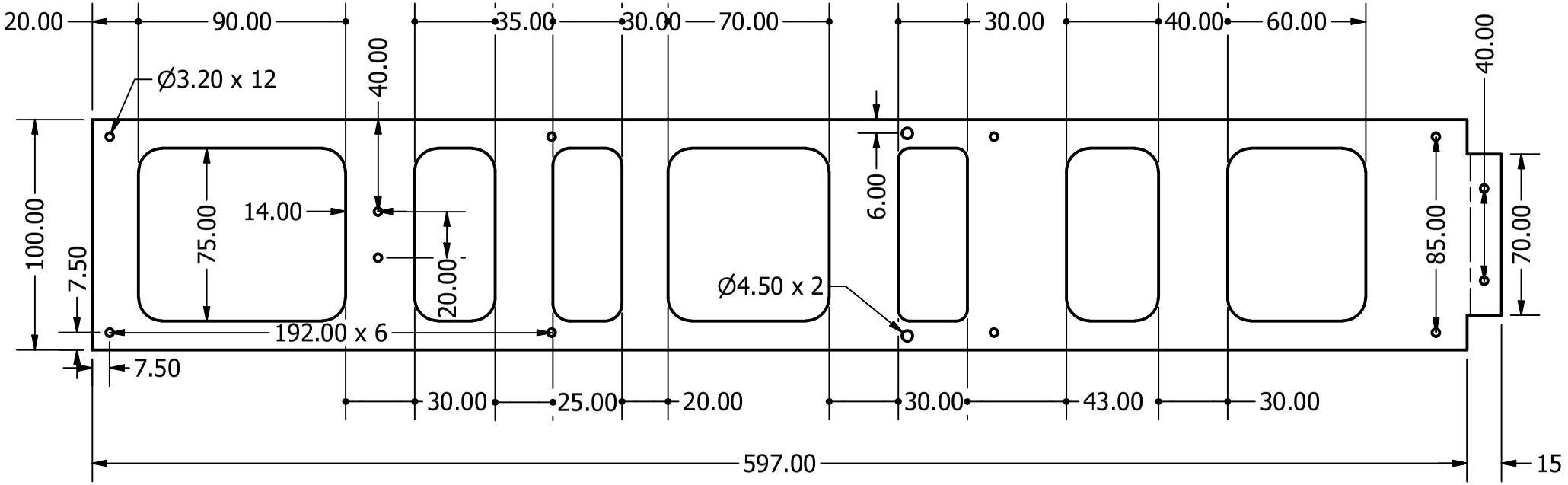




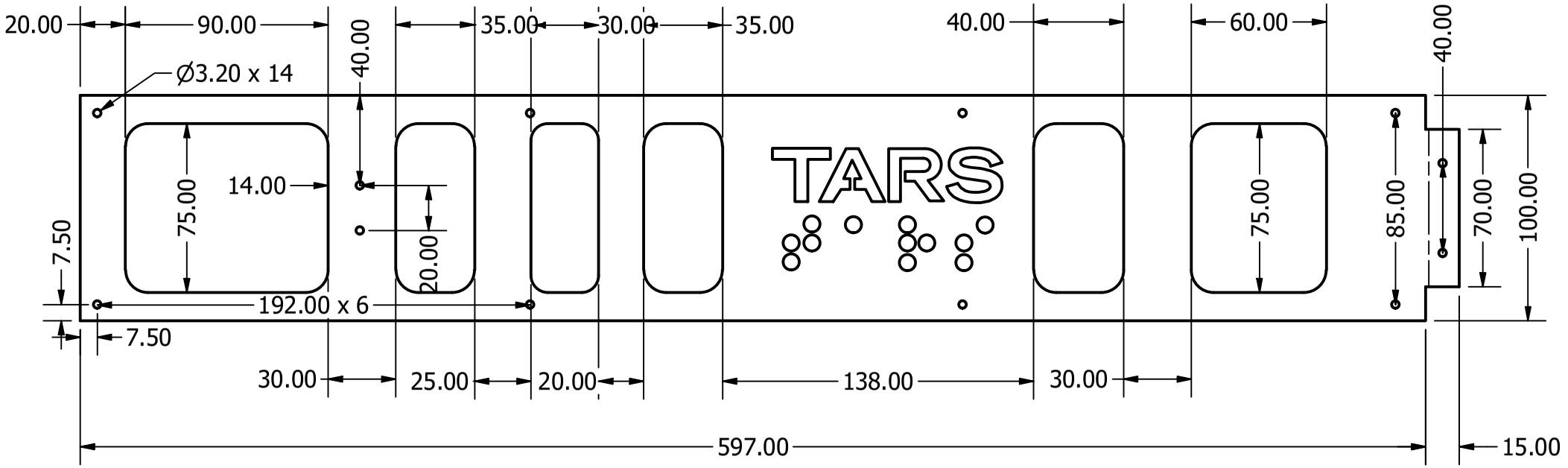




DRAWN Hampus	2019-03-05	EENX15-19-20		
CHECKED				
QA		TITLE Axelkoppling		
MFG				
APPROVED				
		SIZE A4	DWG NO axel	REV
		SCALE 1.5 : 1	[mm]	SHEET 1 OF 1

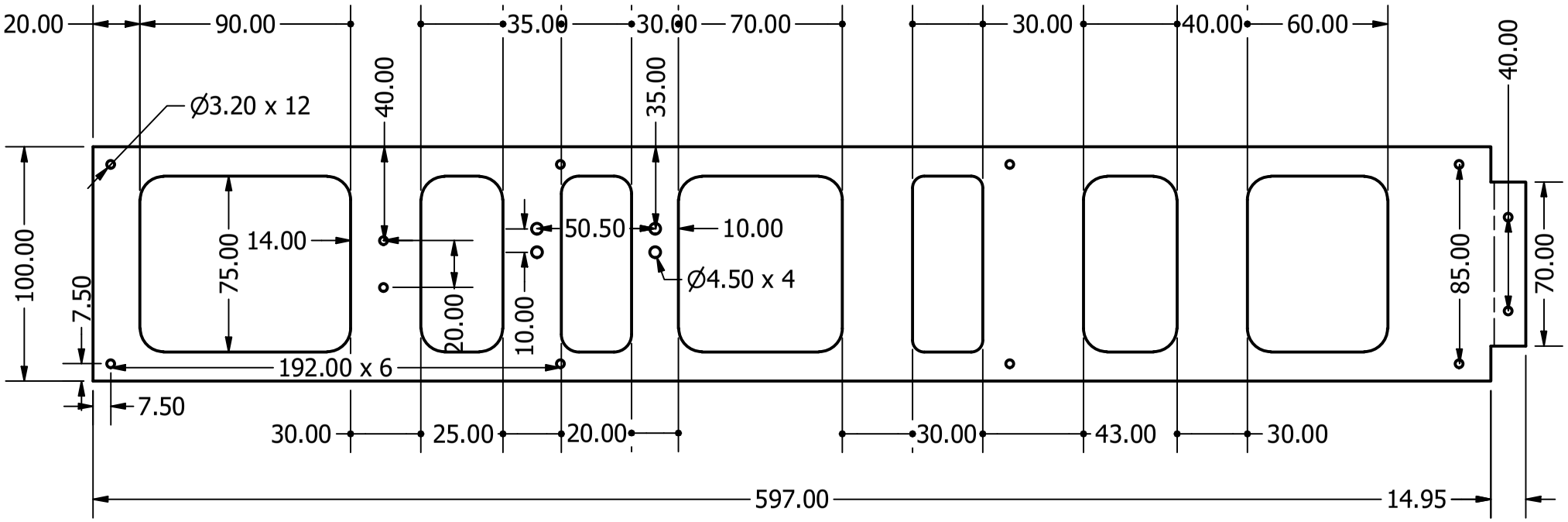


DRAWN Hampus	2019-03-05	EENX15-19-20
CHECKED		
QA		TITLE: Framsida 1
MFG		
APPROVED		SIZE A4 | DWG NO Ben1F | REV
		SCALE 0.4 | [mm] | SHEET 1 OF 1



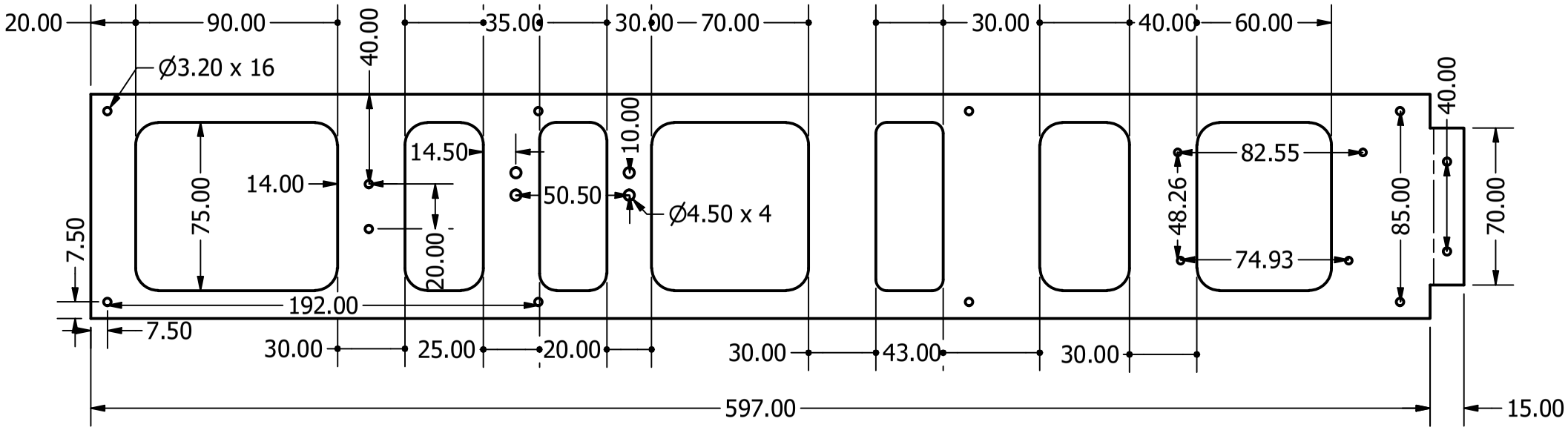
IX

DRAWN Hampus	2019-03-04	EENX15-19-20			
CHECKED		TITLE			
QA		Framsida 2			
MFG					
APPROVED		SIZE A4		DWG NO Ben2F	REV
		SCALE 0,4 : 1	[mm]	SHEET 1 OF 1	



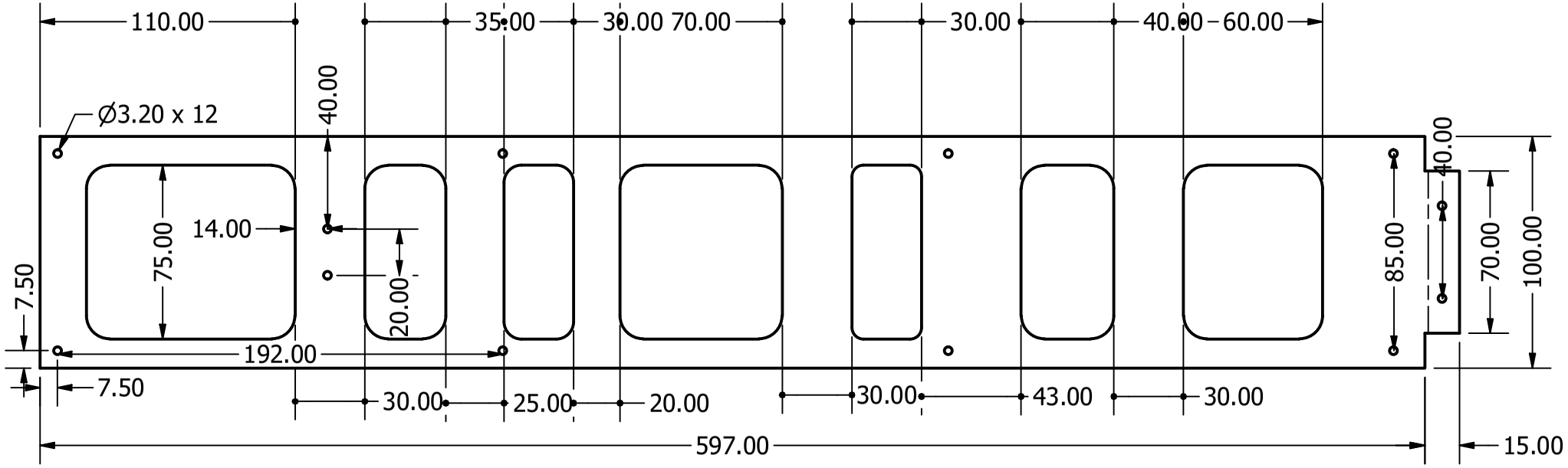
XII

DRAWN Hampus	2019-03-05	EENX15-19-20		
CHECKED		TITLE Baksida 1		
QA				
MFG				
APPROVED		SIZE A4	DWG NO Ben1B	REV
		SCALE 0,4 : 1		SHEET 1 OF 1



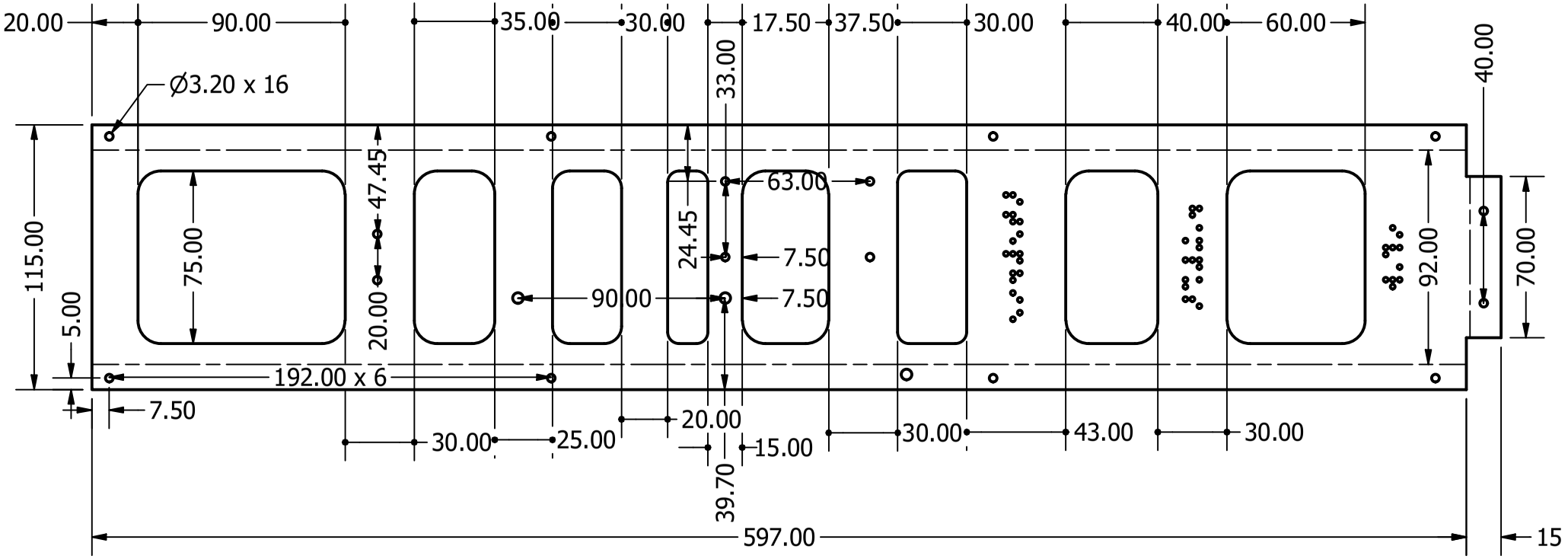
IIIX

DRAWN Hampus	2019-04-15	EENX15-19-20			
CHECKED					
QA		TITLE			
MFG		Baksida 2			
APPROVED					
		SIZE A4		DWG NO Ben2B	REV
		SCALE 0,4 : 1	[mm]	SHEET 1 OF 1	

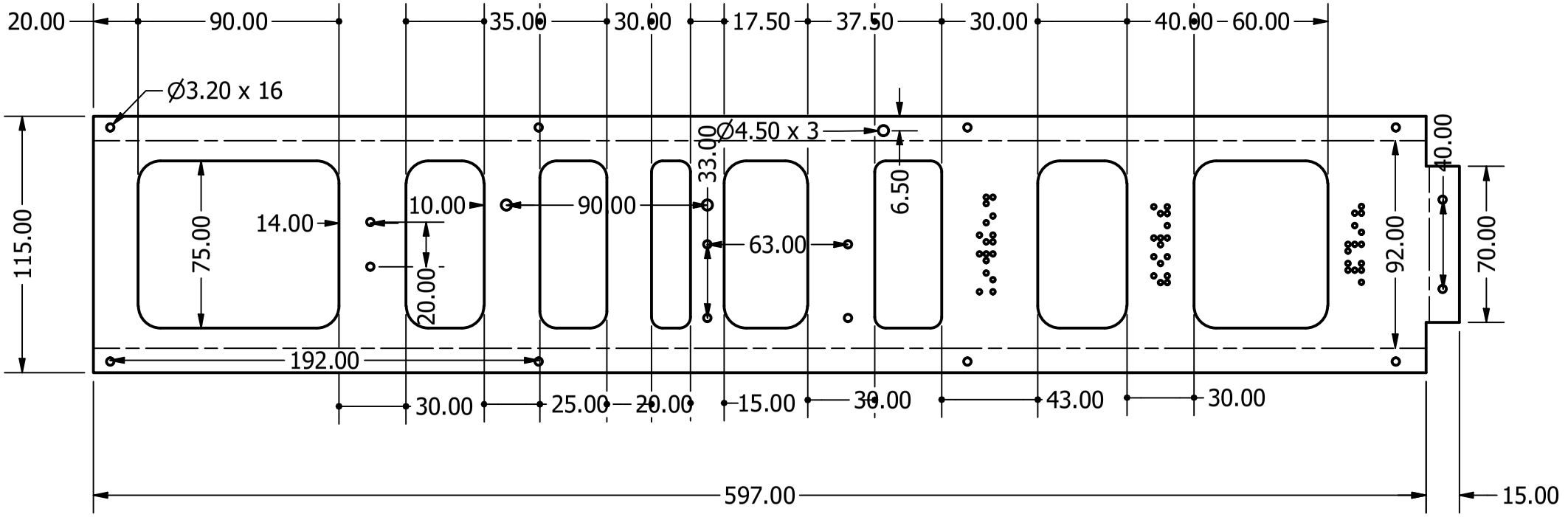


XIV

DRAWN Hampus	2019-04-15	EENX15-19-20		
CHECKED		TITLE		
QA		Baksida 3		
MFG				
APPROVED		SIZE A4	DWG NO	REV
		SCALE 0,4 : 1	[mm]	SHEET 1 OF 1

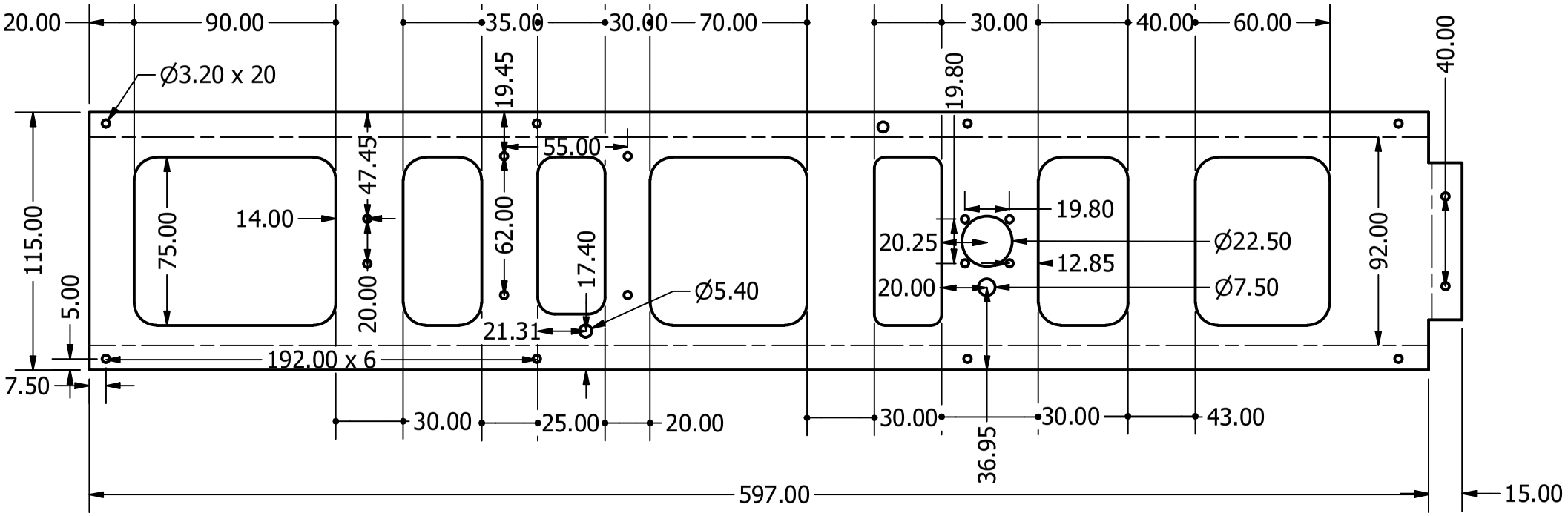


DRAWN Hampus	2019-03-05	EENX15-19-20
CHECKED		
QA		TITLE: Ben1 vänstersida
MFG		
APPROVED		
		SIZE A4 | DWG NO Ben1V | REV
		SCALE 0,4:1 | [mm] | SHEET 1 OF 1



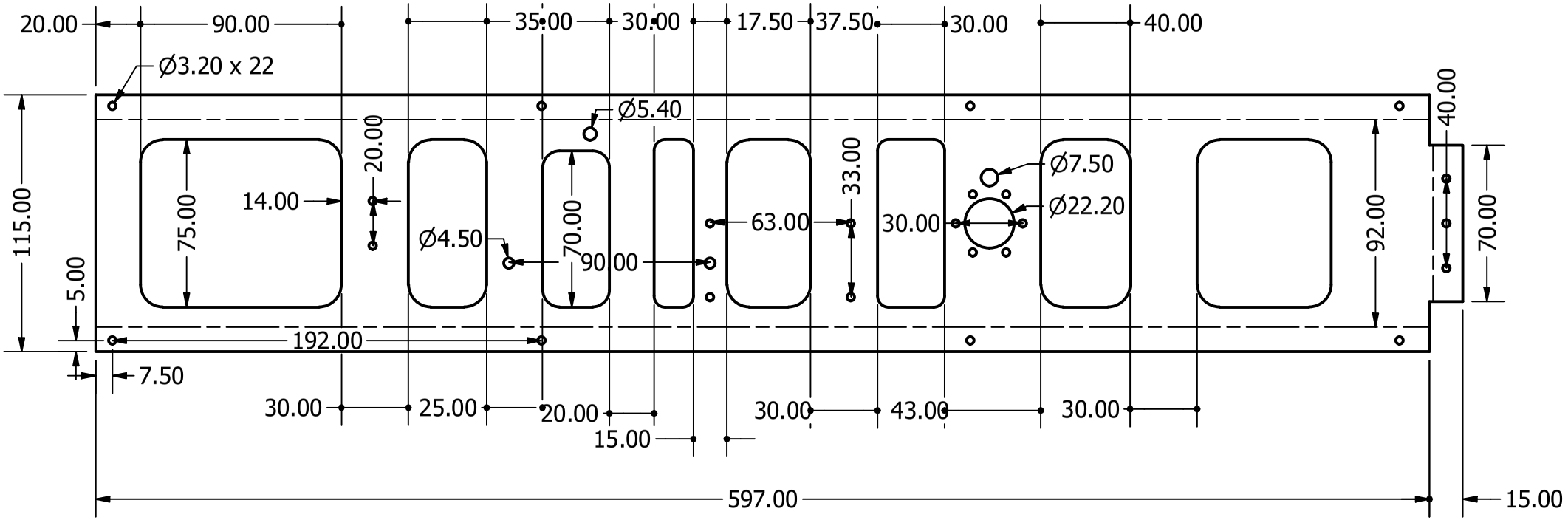
DRAWN Hampus	2019-04-15	EENX15-19-20
CHECKED		
QA		TITLE: Ben 4 högersida
MFG		
APPROVED		
		SIZE A4 | DWG NO | REV
		SCALE 0,4 : 1 | [mm] | SHEET 1 OF 1

XVI



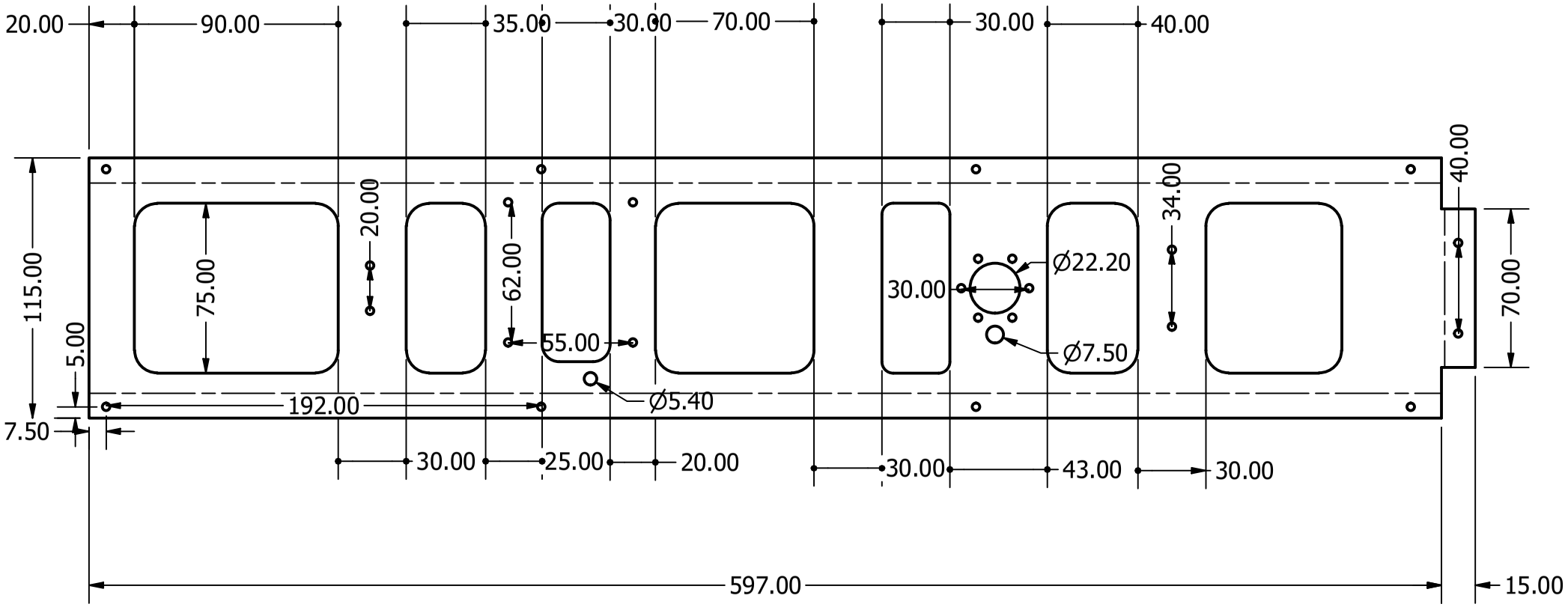
DRAWN Hampus	2019-03-05	EENX15-19-20
CHECKED		
QA		TITLE: Ben 1 högersida Ben 3 vänstersida Ben 4 vänstersida
MFG		
APPROVED		
		SIZE A4 / DWG NO Ben1H / REV
		SCALE 0.4:1 / SHEET 1 OF 1

XVII



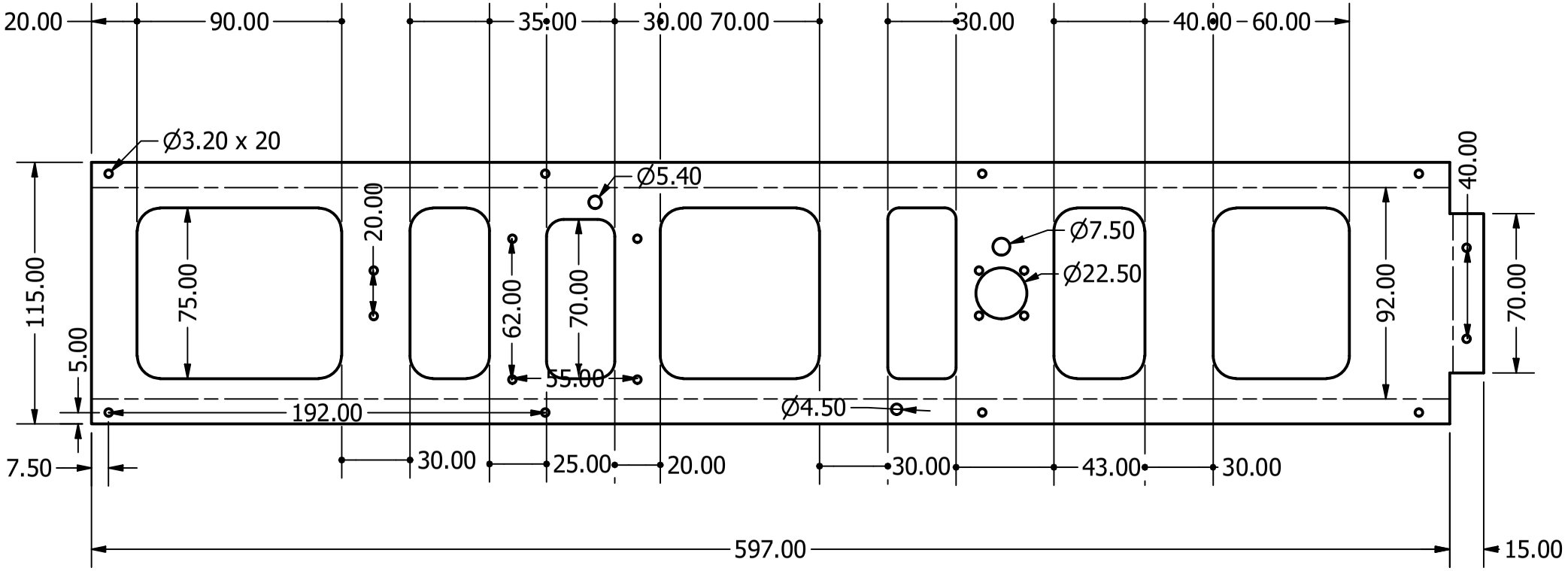
XVIII

DRAWN Hampus	2019-04-15	EENX15-19-20
CHECKED		
QA		TITLE: Ben 2 vänstersida
MFG		
APPROVED		
		SIZE A4 | DWG NO | REV
		SCALE 0,4: 1 | [mm] | SHEET 1 OF 1



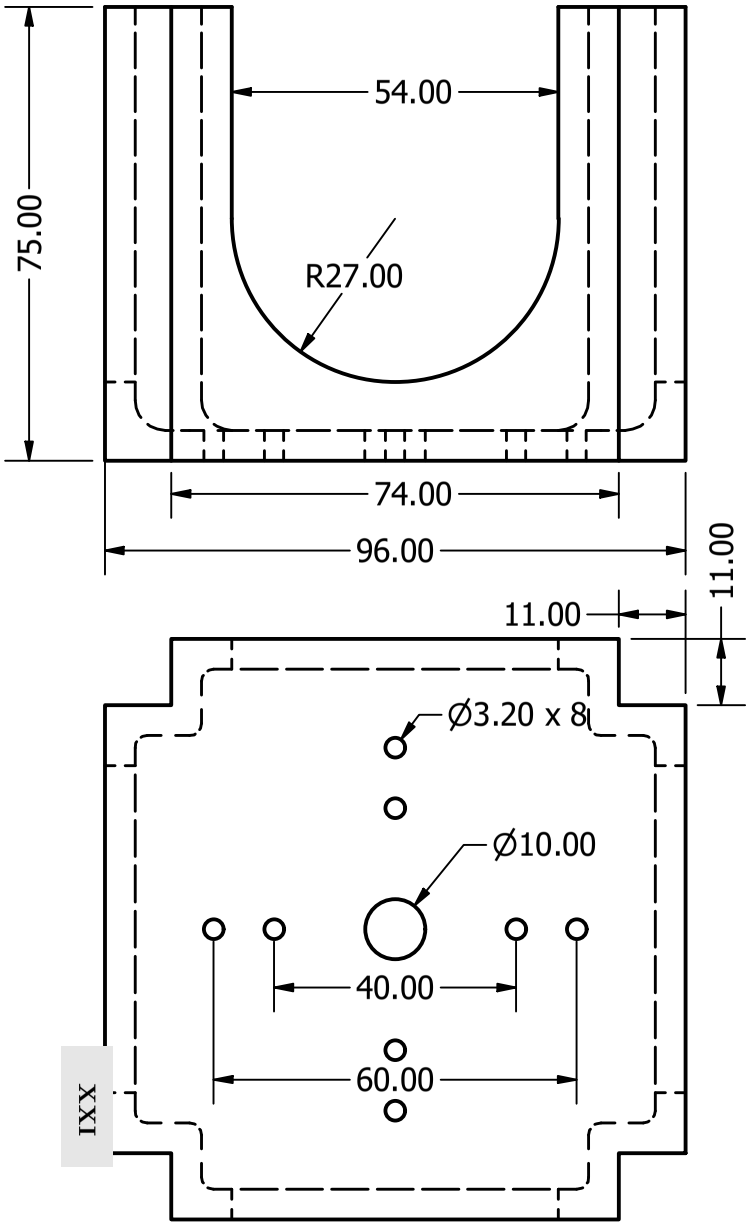
XIX

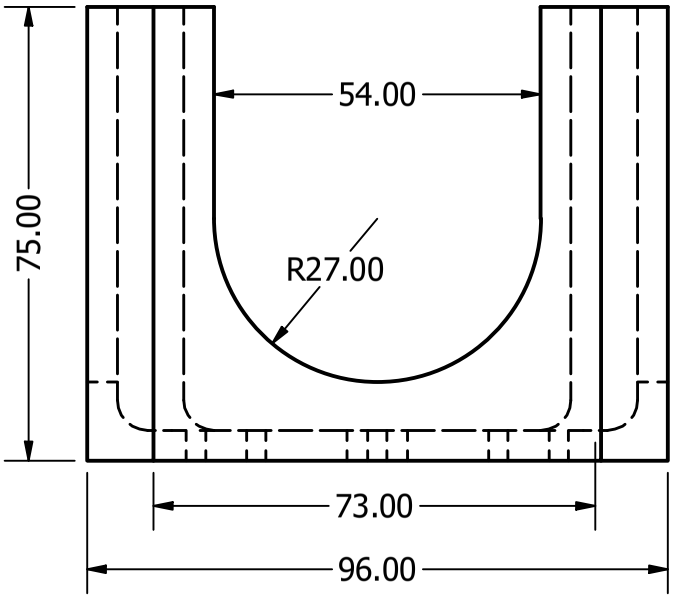
DRAWN Hampus	2019-04-15	EENX15-19-20
CHECKED		
QA		TITLE: Ben 2 högersida
MFG		
APPROVED		
		SIZE A4 | DWG NO | REV
		SCALE 0,4 : 1 | [mm] | SHEET 1 OF 1



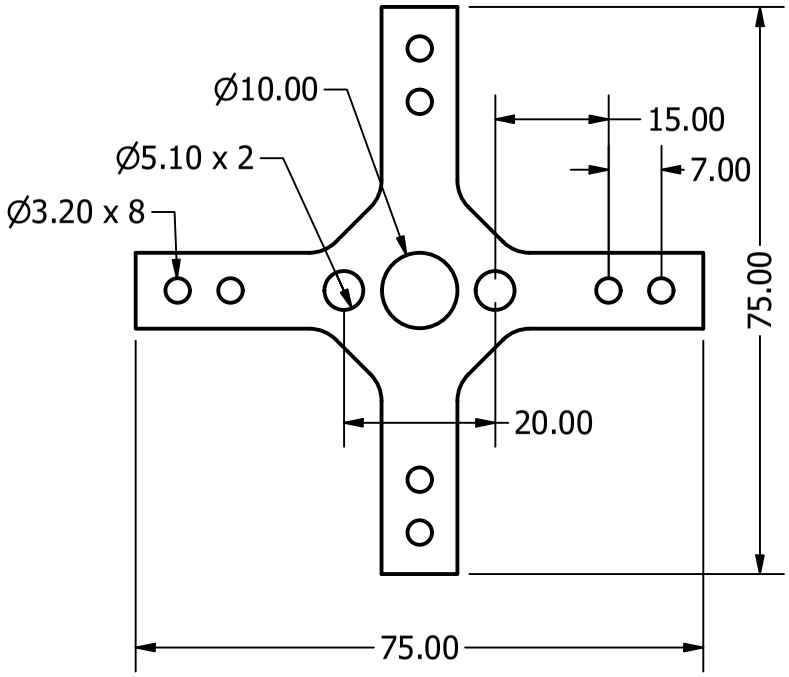
XX

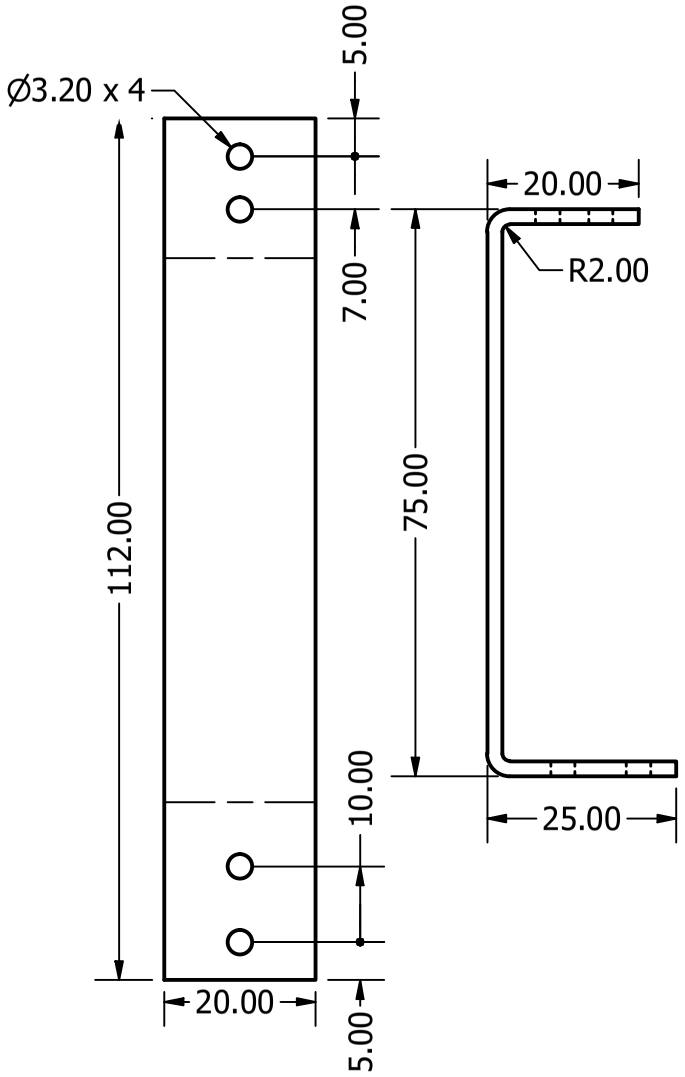
DRAWN Hampus	2019-04-15	EENX15-19-20
CHECKED		
QA		TITLE: Ben 3 högersida
MFG		
APPROVED		
		SIZE A4 | DWG NO | REV
		SCALE 0,4 : 1 | [mm] | SHEET 1 OF 1



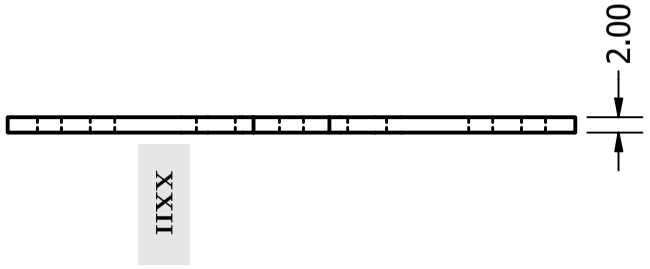


DRAWN Hampus	2019-03-05	EENX15-19-20		
CHECKED				
QA		TITLE Fot del 1		
MFG				
APPROVED				
		SIZE A4	DWG NO fot1	REV
		SCALE 0.8:1	[mm]	SHEET 1 OF 1



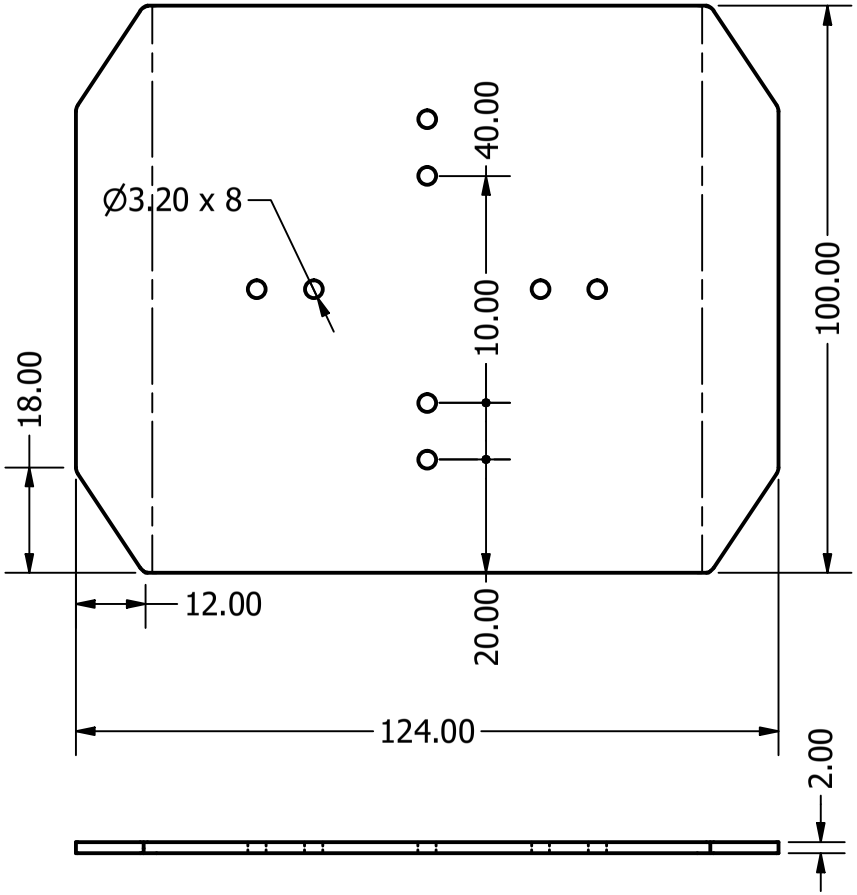


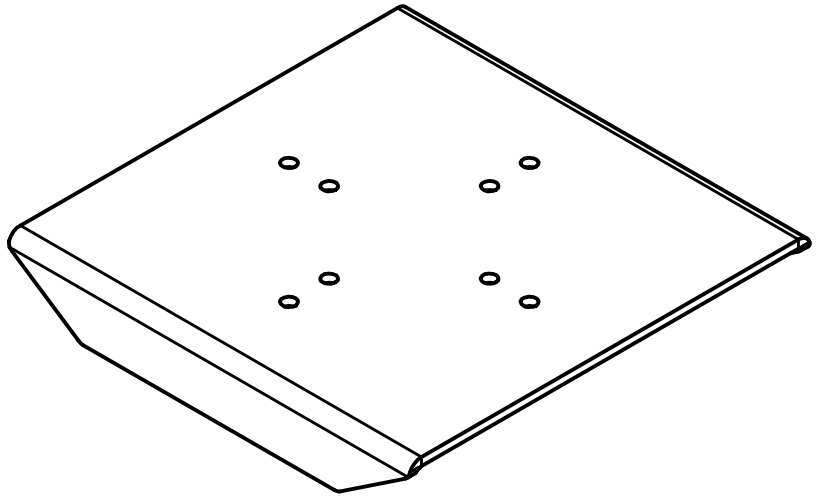




IIXXII

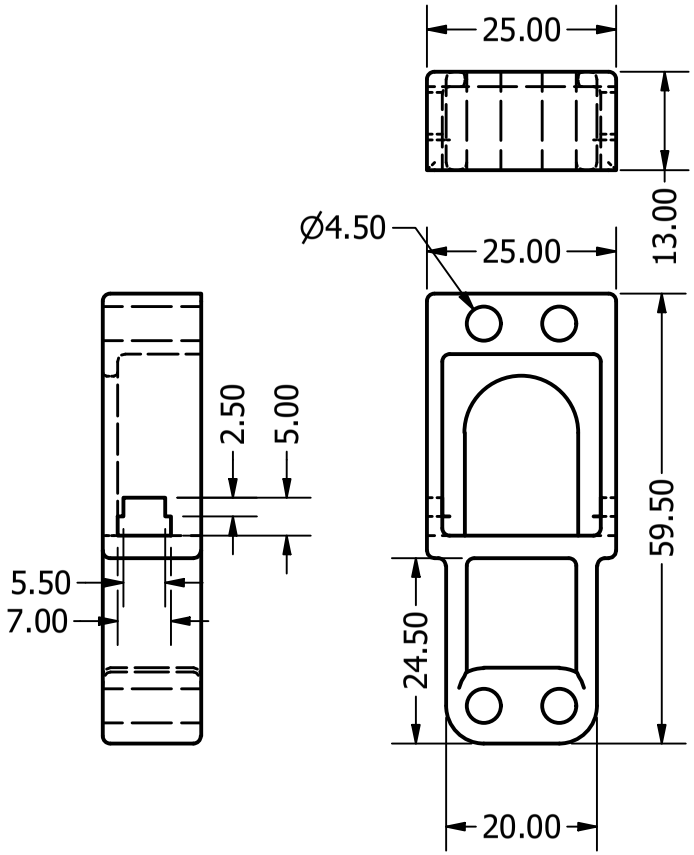
DRAWN Hampus	2019-03-05	EENX15-19-20		
CHECKED				
QA		TITLE		
MFG		Fot del 2 och 3		
APPROVED				
		SIZE A4	DWG NO fot2o3	REV
		SCALE 1 : 1	[mm]	SHEET 1 OF 1

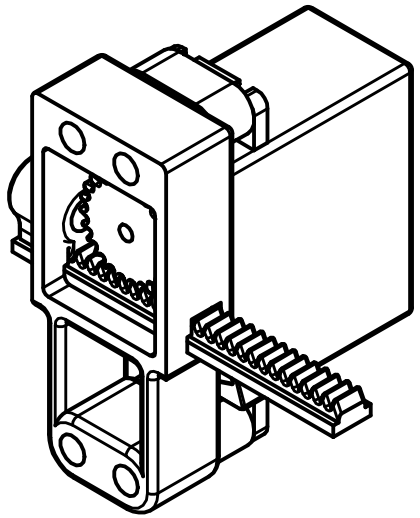


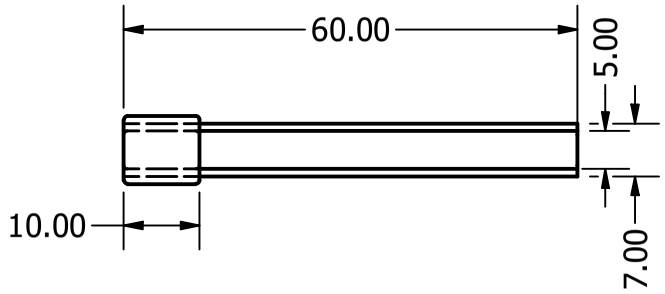


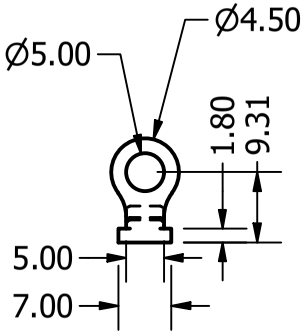
IIIXX

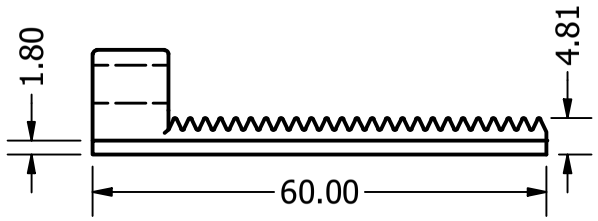
DRAWN Hampus	2019-03-05	EENX15-19-20		
CHECKED				
QA		TITLE		
MFG		Fot del 4		
APPROVED				
		SIZE A4	DWG NO fot4	REV
		SCALE 0.75	SHEET 1 OF 1	

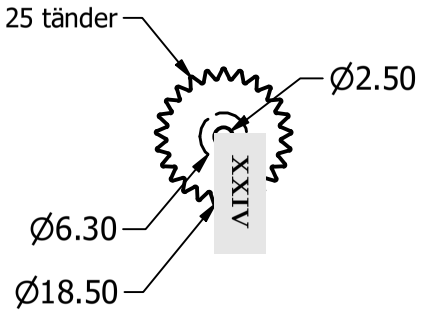


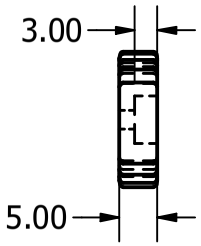




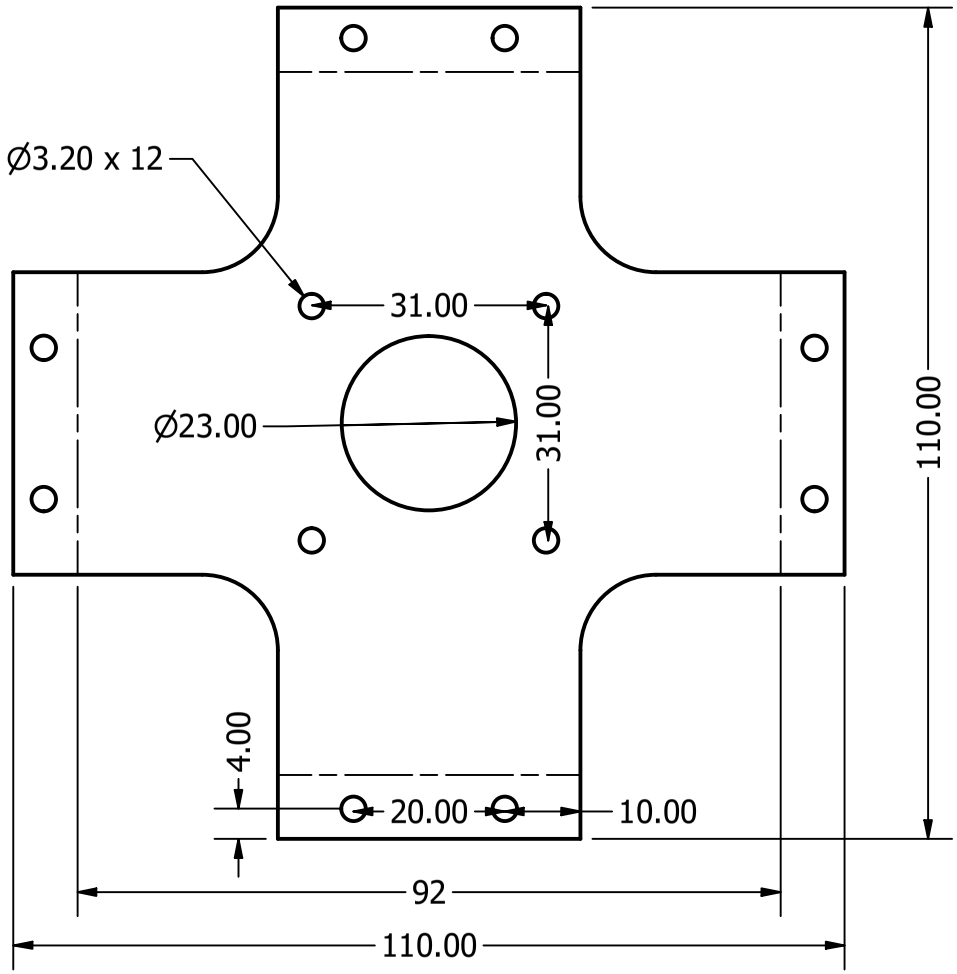


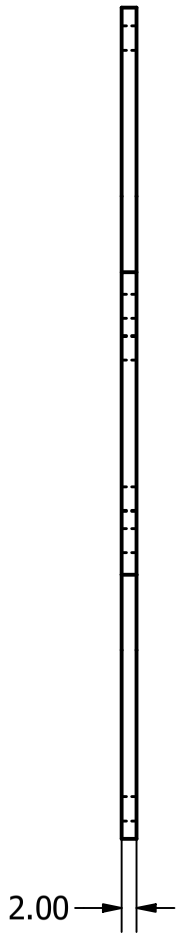


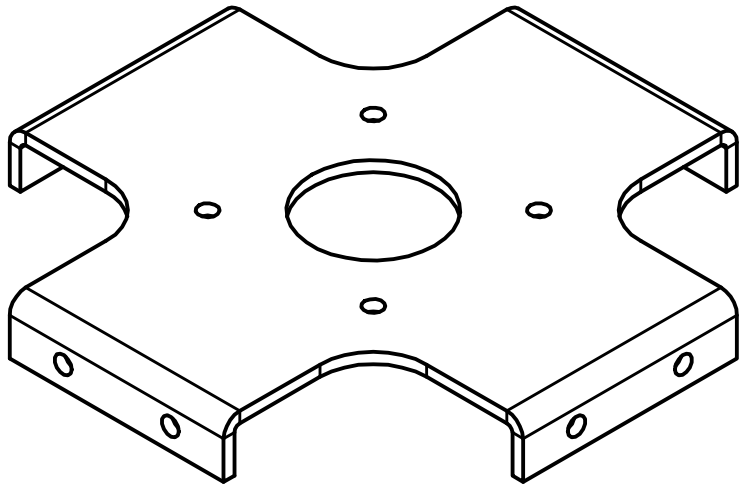




DRAWN Hampus	2019-03-05	EENX15-19-20		
CHECKED				
QA		TITLE		
MFG		Låsanordning		
APPROVED				
		SIZE A4	DWG NO motorsak2	REV
		SCALE 1 : 1	[mm]	SHEET 1 OF 1

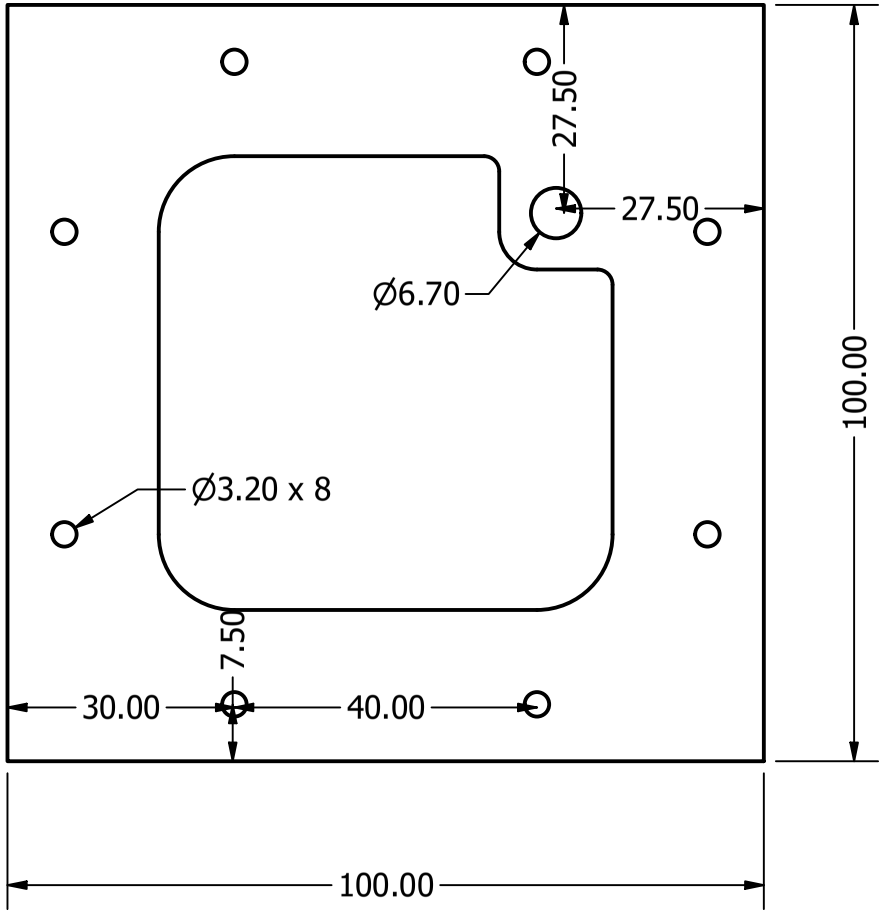


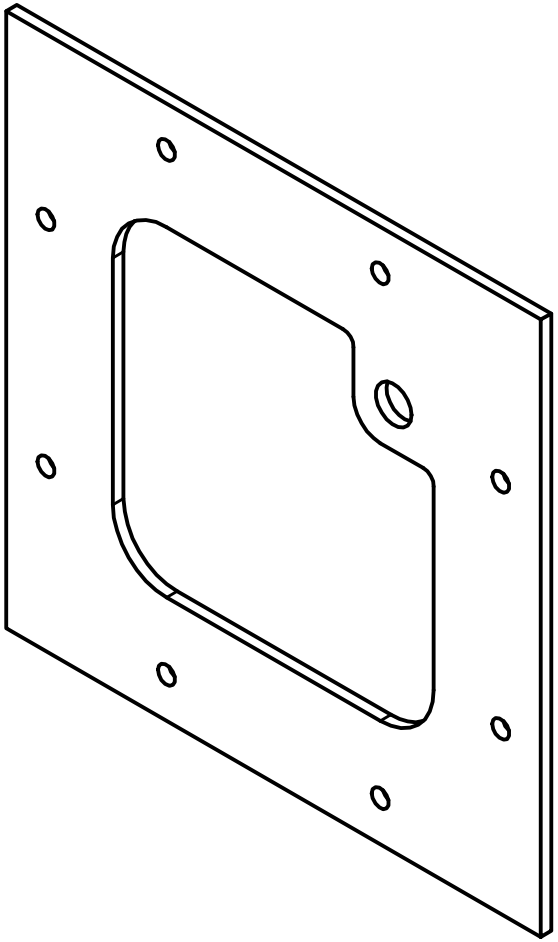


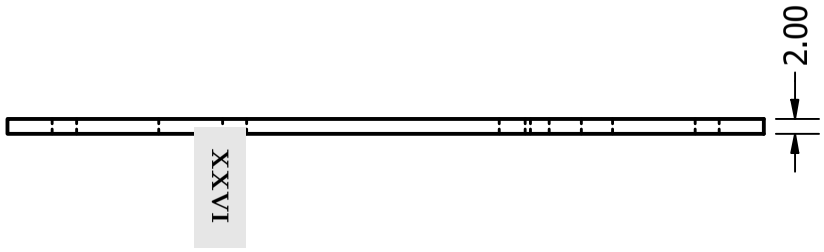


DRAWN Hampus	2019-03-21	EENX15-19-20
CHECKED		
QA		TITLE
MFG		Platå
APPROVED		
		SIZE A4 / DWG NO platå / REV
		SCALE 1 / [mm] / SHEET 1 OF 1

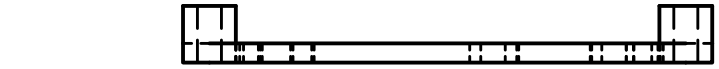
XXV

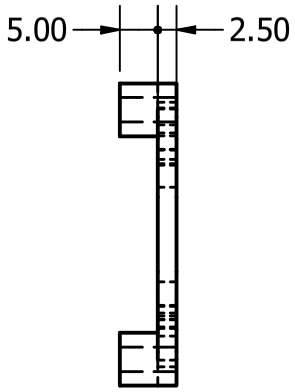


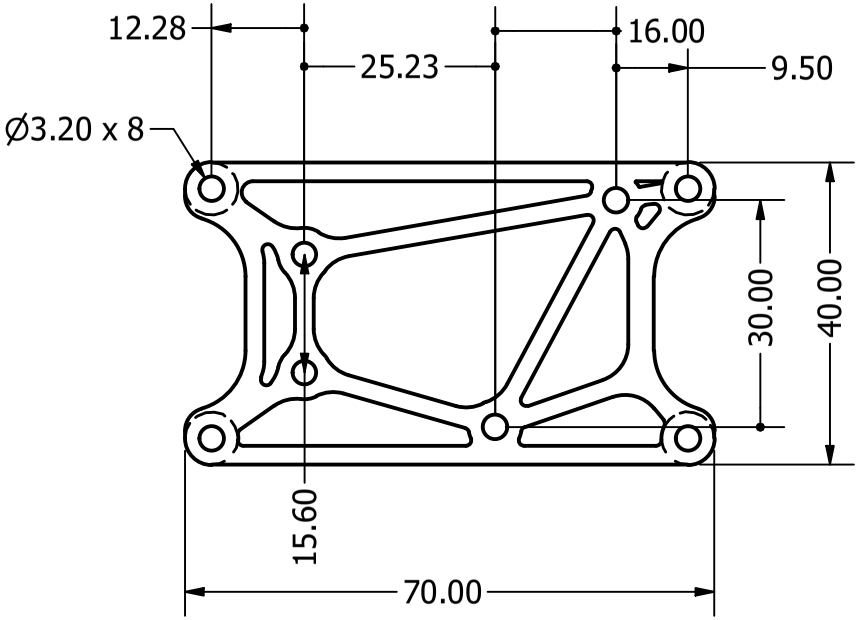


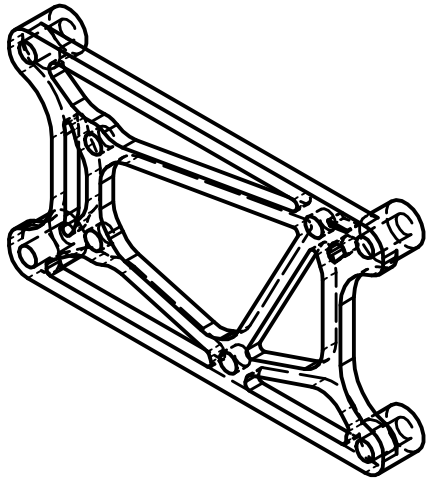


DRAWN Hampus	2019-03-14	EENX15-19-20		
CHECKED				
QA		TITLE Top		
MFG				
APPROVED		SIZE A4	DWG NO top	REV
		SCALE 1 : 1	[mm]	SHEET 1 OF 1

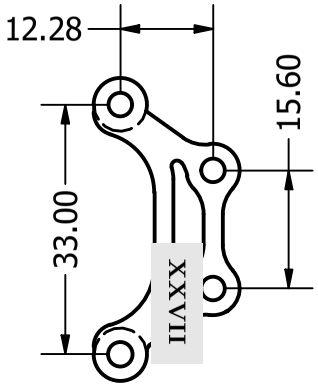




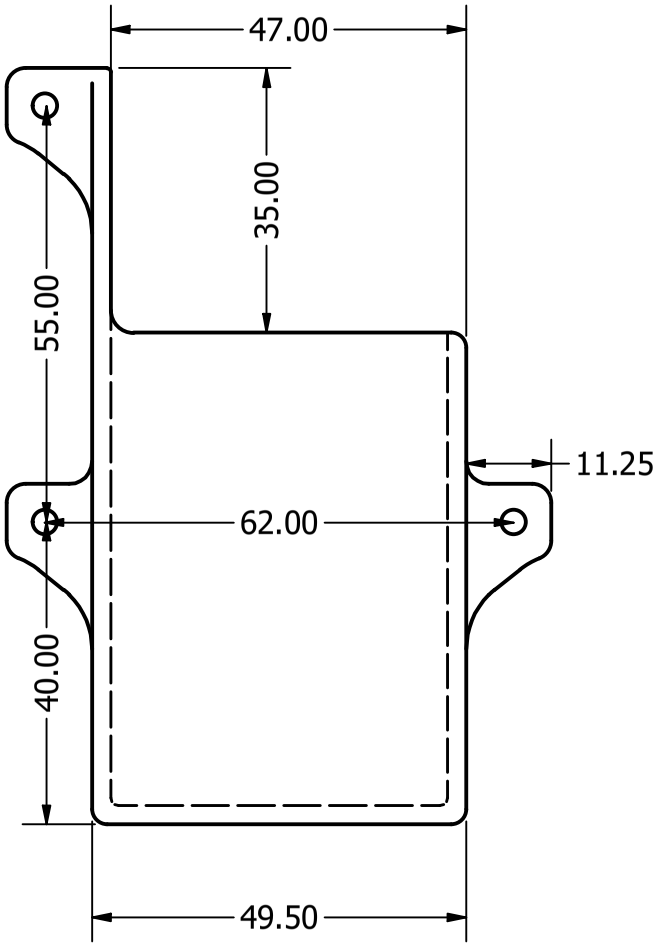


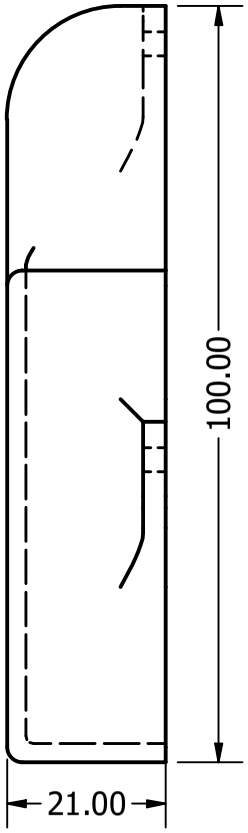


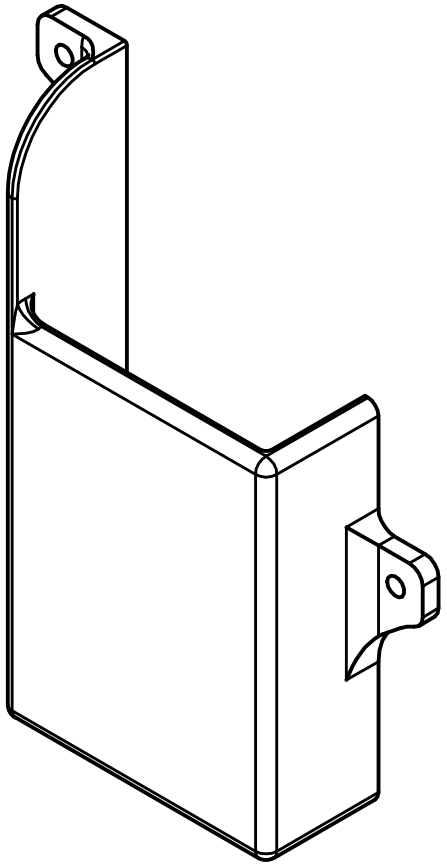


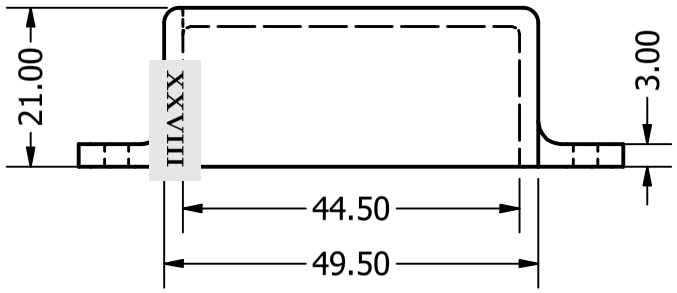


DRAWN Hampus	2019-03-05	EENX15-19-20		
CHECKED				
QA		TITLE		
MFG		Sensor fästen		
APPROVED				
		SIZE A4	DWG NO sensorhållare	REV
		SCALE 1 : 1	[mm]	SHEET 1 OF 1

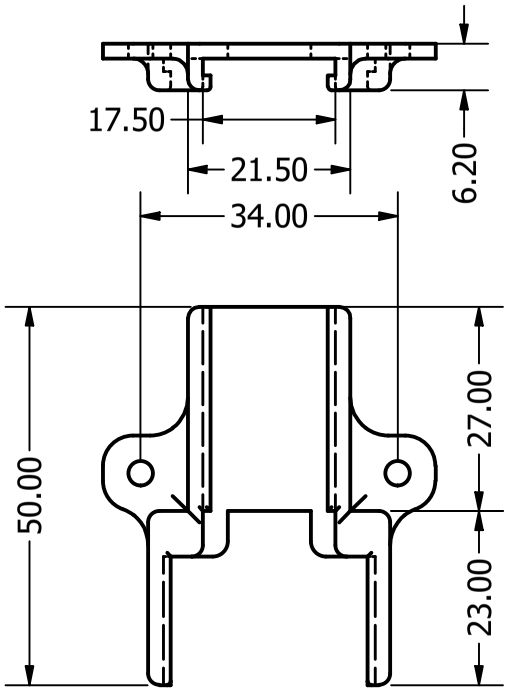


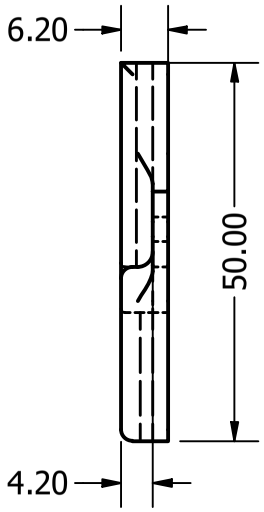


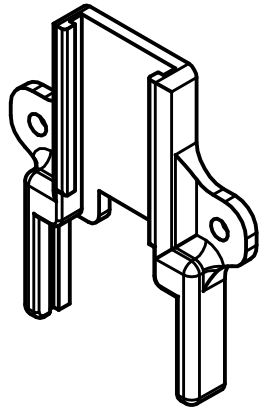


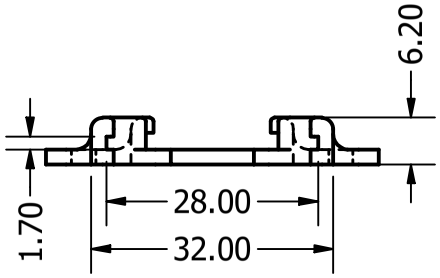


DRAWN Hampus	2019-04-15	EENX15-19-20		
CHECKED		TITLE Batterihållare		
QA				
MFG				
APPROVED				
		SIZE A4	DWG NO	REV
		SCALE 1	[mm]	SHEET 1 OF 1



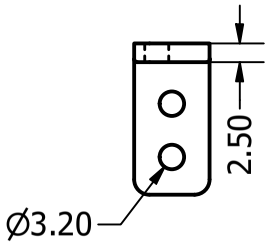


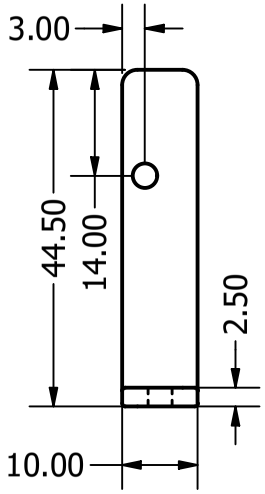


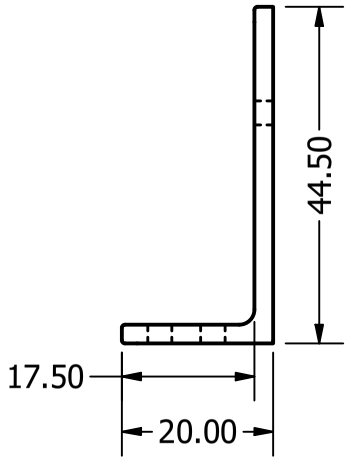


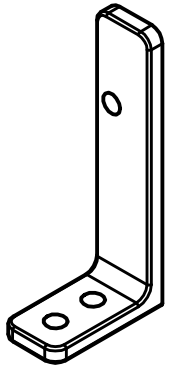
XIXX

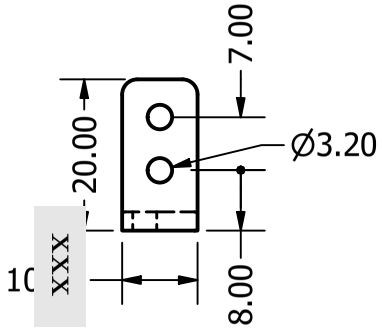
DRAWN Hampus	2019-04-15	EENX15-19-20		
CHECKED		TITLE Bluetoothhållare		
QA				
MFG				
APPROVED				
		SIZE A4	DWG NO hc-06 hållare	REV
		SCALE 1 : 1	[mm]	SHEET 1 OF 1







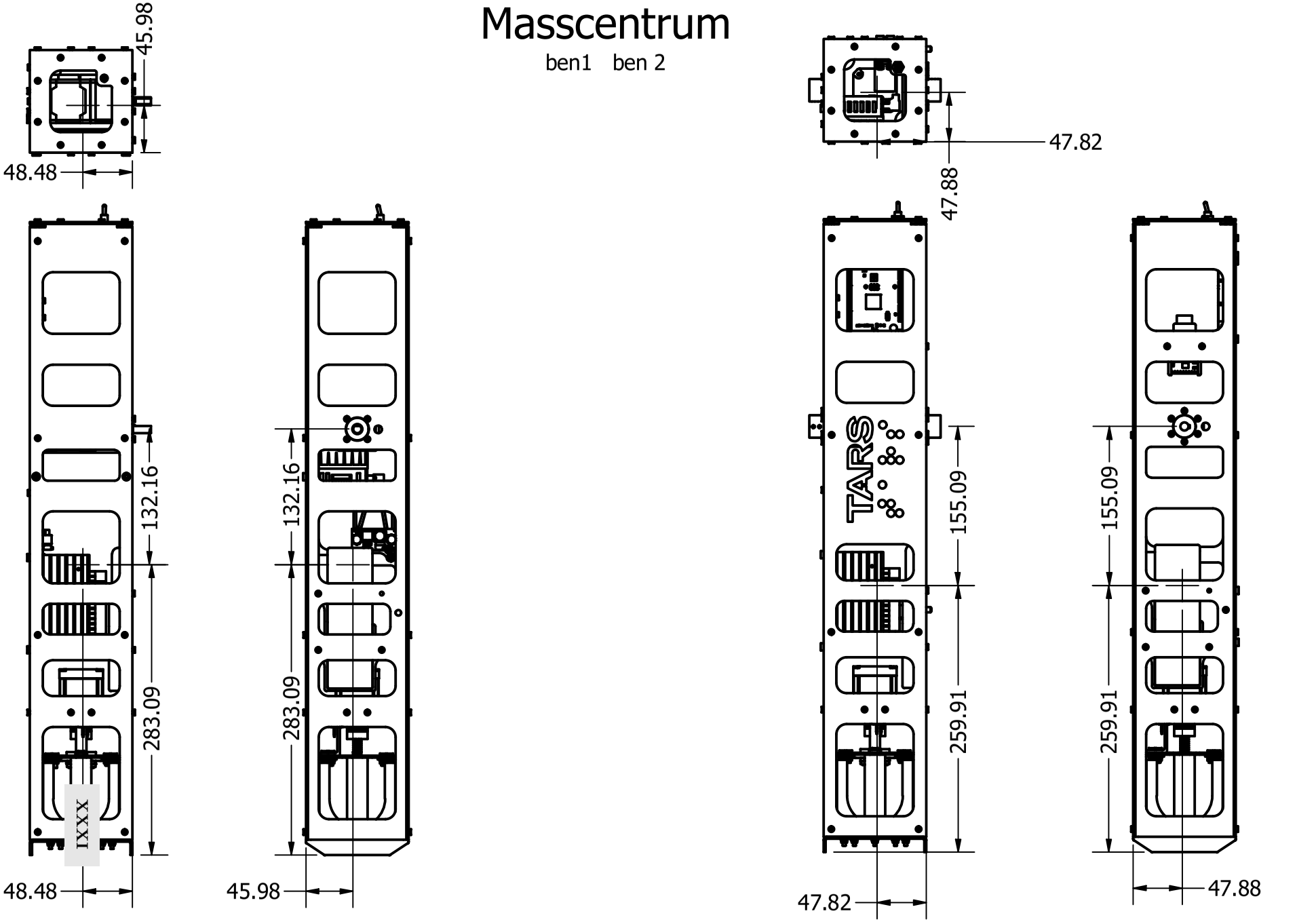




DRAWN Hampus	2019-04-16	EENX15-19-20		
CHECKED		TITLE Ändlägeshållare		
QA				
MFG				
APPROVED		SIZE A4	DWG NO	REV
		SCALE 1 : 1	[mm]	SHEET 1 OF 1

Masscentrum

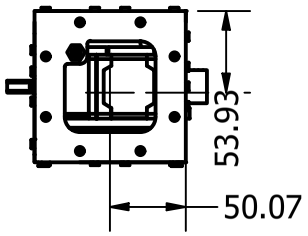
ben1 ben 2

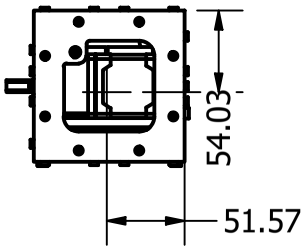


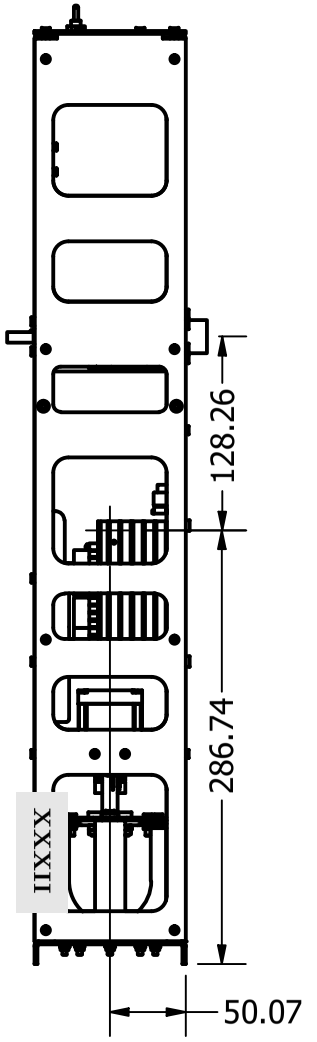
[mm]

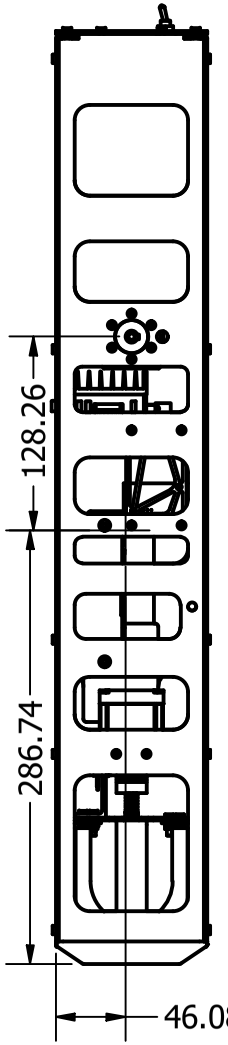
Masscentrum

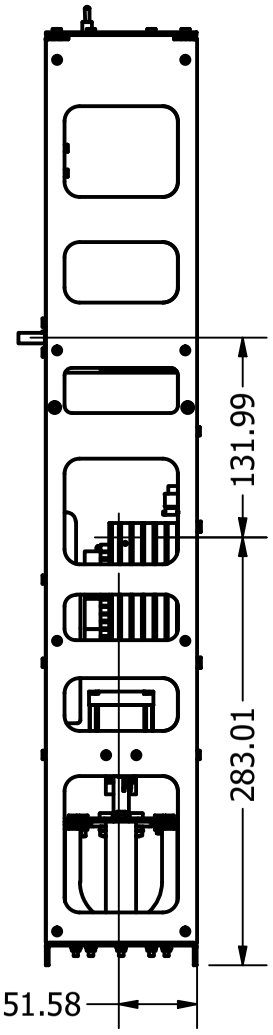
ben 3 ben 4

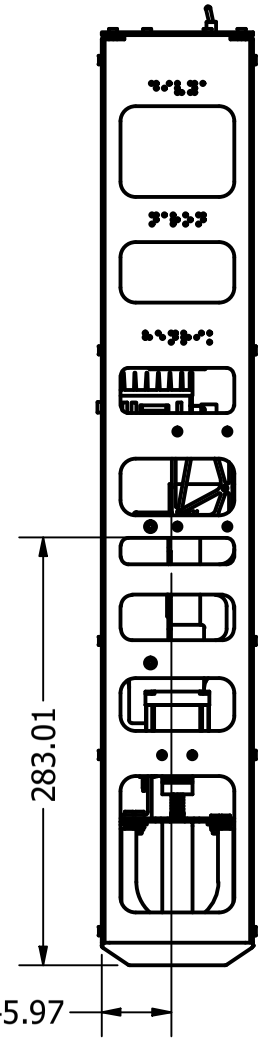






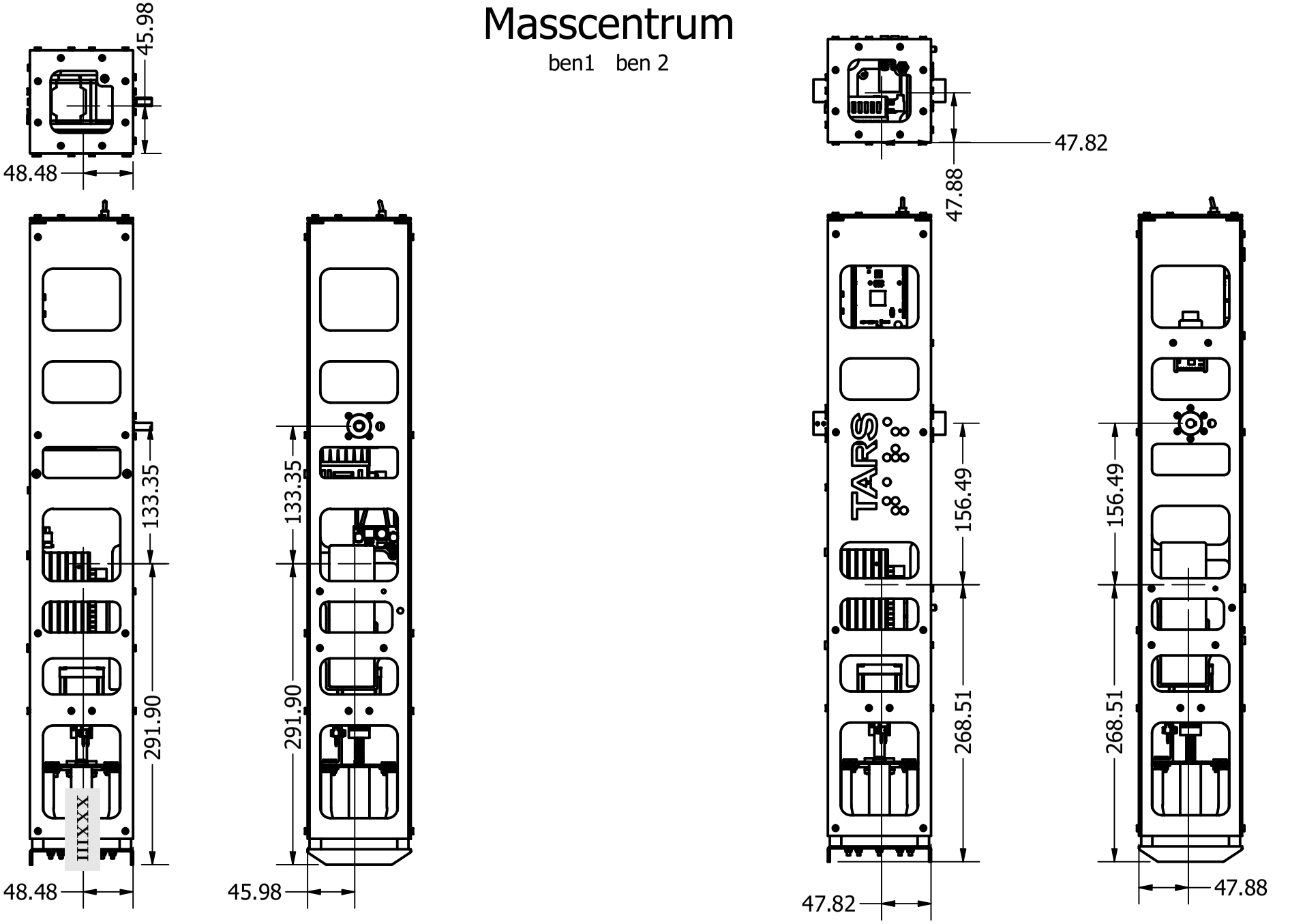






Masscentrum

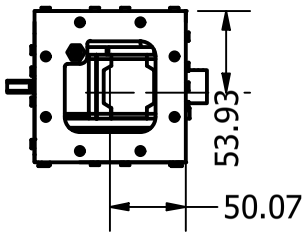
ben1 ben 2

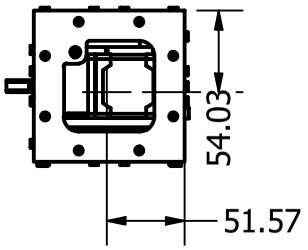


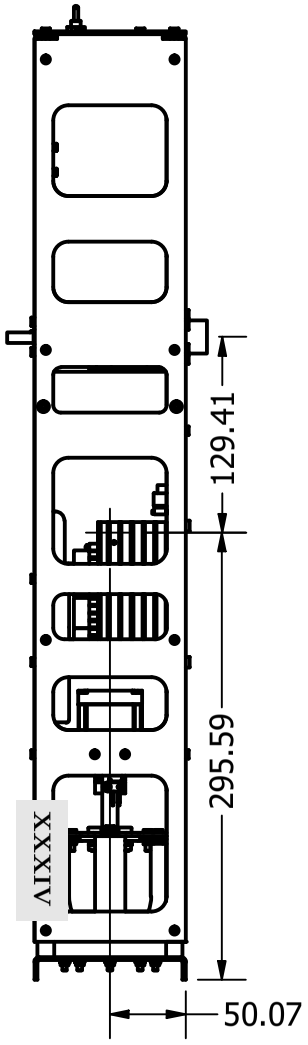
[mm]

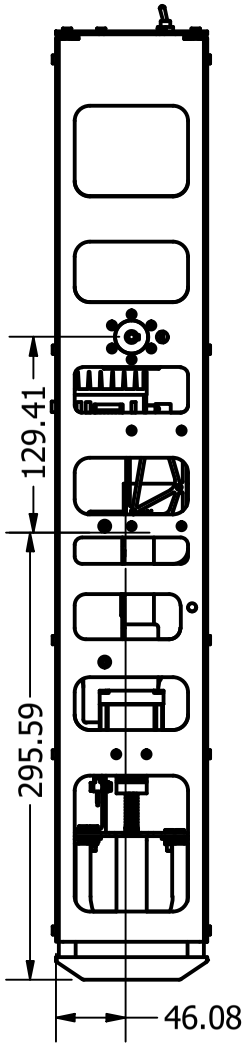
Masscentrum

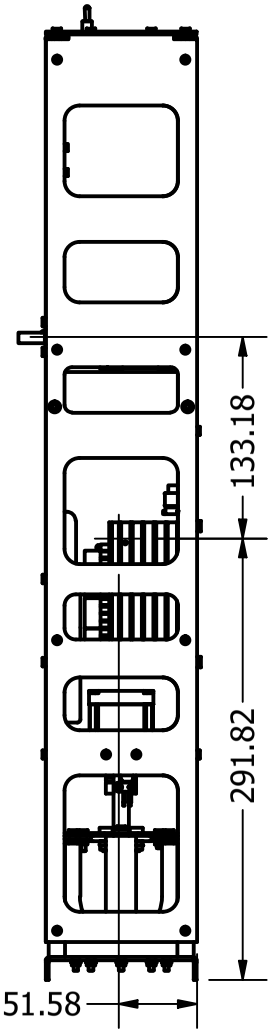
ben 3 ben 4

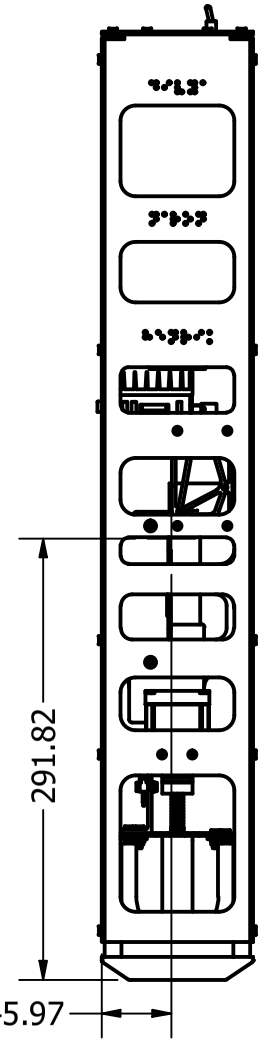




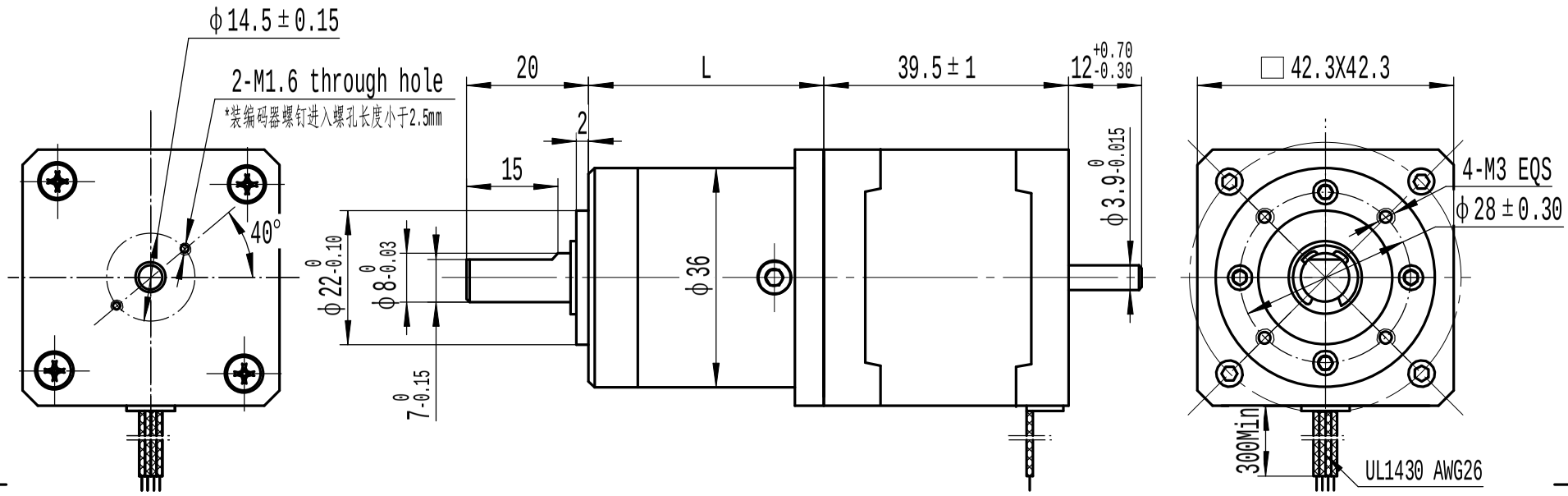








0DZ.242.001



Gear Box Specifications:

Gear Ratio	1:5.18	1:14	1:27	1:51	1:100
Number of gear trains	1	2		3	
Rated torque(kg.cm)	18	30		48	
Max.permissible load in a short time(kg.cm)	45	80		100	
Length (L) mm	31	38.5		48.2	

Stepper Motor Specifications:

MODEL	Phase	Step Angle	Rated Voltage	Current /Phase	Resistance /Phase	Inductance /Phase	Holding Torque	Rotor Inertia
		Deg/Step	V	A	Ω	mH	Kg.cm	g.cm²
42STH38-1684B	2	1.8	2.8	1.68	1.65	3.2	3.6	57

XXXV

REV.	QTY.	No.	SIG.	DATE
DESIGN	Zhu Guanghua	2013.09.09	STD.	
CHECK				
PROC.			APPR.	

CLS.MAK | QTY | SCALE | 1:1

PAGE OF

DONGZHENG MOTOR

36JXS60K/42STH38-1684B

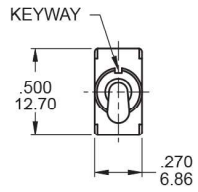
PLANET GEAR STEPPER MOTOR

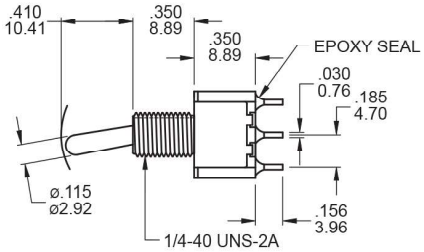
0DZ.242.001 REV

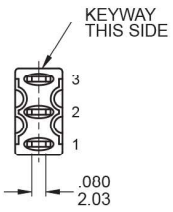
PDF 文件使用 "pdfFactory Pro" 试用版本创建 www.fineprint.cn

Pole Options:



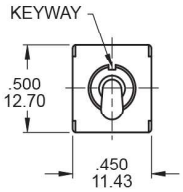


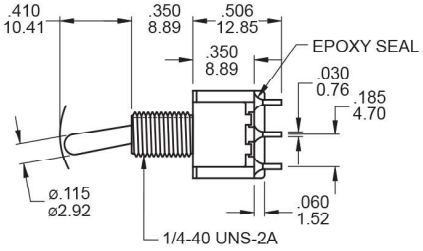


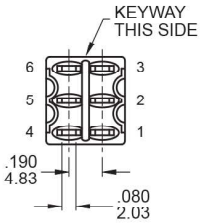


SPDT **Part No. Shown : 1MS1T1B1M1QE**



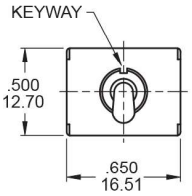


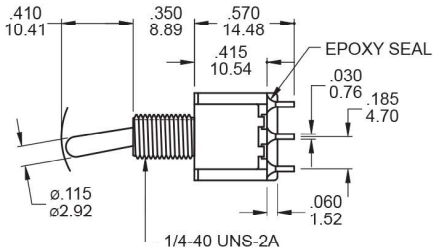


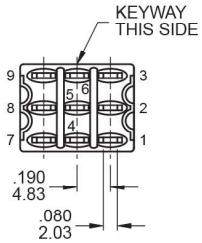


DPDT **Part No. Shown : 1MD1T1B1M1QE**



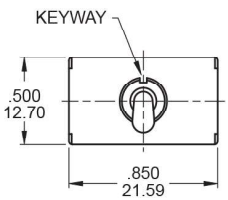


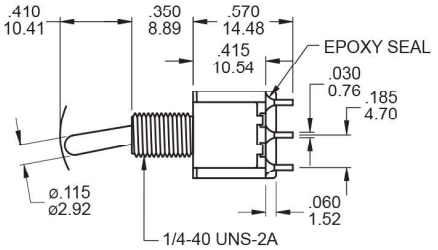


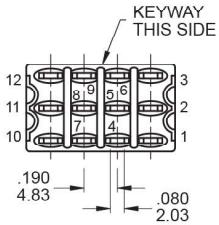


3PDT **Part No. Shown : 1M31T1B1M1QE**



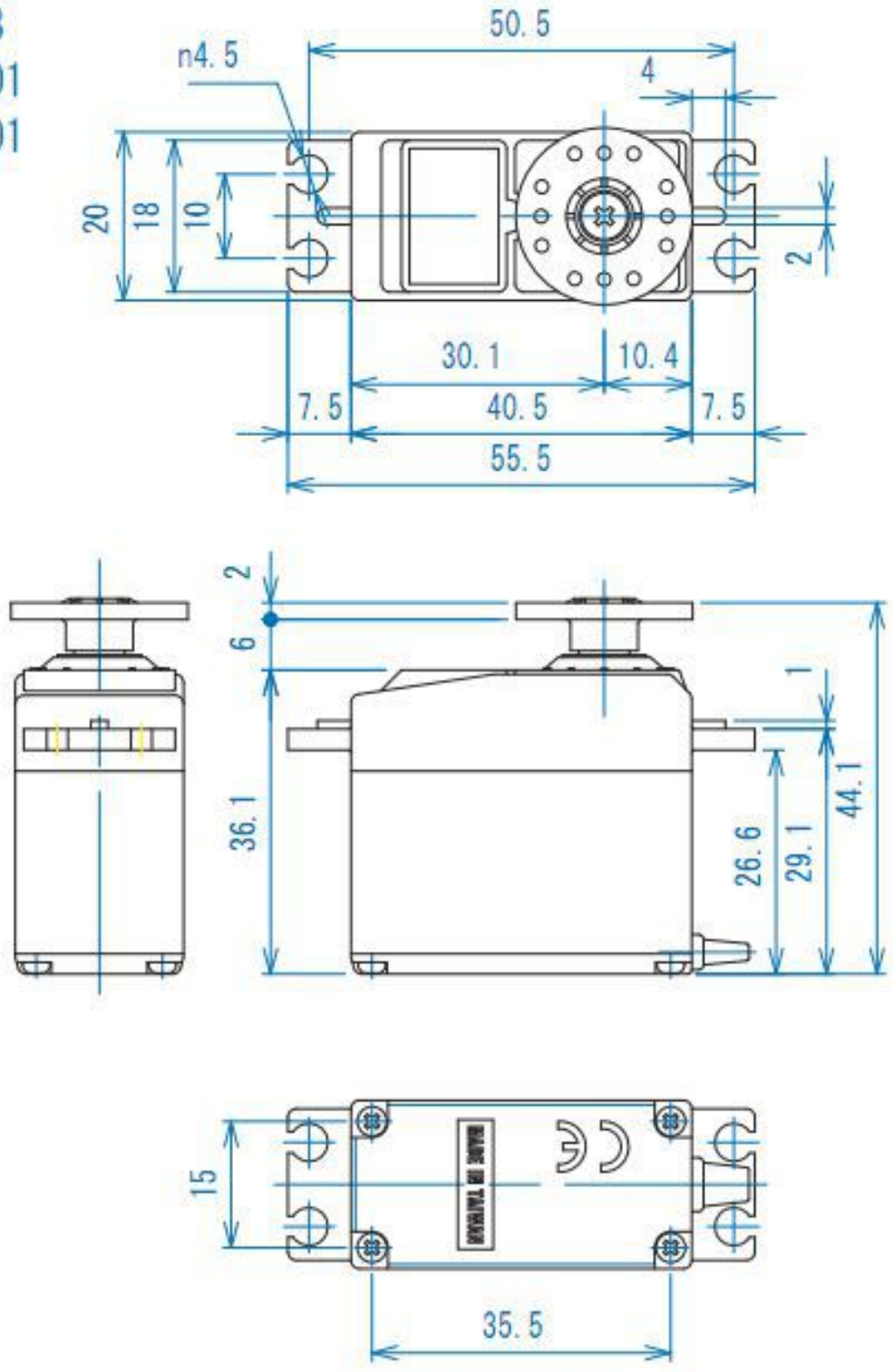






4PDT **Part No. Shown : 1M41T1B1M1QE**

S148
S3001
S9001



4 Mechanical dimensions

4.1 Lead wire configuration

Cable type	Coil	Function
Black	A	Motor coil A pin 1
Green	A-	Motor coil A pin 2
Red	B	Motor coil B pin 1
Blue	B-	Motor coil B pin 2

Table 4.1: Lead wire configuration

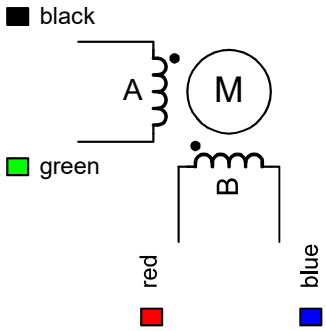
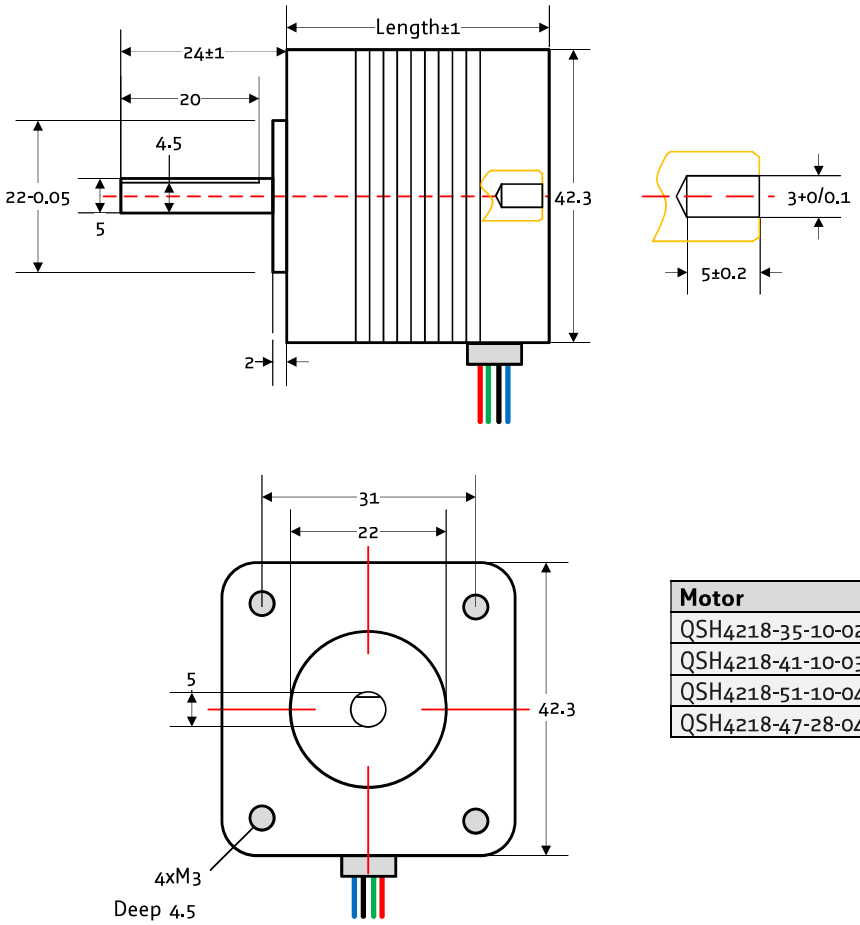


Figure 4.1: Lead wire configuration

4.2 Dimensions



Motor	Length
QSH4218-35-10-027	33.5mm
QSH4218-41-10-035	38mm
QSH4218-51-10-049	47mm
QSH4218-47-28-040	47mm

Figure 4.2: Dimensions (all values in mm)

Copyright © 2011, TRINAMIC Motion Control GmbH & Co. KG