

# Volvo CE

AMPR - Autonomous Machine Power Recharger

## **Authors:**

Joel Bergman  
Helgi Hrafn Björnsson  
Ludvig Boczar  
Rogelio Medina  
Jonathan Pernow  
Andreas Sjögren  
Tobias Sjölin  
Alexander Sundberg  
Victor Wolff

## **Supervisor:**

Tobias Vahlne, KTH Royal Institute of Technology

## **Examiner:**

Björn Möller, KTH Royal Institute of Technology

## **Stakeholders:**

Bobbie Frank, Volvo Construction Equipment  
Joakim Unneback, Volvo Construction Equipment

# Abstract

## AMPR

There is an increasing demand in electrification and automation for machines. Volvo Construction Equipment (Volvo CE) is part of this revolutionary change. They have created large autonomous electric haulers, whose task is to transport gravel or stones around a quarry. Large machinery requires a lot of energy, which has to be recharged somehow. This is where AMPR (Autonomous Machine Power Recharger) comes in. The solution before AMPR was a bulky charging station that had to be incorporated into the ground, thus almost impossible to move around the work site. The goal of AMPR was to develop an autonomous charging station for the haulers with more flexibility.

A camera is used to detect and track the charging port on the hauler parked in front of the charging station. This is done by training a neural network called *faster RCNN* with images of the port. The system uses a Cartesian coordinate system to measure distances from the camera to the port in horizontal, vertical, and outwards position. The camera is mounted right above the handle, which means they move together. The handle itself is built into a gimbal style assembly in order for it to be able to correct any deviations of the rotational position of the port. The handle is moved translationally via linear ballscrew actuators equipped with stepper motors in three directions. The coordinates obtained from the camera are sent to a micro-controller which moves the handle until the camera is facing the port dead on. Once this position is achieved, the plug on the handle will be moved into the port by using dead reckoning, meaning the handle is moved by a relative fixed offset from the cameras position dead on position, into the port. Once there has been a connection, the handle is locked in place during the charging sequence, once it is released the handle is retracted from the port and the hauler can drive away. This solution is based on a *State of the Art* analysis and requirements from the stakeholder.

The station had success in connecting the plug into to port. However, it was found that the correct lighting conditions were very important for the success rate, with the incorrect illumination the camera would not detect the port. Furthermore, the offset for the dead reckoning was not successful for all positions of the port. The handle could be maneuvered into the port while there were some rotational deviations, however these cannot be too big.

**Keywords:** AMPR, Autonomous, Charging, Cartesian robot, Pulley system, Electric vehicle, Recognition, Object detection



## Acknowledgements

We would like to thank Tobias Vahlne for being our coach for this project, helping us with ordering material and components to make this possible. We would also like to thank Staffan Qvarnström and Tomas Östberg for helping us with electronics and guiding us with our mechanical needs, respectively.

Furthermore, we would like to thank Pheonix Contacts for sponsoring us with a charging port and plug, and also Bobbie Frank and Joakim Unnebäck from Volvo CE.

Finally, we would like to thank Seshagopalan Thorapalli Muralidharan, who has given us a lot of great ideas and assisted us throughout the whole project.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Background . . . . .	1
1.2	Project Description . . . . .	2
1.3	Requirements . . . . .	2
1.4	Delimitations . . . . .	3
1.5	Readers guide . . . . .	3
<b>2</b>	<b>Literature Review and State of the Art</b>	<b>4</b>
2.1	Alignment . . . . .	4
2.1.1	Autonomous charging station prototype by BMW Group et. al. . . . .	4
2.1.2	Ericsson A.R.M.S . . . . .	5
2.1.3	Ultrasonic Phase Accordance method . . . . .	8
2.1.4	High Precision Ultrasonic Positioning Using a Robust Optimization Approach . . . . .	10
2.2	Computer vision . . . . .	11
2.2.1	HAAR cascade classifier . . . . .	11
2.2.2	CNN . . . . .	11
2.2.3	faster RCNN . . . . .	12
2.2.4	Geometry matching . . . . .	12
2.3	Electronics . . . . .	13
2.3.1	Encoder . . . . .	13
2.3.2	Potentiometer . . . . .	13
2.3.3	IR sensor . . . . .	13
2.3.4	IR Depth Cameras . . . . .	14
2.3.5	Ultrasonic transducers . . . . .	15
2.3.6	Microcontrollers . . . . .	15
2.4	Actuators . . . . .	16
2.4.1	DC-motor . . . . .	16
2.4.2	Stepper motor . . . . .	16
2.4.3	Servo motor . . . . .	17
2.4.4	Linear actuator . . . . .	17
2.5	Software . . . . .	17
2.5.1	ROS . . . . .	17
2.5.2	OpenCv . . . . .	17
2.5.3	Tensorflow . . . . .	18
2.5.4	Intel OpenVino . . . . .	18
<b>3</b>	<b>Methodology</b>	<b>19</b>
3.1	Project management/Organization . . . . .	19
3.2	Engineering approaches . . . . .	20
3.3	Development Process . . . . .	20
<b>4</b>	<b>Implementation</b>	<b>21</b>
4.1	Concepts Ideas for Station and Alignment . . . . .	21
4.1.1	Charging station . . . . .	21
4.1.2	Cartesian charging station . . . . .	21

4.1.3	Robot arm and housing . . . . .	22
4.1.4	Motion . . . . .	23
4.1.5	Handle . . . . .	24
4.1.6	Single IR Camera with tape . . . . .	27
4.1.7	Single IR Camera with diodes . . . . .	27
4.1.8	Intel IR depth camera . . . . .	27
4.1.9	Ultrasonic and a single camera . . . . .	27
4.1.10	Mechanical guide . . . . .	27
4.1.11	Charging port . . . . .	29
4.1.12	Concept evaluation . . . . .	29
4.2	Final Design . . . . .	33
4.2.1	Mechanical Design . . . . .	33
4.2.2	Charging cable . . . . .	33
4.2.3	Motor setup . . . . .	34
4.2.4	Handle solution . . . . .	37
4.2.5	Carrier . . . . .	40
4.2.6	Station . . . . .	43
4.3	Electrical Design . . . . .	44
4.3.1	Station . . . . .	44
4.3.2	Carrier . . . . .	45
4.4	Software Design . . . . .	47
4.4.1	Station . . . . .	47
4.4.2	Hauler . . . . .	48
4.5	Recognition . . . . .	49
4.6	Code Optimization . . . . .	50
<b>5</b>	<b>Verification and Validation</b>	<b>52</b>
<b>6</b>	<b>Results</b>	<b>53</b>
6.1	Requirements . . . . .	53
6.1.1	Stakeholder requirements . . . . .	53
6.1.2	Technical requirements . . . . .	54
6.2	Testing . . . . .	54
6.2.1	Image Recognition . . . . .	54
6.2.2	Lighting . . . . .	55
6.2.3	Connector Angle Deviation . . . . .	56
6.2.4	Accuracy . . . . .	57
<b>7</b>	<b>Discussions and Conclusions</b>	<b>58</b>
<b>8</b>	<b>Future Work</b>	<b>59</b>
8.1	Plug . . . . .	59
8.2	Construction of station . . . . .	59
8.3	Camera . . . . .	59
8.4	Neural network . . . . .	60
8.5	Micro-controllers . . . . .	60
8.6	Implementation for Other Vehicles . . . . .	60

<b>9</b>	<b>Appendix</b>	<b>65</b>
9.1	Appendix A - Code Hauler in C++ . . . . .	65
9.2	Appendix B - Main Node Code . . . . .	66

# List of Figures

1	HX02 - the autonomous load carrier [23] . . . . .	1
2	Inverted pantograph charging solution currently used by Volvo [19] . . . . .	1
3	Left: Overall picture of the prototype. Right: Detailed picture of the handle [3].	4
4	Digital unit with inserted SFP as well as fiber cables [44]. . . . .	6
5	The total automated repair and maintenance system [1]. . . . .	6
6	The resulting forces and torques when in contact with the digital unit [1]. . . . .	7
7	Left: SFP connector image before processing. Right: SFP connection image after processing [1]. . . . .	7
8	Picture describing the parameters $\rho$ and $\theta$ [4]. . . . .	8
9	The resulting four lines illustrating the detected SFP [1]. . . . .	8
10	Carrier waves constituting the sync pattern. [21] . . . . .	9
11	TOA schematic [21] . . . . .	9
12	US receiver grid and transmitter positions illustrated [30] . . . . .	10
13	Advantageous situations for geometric matching [12]. . . . .	12
14	Results of the template matching [33]. . . . .	13
15	Intel D435 Camera . . . . .	14
16	Stereo Images from the Intel D435 Camera [13] . . . . .	15
17	Point Cloud from the Intel D435 Camera [13] . . . . .	15
18	An ultrasonic transducer [14] . . . . .	15
19	Cartesian robot [10]. . . . .	21
20	Cartesian charging station motor setup. Made in Solid Edge. . . . .	22
21	Cartesian charging station without handle. Made in Solid Edge. . . . .	22
22	A picture of a robot called Carla_connect, that has both linear and rotational movements [35]. . . . .	23
23	A picture of a linear ball screw actuator. Made in Solid Edge. . . . .	23
24	A picture that illustrates where the rotational movements are located. Made in Solid Edge. . . . .	24
25	Servo handle with linear actuator. Made in Solid Edge. . . . .	25
26	Handle with spring mechanism. Made in Solid Edge. . . . .	26
27	Side view. . . . .	28
28	Top view. . . . .	28
29	Charging port with guide, mounted on carrier. Made in Solid Edge. . . . .	28
30	Charging port with lid open. Marker for visual system to recognize. Made in Solid Edge. . . . .	28
31	Example of types of connectors available [8]. . . . .	29
32	Inside of the charging station. Made in Solid Edge. . . . .	31
33	Complete design of the system. Made in Solid Edge. . . . .	31
34	Complete design of the system. Charging station and lid made in Solid Edge. Carrier is from Volvo [23]. . . . .	32
35	Left:AMPR seen from the front. Right: AMPR seen from a sideways rear view.	33
36	Picture showing the attachment for cable clamping. . . . .	34
37	Picture showing the rear attachment for cable clamping. . . . .	34
38	Left:The motor setup. Right: The pulley system and counterweight . . . . .	35
39	The steel plate and the extra guide rail . . . . .	36
40	The outwards-going actuator with aluminium pipes attached. . . . .	36
41	A gimbal system [11]. . . . .	37

42	The simplified carrier given by Volvo CE. . . . .	38
43	Left: Side view of the port and handle assembly. Right: Behind view of port and handle . . . . .	38
44	Spring based solution. Springs not included. . . . .	39
45	The full-scale handle concept with springs. . . . .	39
46	The carrier replica. Made in Solid Edge. . . . .	40
47	Left: Side view of the carrier replica. Right: Behind view of the carrier replica. Made in Solid Edge. . . . .	41
48	Left: A sub assembly of the carrier replica including the port and circular back plate etc. Right: A sub assembly showing the guide plates and screws used for clamping. Made in Solid Edge. . . . .	42
49	The port solution rotated (roll). Made in Solid Edge. . . . .	42
50	Left: The joint specifications. Right: Rotated view of the port solution (Pitch). Made in Solid Edge. . . . .	43
51	The full scale prototype. . . . .	43
52	The charging station. . . . .	44
53	Planned stepper-driver protection circuit . . . . .	45
54	Sketch of the main PCB, Made in Eagle . . . . .	46
55	Sketch of the LED PCB, Made in Eagle . . . . .	46
56	Flowchart of the code for the hauler, Made in Drawio . . . . .	48
57	Camera view with recognition of the port. . . . .	51
58	Port detection tests with different lighting. . . . .	56

## List of Tables

1	The resulting evaluation table . . . . .	30
2	The resulting evaluation table for the mechanical components. . . . .	30
3	The distribution of images for each classification . . . . .	49
4	Table declaring whether stakeholder requirements were met . . . . .	53
5	Table declaring whether the technical requirements were met. . . . .	54
6	Table presenting the results when investigating the angular deviation capabilities of the camera detection . . . . .	55
7	Table presenting the results when investigating the angular deviation capabilities of the connector solution . . . . .	57
8	Table presenting the results when investigating the accuracy of a connection when the port is facing straight forward . . . . .	57

# Nomenclature

## List of abbreviations

AI	Artificial Intelligence
CCS	Combined Charging System
CNN	Convolutional Neural Network
CPU	Central Processing Unit
DC	Direct Current
GFCI	Ground-fault circuit interrupter
I/O	Input-output
IC	integrated circuit
LED	Light Emitting Diode
MCU	Microcontroller unit
PCB	Printed Circuit Board
PSU	Power Supply Unit
RCNN	Region based Convolutional Neural Network
ROS	Robot Operating System
RX	Receiving
SR	Stakeholder requirement
TOA	Time of arrival
TR	Technical requirement
TX	Transmitting
USB	Universal Serial Bus
VPU	Vision Processing Unit
Volvo CE	Volvo Construction Equipment
YOLO	You Only Look Once



# 1 Introduction

This chapter will describe the project, its background scope, and stakeholder requirements.

## 1.1 Background

Volvo CE, a construction equipment company, are moving towards more electrified and automated solutions to increase safety and radically reduce the environmental impact [20]. In 2017 Volvo unveiled the autonomous load carrier HX02 [23], see Figure 1.



Figure 1: HX02 - the autonomous load carrier [23]

The carriers' mission is, for instance, to electrify and automate the transport state in a quarry [23]. The carriers have a work cycle of only 7 minutes after which drive to charge [18]. Volvo are currently using an inverted pantograph charging solution supplying energy from below [19], see Figure 2.



Figure 2: Inverted pantograph charging solution currently used by Volvo [19]

This solution is however remarkably large in size and specialized for only the HX02 to a great extent. The interest in a smaller, nonexclusive and automated charging solution is thus a fact.

## 1.2 Project Description

A robust charging solution was researched and developed to help electrify heavy-industry equipment. As has been mentioned before, the existing solution is bulky and not easily transportable, making it ill-suited for the equipment it is meant to service, considering the required time and money for transportation and setup while moving between sites. The charging solution should be able to service the HX02 carrier in the field with a possibility to service various vehicles since it is the idea to have an interchangeable charging connector.

The goal of this is to design and construct a proof-of-concept easily transportable charging solution which fulfills the stakeholders' requirements.

## 1.3 Requirements

To solve the problems presented in the section above, requirements were devised. The following section describes the stakeholder as well as the technical requirements for this project.

### Stakeholder requirements (SR)

- SR1: The structure shall manage to operate in harsh weather conditions and harsh environments
- SR2: The charger plug should be based on an existing connector
- SR3: Project cost should be minimized
- SR4: A charging port should be positioned on the HX02 hauler
- SR5: The structure shall have support for a charger with a power level of at least 150  $kW$  (for 650  $V$  system)
- SR6: The structure shall be able to charge a slightly misaligned HX02
- SR7: The structure shall be able to align to the port enough create a solid connection
- SR8: The station shall be able to communicate with the HX02
- SR9: Aesthetics shall be taken into consideration when designing the structure
- SR10: The connector shall be interchangeable

### Technical requirements (TR)

The charging station:

- TR1: Shall be able to carry the weight of the cable + own weight + safety margin
- TR2: Shall be able to extend at least 500 mm from its starting position
- TR3: Shall be able to move in a minimum range of 200 mm horizontally
- TR4: Shall be able to move in a minimum range of 100 mm vertically
- TR5: Shall be able to handle a 3° deviation around all axes

- TR6: Shall not take up more than  $2.5\text{ m}^3$  and not weigh more than 300 kg
- TR7: Shall have a capability to charge up to 150 kW
- TR8: Shall be able to sense the port from a distance of 600 mm and through iterations, obtain its coordinates within  $\pm 10\text{ mm}$
- TR9: Shall be equipped with at least one camera
- TR10: Shall be able to communicate with the main server host
- TR11: Shall have protection, at least of standard IP13 with possibility for improvement

The carrier:

- TR12: Shall have a port placed on the carrier right/left side
- TR13: Shall have protection for the charging port. At least of standard IP55 with possibility for improvement

## 1.4 Delimitations

The focus of this project is the construction of an autonomous charger solution feasible for the task at hand. The influence of atmospheric conditions like snow, rain, hail, wind are left out in the scope of this project.

## 1.5 Readers guide

After the introduction, in Chapter 2 called "Literature Review and State of the Art" the report starts to investigate the theory that is necessary for this project. This Chapter begins with examining already existing solutions. Afterwards, the main theory to building an autonomous charging solution is presented.

The next chapter is called "Methodology". In this Chapter, one can read the engineering approaches the group has used within this project. The explanation of how the group worked and what kind of subgroups were created, to work as efficiently as possible are also explained in this chapter.

Chapter 4 called "Implementation" starts with presenting a couple of concepts ideas for each main task. All concept ideas are evaluated and one final solution per main task is decided. From the solutions a final design of the station is presented. The final design is presented in detail, including mechanical-, electrical-, software- and recognition-design.

This is followed by Chapter 5 and is called "Validation and verification". This Chapter explains what kind of tests were carried out on the station. The tests were executed and the results are presented in Chapter 6, called "Results". In Chapter 7, "Discussions and Conclusion" the results are discussed. The discussion continues in the next chapter, Chapter 8, "Future Work". Were a discussion about possible improvements to the station is made, to make it more efficient, reliable and robust.

## 2 Literature Review and State of the Art

This chapter thoroughly describes similar projects solving the automated and transportable charging problem or any portion of it.

### 2.1 Alignment

The projects leading challenge is the alignment. Which includes detecting the charging port, determining its position and enabling motion towards it.

#### 2.1.1 Autonomous charging station prototype by BMW Group et. al.

An automated charging station prototype was in 2019 developed by B.Walzel and H.Brunner et. al. [3]. The prototype was invented for the electric automotive industry to manage autonomous vehicles as well as the demand of fast charging which would result in a heavy and unhandy cable [3].

The charging station in question is equipped with three mono cameras, see Figure 3. One camera is placed on the robotic arm, represented by the number 11 in the figure, and the two remaining cameras are placed on the station [3], represented by the number four in the figure.

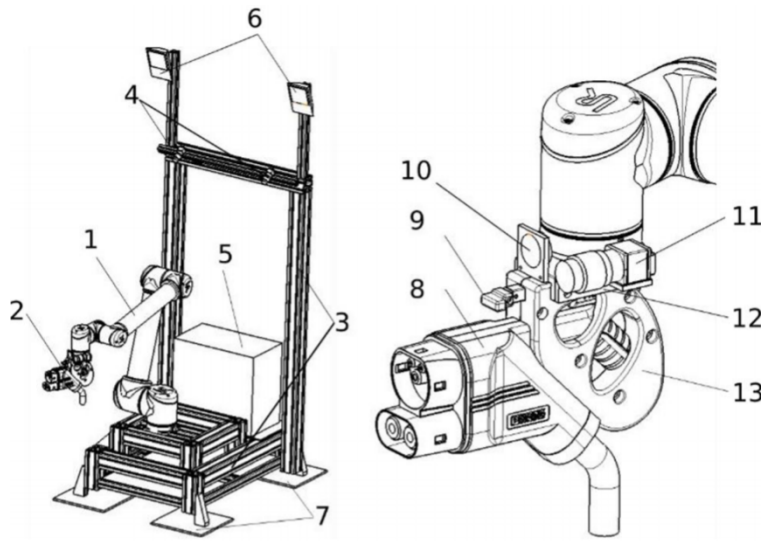


Figure 3: Left: Overall picture of the prototype. Right: Detailed picture of the handle [3].

The cameras located on the station are responsible for checking the occupancy of the charging area as well as finding the approximate position of the charging port using shape-based 3D matching, which will be explained in the next paragraph. A vehicle entering the charging area results in a contour classification of the vehicle. The charging process can begin if the vehicle is successfully identified, recognized and furthermore has opened its charging lid. The station-placed cameras start searching for the charging port and compare the found position with stored data of the vehicle, the rotational coordinates of the port are not of importance in this step. The robot starts moving towards the port using a predefined path if the found position is within the working range of the robot. The camera placed on the robotic arm launches an accurate position detection using shape-based 3D matching as well. Where a robust and accurate detection of the six dimensions (x-coordinate, y-coordinate, z-coordinate, rotation around x, rotation around y and rotation

around z) results in a highly accurate robot motion sequence docking the connector with the port. Undocking is launched when charging is complete including a predefined movement closing the charging lid. The author emphasizes that this predefined movement can be problematic when dealing with cars using dissimilar lid kinematics and thus needs to be taken into consideration [3].

Shape-based 3D matching is a method using the contours of a known object to determine its position in a camera image. A 3D model is generated from a CAD model which is used to create 2D projections from different views using virtual cameras. These projections include the position in relation to the camera coordinate system. The resulting 2D representations for each view are used when searching for a specified object in an image. The best matching representation is used to obtain the object position. The project in question used a 3D CAD model which was converted into a 2D representation in DXL-format for further processing using a software called "HALCON". This software was moreover used when calibrating the mono cameras [3].

Various tests were carried out to evaluate the prototype functionality, the robustness of the system and the maximal vehicle misalignment possible for successful charging. The tests were performed by allowing human drivers to park a car in an indoors charging area and letting the prototype run its course. 42 tests were carried out resulting in 42 occurrences of successful docking, plug-in and undocking and 40 times of successful closing of the charging lid. The two unsuccessful tests were a result of an overly angled vehicle resulting in insufficient closing of the lid [3].

### **2.1.2 Ericsson A.R.M.S**

An automated repair and maintenance robotic solution for maintaining Ericsson base stations, primarily in areas with high variation in operation or extreme climates, was in 2018 developed by A.Hosseini and A.Karlsson et. al. [1]. The solution in question was designed to, when triggered, automatically replace malfunctioning Small Form-factor Pluggable (SFP) modules, which is a common fault in base stations, with new ones to reduce the time and effort regarding these maintenance procedures. The procedure includes extracting the cable from the malfunctioning SFP, inserting the cable into a new functioning SFP and maneuvering as well as inserting the SFP and cable into a new port. The removal of the malfunctioning SFP was neglected in this project [1]. A picture of a digital unit with connected SFPs and optic cables is shown in Figure 4.

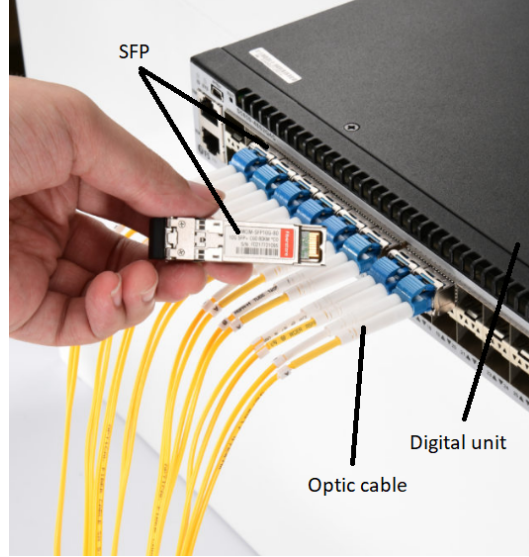


Figure 4: Digital unit with inserted SFP as well as fiber cables [44].

The system considered is utilizing an ABB robot arm (model IRB120) which is equipped with several items including a camera, a six axis force/torque sensor and a pressure sensor. [1], see Figure 5

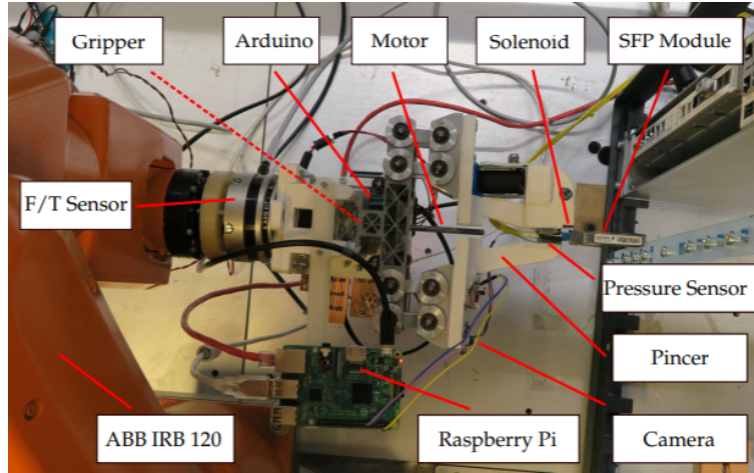


Figure 5: The total automated repair and maintenance system [1].

The pressure sensor is used to control the force applied to the cable when extracting it. The camera and force/torque sensor are used for aligning and injecting the SFP (and cable) into a port on the digital unit. The problem of aligning the SFP to the port can be split up into four different steps [1]:

1. The system moves to a predefined position where the camera captures an image of the port which is used to estimate the next position.
2. The system moves to the estimated position which is adjacent to the port.
3. The system uses the force/torque sensor to find the port utilizing an algorithm correcting the position in direction of the resulting force or torque when in contact with the digital unit, see Figure 6.

4. When the system has moved past a given threshold, symbolized by the red non-dashed line in Figure 6, the problem of injecting the SFP starts.

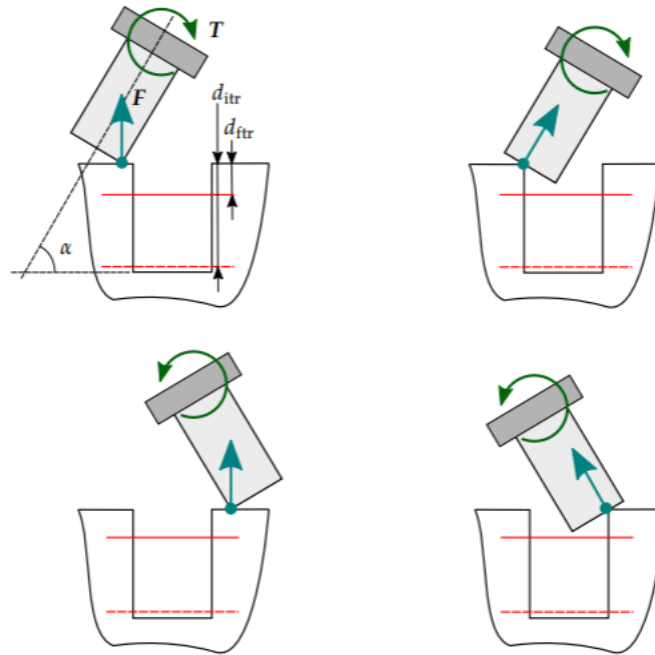


Figure 6: The resulting forces and torques when in contact with the digital unit [1].

The problem of injecting the SFP utilizes a similar algorithm as the one given above, controlling the position corresponding to the resulting forces and torques [1].

Aligning through computer vision, using OpenCV, was furthermore investigated and virtually tested in the project, using a method called "Hough Line Transform". Hough Line Transform operates by detecting and confirming lines in an image. A threshold is set which specifies how many times a line has to be confirmed to be interpreted as an actual line. This threshold can be altered to reduce noise. One can more accurately obtain relative positions and angles of an object with known dimensions using Hough Line Transform. Some kind of image processing is however essential before applying the transform. The project in question utilized a method called "Canny Edge detection" [1] which is a built-in function in OpenCV [4]. The image was first converted to gray-scale, secondly blurred and finally processed using Canny Edge detection resulting in an image with the distinct lines [1], see Figure 7.



Figure 7: Left: SFP connector image before processing. Right: SFP connection image after processing [1].



The processed image was thereafter analysed using the Hough Line Transform [1]. The transform returns parameters  $\rho$  and  $\theta$  for each line detected.  $\rho$  represents the perpendicular distance from the origin of the image to the line and  $\theta$  represents the angle formed by  $\rho$  and the horizontal axis [2], see Figure 8.

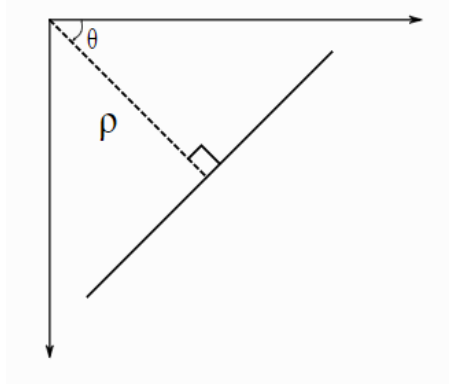


Figure 8: Picture describing the parameters  $\rho$  and  $\theta$  [4].

A line-filtering process was initiated filtering out non-vertical or non-horizontal lines with a margin of  $\pm 0.1$  radians. The remaining lines were henceforth iterated through to find the outermost lines using the value of  $\rho$ . The resulting four lines illustrate the detected object (SFP in this case) [1], see Figure 9.

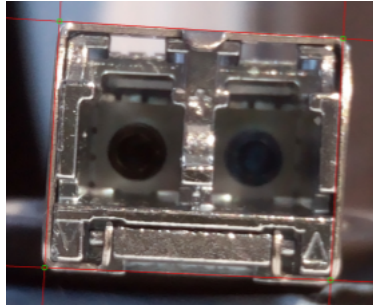


Figure 9: The resulting four lines illustrating the detected SFP [1].

The method was verified by taking a picture of an SFP in lighting conditions similar to the lighting conditions acting on the actual prototype and running the algorithm presented above. This previously stated verification process was carried out using multiple light angles and with slight differences in camera positions and angles. The computer vision method was however not included in the prototype [1].

### 2.1.3 Ultrasonic Phase Accordance method

A measurement technique for positioning nodes using ultrasonic (US) transmissions called the *Phase Accordance Method* was developed by Hashizume et. al. at the National Institute of Informatics and the University of Tokyo in 2005 [21]. This technique can accurately identify the relative distance and orientation between nodes by using a one-time US packet.

This technique uses two or more US carrier waves, with different frequencies, which together constitute a waveform called a *sync pattern*, seen in Figure 10a. An US burst signal of this type



is used in the header part of the communication packet between the transmitting node (TX) and the receiving node (RX). By measuring the phase difference in the two frequencies, it can be seen that the phase difference  $\Phi_2 - \Phi_1 = 0$  occurs only once in the US burst signal - see Figure 10b. This point is used as the *epoch* - a base point for time measurement. The epoch can then be used as an input for the commonly used time-based positioning method *Time of Arrival* (TOA). A simple TOA schematic is shown in Figure 11.

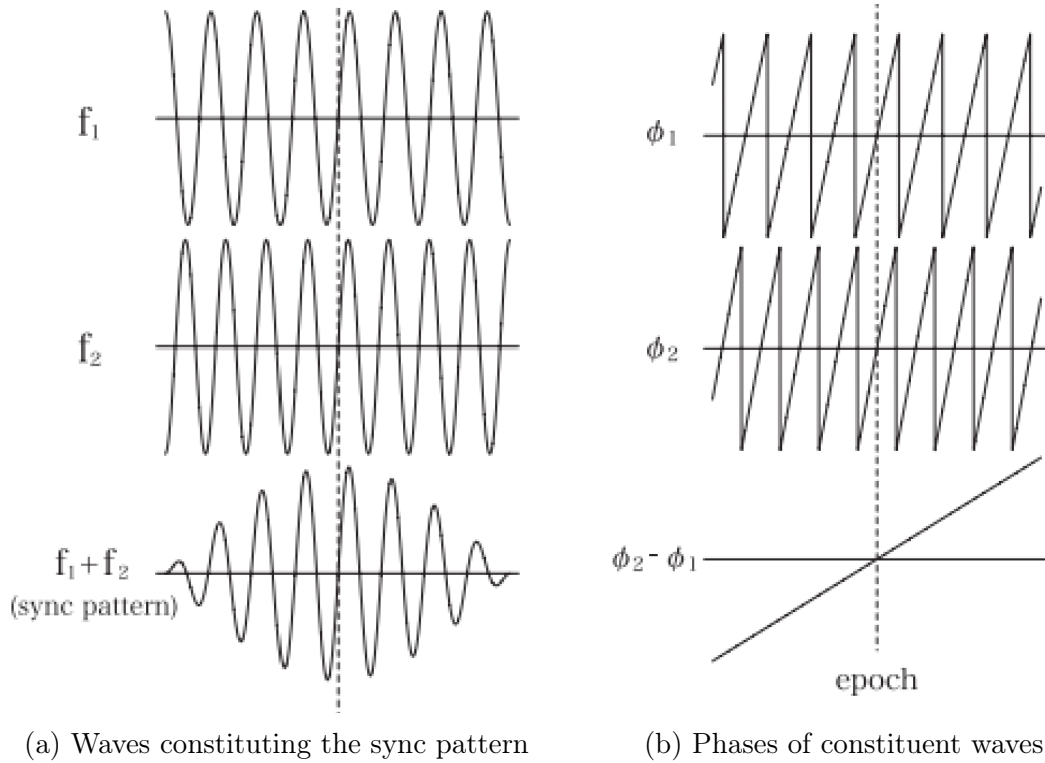


Figure 10: Carrier waves constituting the sync pattern. [21]

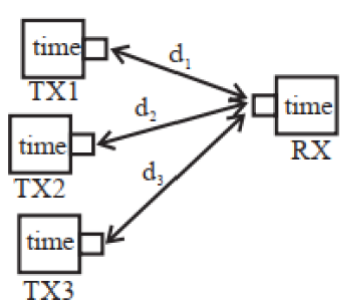


Figure 11: TOA schematic [21]

In the TOA method the distance between each TX and RX node is calculated by dividing the time between transmission and receipt (using the epoch) of a US signal and dividing by the speed of sound. Time synchronization between the nodes is required and can be achieved using for instance a US communication channel or a separate Local Area Network (LAN) channel. By using the epoch to determine exactly when the packet is received at the RX, the positioning error can be reduced to a few millimeters [21].

### 2.1.4 High Precision Ultrasonic Positioning Using a Robust Optimization Approach

Another approach for US 3D positioning was developed by Khyam et. al. [30] for medical applications. US 3D positioning commonly uses the trilateration method, wherein three or more US RX's receive a narrowband US signal from a single TX, see Figure 12. The time of flight of this signal is used to determine the distance (a radius) from each RX to the TX, creating a sphere around each RX. When three RX's are used for 3D positioning, these spheres intersect in two points - usually one physically impossible (behind the receiver, below the ground etc, which can be discarded), and one physically possible, which is determined to be the correct position. The problem with this method is that noise and multi-path echoes create uncertainties when the US signal has completed one cycle.

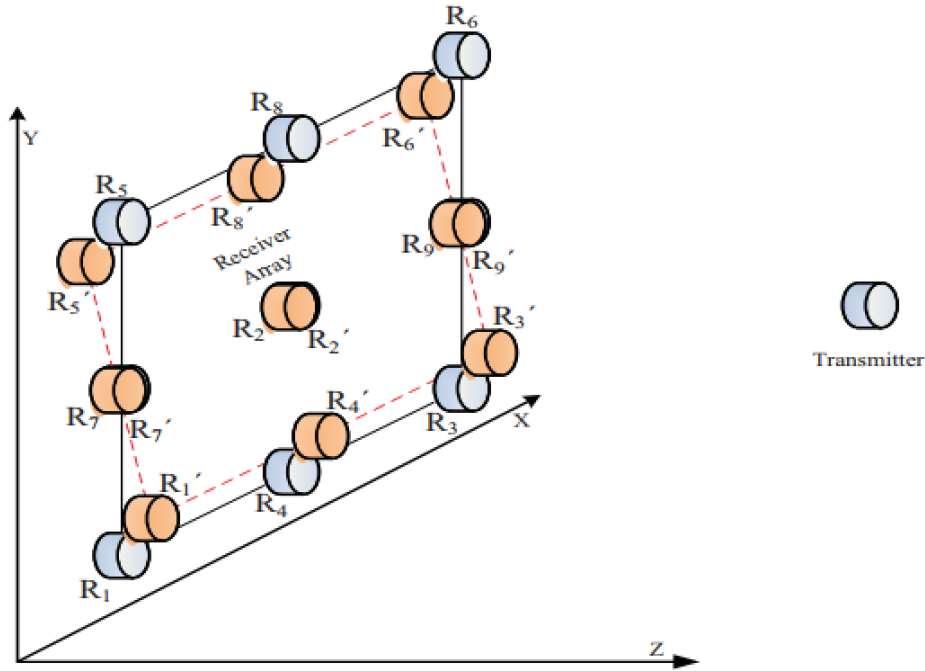


Figure 12: US receiver grid and transmitter positions illustrated [30]

Furthermore, in order to achieve sub-millimeter precision with US methods, the RX nodes are commonly placed on the same fixed plane for logistical reasons. This leads to the sphere surfaces being almost parallel, which in turn means that uncertainties in the sphere radiuses will generate larger errors in the direction tangential to the sphere surfaces (i.e. parallel to the mounting plane) than the direction normal to the sphere surfaces when using trilateration [30].

In order to solve these two problems, an alternate method was implemented - firstly, the multi-path problem was avoided by, instead of using a narrowband signal, using Orthogonal Frequency Division Multiplexing (OFDM) - an approach where the available frequency spectrum is split into a number of subcarriers. The second problem is managed by implementing a steepest-descent optimization algorithm with a 3D rigid body transformation to estimate the 3D position of the TX. Both improvements are described in detail in the paper [30].

By using this approach and an array of nine US receivers and one transmitter, the standard deviation of the positioning error was less than 0.15 mm for all three dimensions (x,y,z) [30].

## 2.2 Computer vision

A part of artificial intelligence (AI) is computer vision, where it enables trained computers to interpret and recognize the visual world [6].

The trained computer broadly involves the research and development of algorithms with deep learning methods to enable machines to identify and classify objects [36].

Image processing is performed by manipulation and analysis, its major subareas include the following processing components:

- **Digitization and compression** of the image by efficient coding or approximations. This is done to reduce storage or channel capacity.
- **Restoration and reconstruction** to receive an optimal processing view of the image to improve degraded images.
- **Recognition and matching process** comparing and registering images to one another by segmenting into parts and measuring properties. This is done comparing the resulting descriptions to models that define classes of pictures [38].

There are many ways to match or register two pictures with one another, or to match to some given pattern. This part is known as *pattern matching*.

Pattern matching in computer science is the checking and locating of specific sequences of data of some pattern among raw data or a sequence of tokens [46]. The sequences are based on gray scale i.e how dark or light each individual pixel is and also the edge gradient information. Comparing this with a reference pattern, determines the differences to identify an object [25].

### 2.2.1 HAAR cascade classifier

An HAAR cascade classifier is a machine learning method to detect objects. It was firstly presented in [45]. With a larger dataset of images categorized as "Positive" and "Negative". The category with positive images contains various representations of the object that the classifier should be trained to recognize. The negative images contain a variety of other objects. It is used during training so that the model can distinguish between the real object and false objects. Studies such as [27] have been done where a HAAR cascade classifier is used for object detection of guns. The study shows results of more than 80% accuracy.

### 2.2.2 CNN

In recent decades, there has been an extreme increase in deep learning methods such as CNN. The idea of neural networks is to imitate the way the biological neurons in the human brain work. By creating an artificial neural network consisting of layers. The layers consist firstly of an input layer, and ends with an output layer, In between are the hidden layers. With the help of artificial intelligence, the hidden layers are trained through a number of iterations, data is sent in as input and after processing the data an classification of the data is output. The

result of classification is then compared to an actual classification of that set. The hidden layers calibrate after each iteration to perform a better classification [39].

Recent studies made by [40] were a CNN object detection method was used to recognize the charging port of an electric vehicle, showed a test accuracy was at 99%.

### 2.2.3 faster RCNN

As an addition to the CNN method is the more advanced faster-RCNN model. It has the same approach as the CNN, but in addition it uses information about class, size and location of the object in the form of a bounding box within the image. For most pictures, there will be different aspects of the object. Because the size and position vary between the images. During the training process, the system tries to detect the object and also improves by limit the area of the bounding box around the object. This technique is aimed to improve the speed and accuracy in training and testing [39].

In a medical study [31], a faster-RCNN model was used for detection of cervical spinal cord injury. With a result of 88% accuracy.

### 2.2.4 Geometry matching

Another similar matching technique is *Geometry matching* which is based on the pattern matching model but also includes matching for a template that is based on geometric features [26]. Due to these included features the geometry matching has a larger workload than pattern matching but gives some extra advantages [12]. Figure 13 shows some of the situations where geometry matching is recommended.




Situation	Example
The object to detect is a different size than the template	
The object to detect is blocked by something else	
There is non-uniform lighting or a change in contrast that softens edges	

Figure 13: Advantageous situations for geometric matching [12].

Previous studies have been carried out to identifying car charging ports with computer vision. They used template matching to identify car charging ports with over 95% accuracy with an offset of 45°, a resulting view can be seen in Figure 14. However, they achieved lower docking results with the help of a robotic arm. Only 80% of successful results when there was a 10° angle offset. According to the performers, this is due to misalignment of the coupling of the charging handle itself [33].

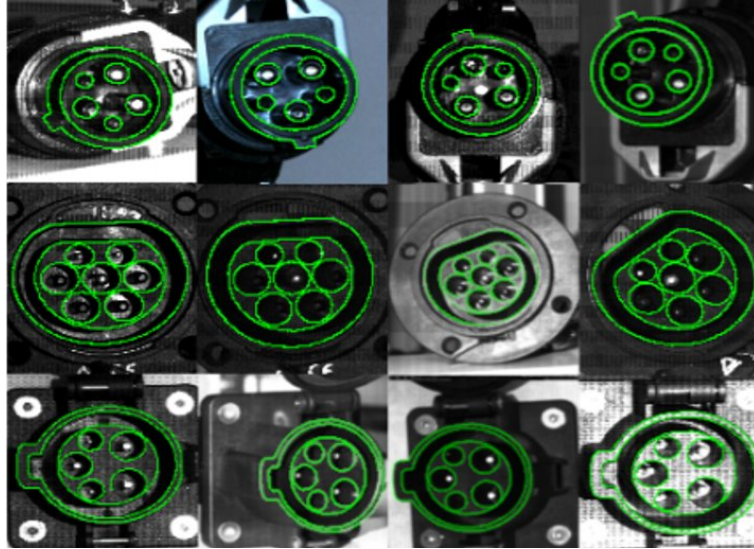


Figure 14: Results of the template matching [33].

## 2.3 Electronics

To be able to control the system, electronics are needed. The following sections describe crucial components such as different kinds of sensors needed to achieve expected behaviours.

### 2.3.1 Encoder

There are several types of encoders that work by different principles, although they all keep track of linear or rotary motion. With a rotary encoder attached to a shaft the position and velocity can be read to form a mean of feedback.

An absolute rotational encoder works by creating two off phase pulses from a rotating disc. These pulses can be created as mentioned in different ways, for example from optic, magnetic or laser. The periodicity and duration of the pulses can be used to receive rotational position, velocity and direction. If the power is lost an absolute rotational encoder would lose track of the exact position, an incremental rotary encoder uses an extra pattern on the rotary disc to overcome this problem. [15]

### 2.3.2 Potentiometer

A potentiometer is a variable resistor with three connections. One for voltage, one for ground or lower voltage and the third as output. The resistance is changed by the rotation of a shaft which will lead to a varying output voltage. This results in that potentiometers can work as a form of feedback of rotary position similarly as an encoder. [43]

### 2.3.3 IR sensor

Infrared sensors operate in the wavelengths which are next to the red visible light section on the spectrum. These wavelengths relate to the temperature emitted from objects. The higher the temperature, the higher the spectral radiant energy from the object and the higher the wavelength.

Infrared sensors are divided into three main types:

- Short Wavelength Infrared Radiation (SWIR). Wavelengths of 0.9 - 1.7  $\mu\text{m}$ .
- Medium Wavelength Infrared Radiation (MWIR). Wavelengths of 3 - 5  $\mu\text{m}$ .
- Long Wavelength Infrared Radiation (LWIR). Wavelengths of 8 - 14  $\mu\text{m}$ .

### SWIR

SWIR sensors, or cameras, rely on reflected infrared radiation. They are well suited for outdoor and low-light environments. The cameras use similar reflecting principles as visible light which is why artificial infrared illumination is required. To these cameras, water vapour, fog and silicone appear transparent therefore, they can provide clearer images than traditional visible light cameras.

### LWIR & MWIR

The LWIR and MWIR sensors are both used for thermal imaging. They both operate in the thermal range and do not require radiation to be reflected. They sense the radiation that is emitted. MWIR is used when a high quality image is required whereas LWIR is used for general temperature and mobility measurements. [17]

#### 2.3.4 IR Depth Cameras

IR depth cameras consist of two cameras with filters so they can only see IR wavelengths. They are usually coupled with an IR laser projector which provides the IR reference point for the cameras. The distance from each point to each camera is measured and used to create a 3D point cloud of the surroundings. As they do not rely on visible light they are unaffected by changes in illumination however they cannot identify colours. The further the objects are from the camera then the more spread out the point cloud and thus a reduction in accuracy. An example is the Intel D435 camera which is shown in Figure 15.



Figure 15: Intel D435 Camera  
[7]

An example of the resulting Intel camera 3D point cloud is shown in Figure 17. The points are given a colour which corresponds to the distance from the camera. The warmer the colour the further away the point is from the camera.



Figure 16: Stereo Images from the Intel D435 Camera [13]

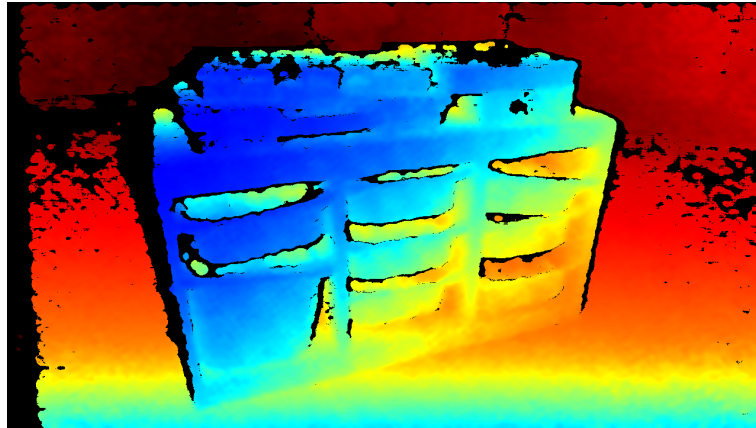


Figure 17: Point Cloud from the Intel D435 Camera [13]

### 2.3.5 Ultrasonic transducers

An ultrasonic transducer can be used to send and receive information via ultrasonic waves. The transducer can be used e.g. to detect distances using the TOA method described in section 2.1.3. An ultrasonic transducer is shown in Figure 18.



Figure 18: An ultrasonic transducer [14]

### 2.3.6 Microcontrollers

A microcontroller is a small integrated circuit board that includes a processor, memory and input/output peripherals, which makes it classify as a computer. Microcontrollers are often



designed for embedded systems where they usually are dedicated to control one, or a few, specific functions. Being compatible with programmable input/output peripherals makes the microcontroller very versatile and used in a wide array of systems, ranging from basic home appliances to advanced medical equipment.

## **2.4 Actuators**

Actuators are an essential part of any autonomous moving object and are therefore a key point of interest. This chapter examines significant actuator types and their strengths and weaknesses.

### **2.4.1 DC-motor**

A DC-motor is a device that converts direct current electrical energy into mechanical energy. The energy conversion is possible thanks to magnets, either permanent magnets or electromagnetic windings, that produce a magnetic field. The two main components of a DC-motor are the stator and the rotor. The content of these components depends on the type of DC-motor, there are two main types:

#### **Brushed**

In a brushed DC-motor the coil is placed on the rotor and the magnets, which provide a constant magnetic field, from the stator. When the rotor is connected to a power source it provides current to the coil via a pair of stationary brushes. Since the coil is rotating against the stationary brushes the direction of the current will alternate, allowing continuous rotation.

The main advantages of a brushed DC-motor are low initial cost and simple control. The main disadvantage is the wear of the brushes that slide against the rotor, this means the brushed DC-motor has a relatively high maintenance.

#### **Brushless**

In a brushless configuration the magnet is located in the rotor and the coils are located in the stator. The power source is connected to the stator in which the coils provide a rotating magnetic field, which in turn rotate the rotor allowing continuous rotation.

The advantages of this type of DC-motor include high efficiency and very low maintenance. The disadvantages are high cost and more complicated controllers compared to other types.

### **2.4.2 Stepper motor**

A stepper motor works in a similar way as a conventional DC-motor except for a few differences. The stepper motor works by having an active control of the electromagnetic fields instead of a passive control. Normally a stepper motor uses permanent magnets as the rotor and electromagnetic windings as the stator. An energized winding in one direction will lead to rotation to that position, the next winding will subsequently be energized keeping the rotor turning. The switching of winding energizing is accomplished by a stepper motor driver. The advantage with a stepper motor compared to a DC motor is the indirect form of feedback, since it takes a certain amount of steps each rotation the rotational speed and position is fairly well known. Although the extra control comes at a cost of size and price [9].



### 2.4.3 Servo motor

Servo motors are also a form of rotational actuator. It is a combination of a DC-motor, a potentiometer, a gear reduction unit and a control circuit. Together they form a motor that has lower speeds than a regular DC-motor but higher torques, as well as control of angular position. Servo motors are not intended to rotate constantly, instead they are made to serve a specific angular span. Most servo motors are limited to work in a 90 degree or 180 degree span while others have the possibility to rotate further. The feedback within a servo motor is often done by a potentiometer but configurations with encoders exist. The servo motor uses the feedback of actual position and reference to control the motor current and in that way achieve the specified angle [16].

### 2.4.4 Linear actuator

A linear actuator creates motion in a straight line. There are many different types, for example: mechanical, electromechanical, hydraulic and pneumatic. Hydraulic and pneumatic actuators operate in a very similar way, they both generate pressure in order to push a cylinder. A mechanical (and electromechanical) actuator usually achieve linear motion by conversion from rotary motion, often by rotating a screw. An advantage regarding the mechanical actuator with a screw solution is that it has static loading capacity i.e. it is capable of locking the mechanism in place and support pushing or pulling loads.

## 2.5 Software

In a project like this, which contains moving parts, actuators, sensors and image recognition, appropriate software is essential. The software needs to be able to handle communication with the carrier, as well as be able to drive the motors. The software should also be able to handle sensors and image recognition, with or without external packages.

ROS is a software that has the all above mentioned features and can, using external packages, handle all the requirements mentioned above. For the image recognition an external software packaged will be needed, the package is called OpenCV.

### 2.5.1 ROS

ROS stands for Robot Operating System. It is a framework for coding robot software. It has a lot of different tools and libraries to facilitate creation of a robot with complex features.

For example, one of the libraries that can be useful, is the "moveit" package. Using this package will for instance reduce the amount of code written as the package only requires a few changes to enable movement of the robot arm

To be able to use the ROS software, a master is needed, it is often a computer that is Linux based. The robot also needs a Linux operating system. Which often utilizes a hardware with higher mobility than a computer, for instance a Raspberry Pi etc.

### 2.5.2 OpenCv

OpenCV is a programming library within computer vision and image-analysis. OpenCV was originally created in C++ but now also has an interface in Python, Java and MATLAB [41].

The library has over 2500 optimized algorithms. Which includes both image vision and machine learning algorithms.

Computer vision will be needed when the robot arm needs to locate the charging port on the carrier. OpenCV Will therefore be of great help with all its advanced algorithms.

### **2.5.3 Tensorflow**

Tensorflow is an end to end user open source platform for machine learning. Which includes a large number of libraries and tools for modeling AI recognition systems. Tensorflow is a multilingual platform that handles C, Java, Go and python [47].

### **2.5.4 Intel OpenVino**

Intel has developed a software which is aimed at facilitating the use of neural networks and deploying them, called OpenVino. The software works by converting neural network models into a specific format which is comprised of only two files [34] . These are then used in Intel's OpenVino software which carries out the inference. This format also allows for the use a VPU called a Myriad X [24] which allows the inference computation to occur on the VPU instead of the computer CPU or GPU. The Myriad X processor is available in various forms and can be embedded straight into single board microcomputers or on external devices. Multiple Myriad X processors can be used to increase the computational power and provide more efficient solutions for using neural networks. An example is the Intel Neural Compute Stick which is has a Myriad X VPU embedded into a USB stick format.

## 3 Methodology

The project can be split up into three distinct parts. Beginning with the SOTA report and concept design in the first phase. Where the groundwork for the whole project was set. By reviewing similar technologies and doing basic research. The second part was all about organization and planing. The decision of splitting up into teams to maximize the efficiency was made there. Then finally the execution. Where the prototype was built.

### 3.1 Project management/Organization

To be as effective as possible, the team was divided into three subgroups during the spring semester:

- **Carrier** - Responsible for locating the charging port and alignment between port and connector (4 people).
- **Station** - Responsible for designing a concept for the charging station as well as the plug (3 people).
- **Charging** - Responsible for research on existing connector types and deciding which one is best suited for this project. Also responsible for the connection between plug and charging station (2 people).

During the spring semester, and up until the ordered parts arrived, this was the structure. This method of working is especially effective in the early stage of the project since most of the work is researching existing solutions. Searching for useful information is time consuming since one has to go through a lot of information that is of no interest. Having most members research different topics and only bringing up useful information during meetings saves a lot of time.

In the fall semester, once most parts had arrived, the groups were restructured in order to build a prototype as fast as possible. This meant that the station and charging groups were merged into one group focusing on integration. Developing this kind of machine is mostly hardware in the early stages and then shifting to mainly software towards the end. The division of project members between the groups corresponded to the workload shift from hardware to software.

Since there was a lot to do by many people a time plan had to be made to ensure that deadlines were met and that everyone knew what to do. A great tool for this is the Gantt chart. The Gantt chart was chosen because it illustrates the project schedule and dependency relationships between activities, giving a clear view of what has to be done and when.

Due to the Covid-19 outbreak all communication within the team, during the spring semester, was made online, with at least one meeting per week to catch up and discuss ideas. There were also frequent online meetings with KTH and the stakeholders to ensure that the concept development was going in the right direction. During the fall semester the prototype building had to take place, meaning solely online meetings were not sufficient anymore. While working on the prototype, care was taken to be as few, and as far apart from each other as possible.

## 3.2 Engineering approaches

The whole project was broken down into smaller problem sets and were dealt with piecemeal. The hardware was designed incrementally with rapid prototyping by utilizing CAD software and then building parts using the in-house 3D printers and other equipment. This way problems were spotted early and thus fixed early. Instead of wasting a lot of time welding a piece, only to find out it did not work with another one, resulting in a remake. Prototyping could be done in a quick manner, be test-fitted and experimented with to achieve different configurations before building the final assembly. The software problems were dealt with in a similar way. It was decided to use ROS due to its distributed architecture, each *node* could be tested and run on a separate machine and it was also easy implement model based development by *faking* other nodes for testing purposes.

## 3.3 Development Process

To build the charging station, a bottom up approach was used, since it was designed around the dimensions and limitations of the actuators. Firstly, the actuators where mounted on a wooden construction rig which was a good way to start to get a grasp of how the actuators for upwards and sideways motion should fit together, which helped the development process. After that, a solution for mounting the outward going actuator was made. After it was initially attached to the sideways actuator, it was found that it needed more support than the carrier on the ball screw on the sideways actuator. Thus a steel attachment was made which is seen in figure 39.

A lot of time was spent on designing the handle so it would be able to rotate in all directions. There were many iterations until a final solution was found.

## 4 Implementation

### 4.1 Concepts Ideas for Station and Alignment

To create a fully functioning station for charging, several parts and sub-assemblies are needed. The concepts have been divided into separate sub-assemblies to be able to easily compare between the different solutions. A lot of ideas have been investigated, some of the most promising ones are examined below.

#### 4.1.1 Charging station

The charging station has been divided into two sub-assemblies, the station itself and the charging handle. The charging station sub-assembly contains the main mechanical arm that has to align and insert the charger into the carrier when it is ready to charge. While the charging handle is supposed to hold the connector and compensate for angle deviations during insertion.

#### 4.1.2 Cartesian charging station

One way to realize the charging station is by using a Cartesian robot setup. A Cartesian robot setup is made up of one or several guide rails with one or several linear actuators along each axis as shown in Figure 19 below. The amount and size of the guide rails as well as the linear actuators depends on the desired stability and torque of the construction. There are different kinds of linear actuators as mentioned earlier, although the most common is ball screw and stepper motor.

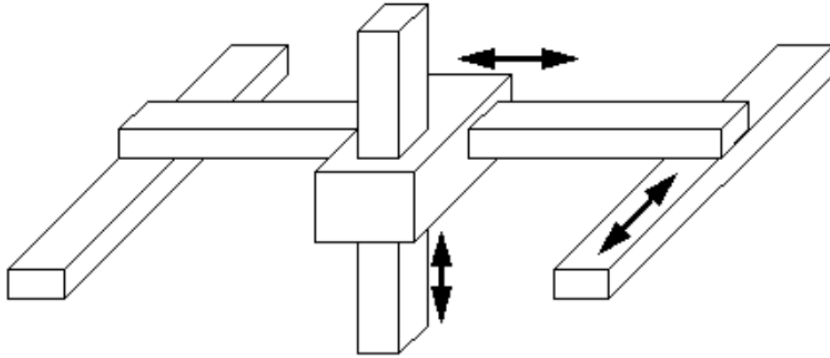


Figure 19: Cartesian robot [10].

The setup has to be fairly robust since the weight of the charging cable and handle can get somewhat heavy with the high power output. Therefore the generated setup uses two ball screws with stepper motors and guides vertically to get some extra torque, shown in Figure 20. For the sideways motion the applied torque should not be a problem but extra stability might be needed, which is why one motor and two guide rails are used. Meanwhile, the outwards motion utilizes one motor and one guide rail of extra length. The longer guide rail would be required since it needs extra distance to reach in between the load carriers wheels to come in contact with the charging port.

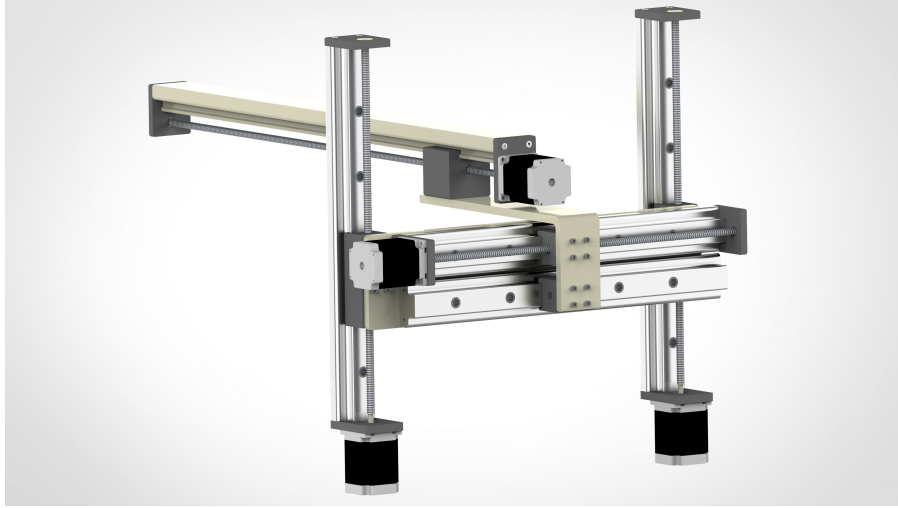


Figure 20: Cartesian charging station motor setup. Made in Solid Edge.

To cover the Cartesian robot setup from the environment and elevate it to the right height, some kind of casing would be required. Shown in Figure 21 is one concept of a charging station with the motor setup described above. The arm would be able to reach the required distance to reach the carrier as well as compensate for the margin or error in position. A handle described in the section below would be mounted at the end of the arm to allow for sideways rotation. In this specific example 11 legs are used on the station to compensate for uneven ground and allow extra height adjustments. Since the station would be made for harsh environments the large front opening is not ideal, therefore some brush strands could be used in the opening to further avoid dirt, rain or snow to enter. Aside from that, electronics would be mounted in casings within the station to further increase robustness.



Figure 21: Cartesian charging station without handle. Made in Solid Edge.

#### 4.1.3 Robot arm and housing

Another solution is a robot that moves in both linear directions and rotational movements. The robot will be able to be mounted on a wall, this will make the robot robust and flexible to move. An example of such a robot is the Carla\_connect, which is shown in Figure 22.



Figure 22: A picture of a robot called Carla\_connect, that has both linear and rotational movements [35].

#### 4.1.4 Motion

To make the robot move in a linear motion a linear ball screw actuator can be used. To enable upward as well as downward movements, the screw will rotate with help of a motor, which will lead the ground ball moving up and down. The robot arm will be fastened to the ground ball and this will make the whole robot arm move up and down. To make the robot arm more robust two rods will be added on each side of the screw, in Figure 23 an example of how this is shown.

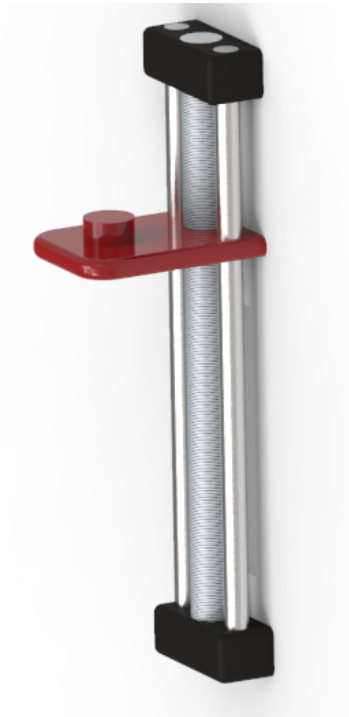


Figure 23: A picture of a linear ball screw actuator. Made in Solid Edge.

The robot arm will at least have three rotational axes. Rotation around those axes will be accomplished using motors. The first rotation will make the first arm rotate. The second rotation is located on edge of the first arm and this will make the second arm rotate. The third rotation will be located on the edge of the second arm and will help rotate the handle. The rotations can be seen in Figure 24 and they are marked with numbers one to three.

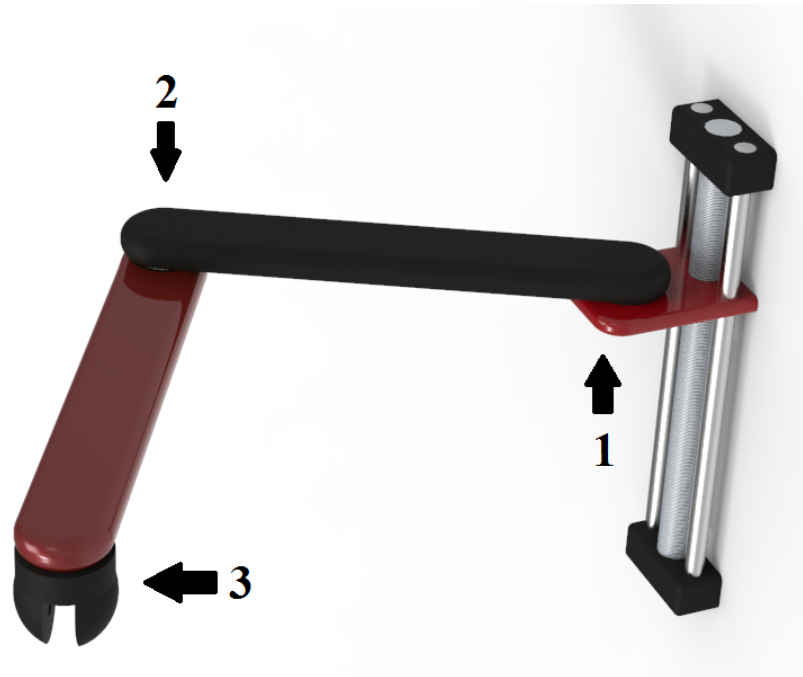


Figure 24: A picture that illustrates where the rotational movements are located. Made in Solid Edge.

#### 4.1.5 Handle

Once the port has been successfully recognized and the arm has aligned itself with the port, the challenge is to make a solid connection to the charging port. There are two ways ensuring that the connection is possible. Since it is not certain that the carrier will be completely parallel to the charging station, there needs to be adjustment on the handle itself in order for a connection to be made.

One way of achieving this is by using servo motors to align the handle according to Figure 25. For this solution, there will be three motors, two of which will rotate the handle horizontally and vertically, and the third motor will slide the handle forward into the charging port.



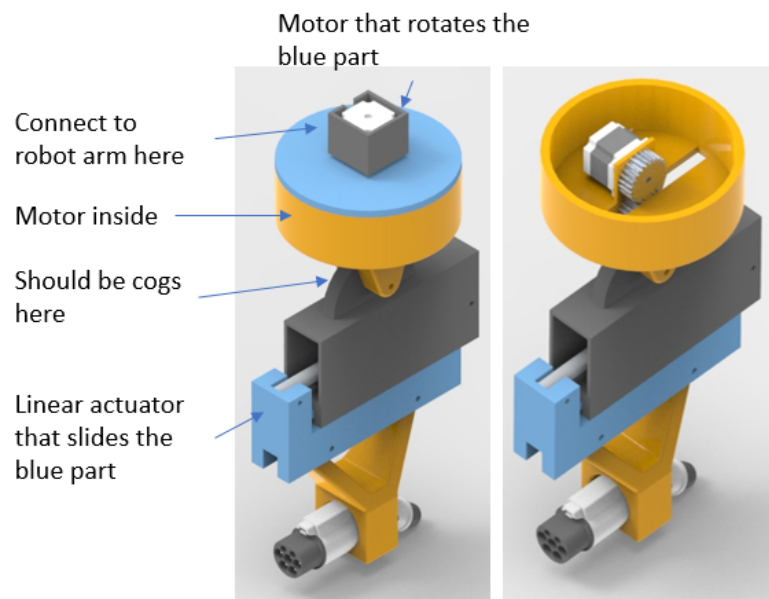


Figure 25: Servo handle with linear actuator. Made in Solid Edge.

The other way of doing this is by a more passive approach, where the handle is guided by a slot inside a funnel in order to make the connection. The handle can be pushed horizontally and vertically and has a spring that brings it back to a straight position when the tension is released, seen in Figure 26.

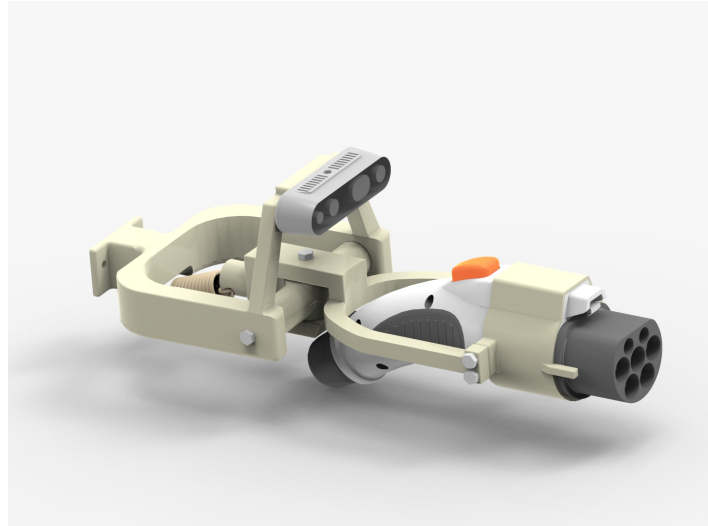


Figure 26: Handle with spring mechanism. Made in Solid Edge.

This version of the handle mechanism is a simpler one compared to the version with servo motors controlling the angle of the handle. However, for this to work there cannot be as big of an angle between the carrier and the charging station compared to in the case with a servo mechanism. The spring is furthermore placed at an angle to compensate for the weight of the handle. The most left part in the figure is where the handle will be connected to the end of the robot arm. Both axes of revolutions are mounted with ball bearings so there is as little friction as possible. The component that holds the charging handle is replaceable with different versions that fit different charging handles. This is important in case the user desires a different charger than the CCS Type 2 that is visible in this figure. An Intel D435 camera is mounted at the front.

#### **4.1.6 Single IR Camera with tape**

A simple solution to identifying the port is to use a single IR camera. IR diodes around the camera are used to illuminate the target area. Reflective IR tape is placed around the charging port in a recognisable pattern such as a square. The IR camera clearly sees the square due to the high contrast from the tape. Computer vision is then used to template match the tape shape and determine the distance and angle of the carrier. The charger can then be moved to the appropriate charging position.

#### **4.1.7 Single IR Camera with diodes**

Another solution for a recognisable pattern is to have IR diodes around the charging port on the carrier. The diodes will provide reference points which will be constant in all lighting conditions. This solution eliminates the need to have IR diodes around the camera.

#### **4.1.8 Intel IR depth camera**

To obtain a more accurate localisation of the charging port a stereo IR depth camera can be used. A camera such as the Intel D435 would be appropriate. It has two infrared cameras and an IR projector which makes it suitable for use in indoor and outdoor environments with illumination variations. The depth camera projects IR lasers which are then sensed by the two cameras and create a 3D point cloud. From the point cloud, the distance, position and orientation of the charging port can be determined.

#### **4.1.9 Ultrasonic and a single camera**

A similar concept to the Intel camera, would be to mount three ultrasonic sensors around a single camera that is attached to the robotic arm. These sensors should be mounted in a triangle pattern, i.e. one on each side and one above the camera. These sensors can by using the TOA method, discussed in Section 2.1.3, calculate the distance to the carrier. In this way, the robot arm can align by making the camera perpendicular to the port on the carrier. After that, no rotating calculations will be needed. Thus only utilizing translational movements to control the connector, through template matching with computer vision, resulting in a correctly connected charging cable.

#### **4.1.10 Mechanical guide**

One way to make sure that the handle goes straight into the charging port is to have guidance on the funnel that is mounted on the carrier, seen in Figure 29. These guides are more spread out the further away from the charging port they are, ensuring that the handle is straight once it has reached the port. The rounded features will ensure that in whichever coordinate axis the carrier is misaligned, the handle has a good chance of connecting.

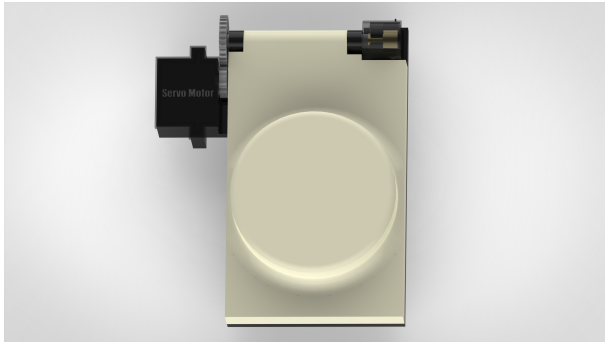


Figure 27: Side view.

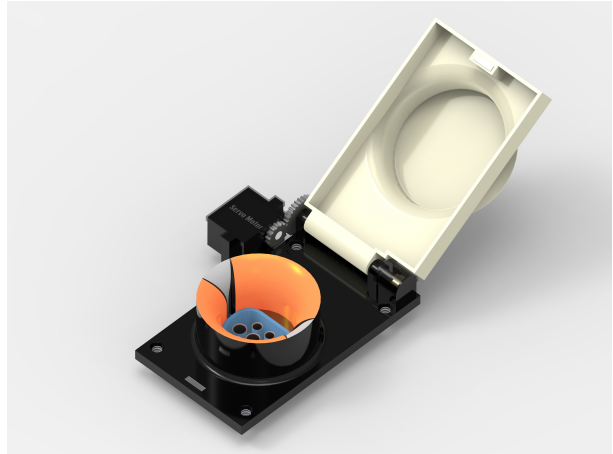


Figure 28: Top view.

Figure 29: Charging port with guide, mounted on carrier. Made in Solid Edge.

On the lid itself there will be a pattern for the visual system to recognize, seen in Figure 30. This will be used for the recognizing the position of the carrier.



Figure 30: Charging port with lid open. Marker for visual system to recognize. Made in Solid Edge.

#### 4.1.11 Charging port

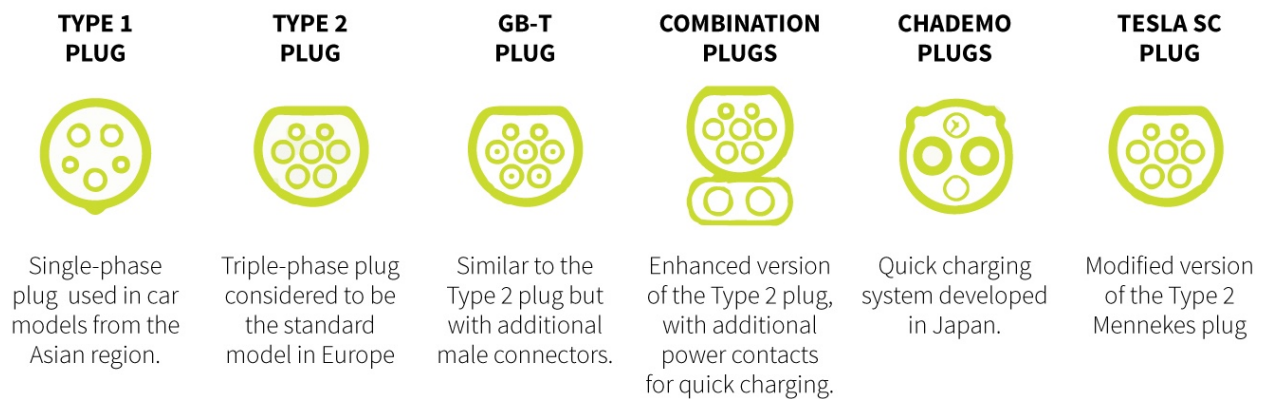
In the design concept the connector used is a Type 2 connector due to it being widely used and has become the de facto standard in Europe which is beneficial as it will bring the price down and make it easily available.

On top of that, it is very versatile, giving us the option of using DC charging or 3 phase AC. It supports up-to 43 kW and 63 A [22].

---

## Electric cars' plug types

Carmakers have come up with different standards for the type of plug used to recharge their electric cars.



Source: The Mobility House.

G. Cabrera, 17/01/2018

 REUTERS

Figure 31: Example of types of connectors available [8].

#### 4.1.12 Concept evaluation

The following chapter evaluates the different solutions with respect to cost, complexity, robustness etc. and uses the evaluation to determine a final concept. The evaluation is divided into four different parts assessing each subsystem.

Five different solutions were investigated when researching answers to the alignment problem, see chapter 4.1.5. These were evaluated with respect to the following aspects on a scale from one to five, where five indicates that an aspect is highly fulfilled and one vice versa:

- Low cost
- Execution simplicity
- Robustness
- Simplicity on the carrier-side

The result is shown in table 1 below.

Table 1: The resulting evaluation table

	IR-camera + Tape	IR-camera + diodes	Intel IR depth camera	Ultrasonic + camera	Mechanical guide
Low cost	5	4	2	5	5
Execution simplicity	3	3	5	3	5
Robustness	4	4	5	3	5
Simplicity on the carrier-side	4	3	5	2	3
<b>Total score:</b>	<b>16</b>	<b>14</b>	<b>17</b>	<b>13</b>	<b>18</b>

One can see that the solution with the highest score is the "Mechanical guide" with "Intel IR depth camera" in the second place. Since the mechanical guide solution cannot solve the problem by it self a decision is made concluding usage of both the "Mechanical guide"-solution and the "Intel IR depth camera"-solution in the final concept.

Four different solutions were investigated for the mechanical part of this project. Just like for the alignment, four different aspects were evaluated on a scale from one to five, where a five indicates that an aspect is highly fulfilled and a one vice versa:

- Low cost
- Execution simplicity
- Robustness
- Simplicity on the station-side

The result is shown in table 2 below.

Table 2: The resulting evaluation table for the mechanical components.

	Robot arm + servo controlled handle	Robot arm + spring controlled handle	Cartesian station + servo controlled handle	Cartesian station + spring controlled handle
Low cost	2	3	4	5
Execution simplicity	5	3	3	2
Robustness	4	3	4	3
Simplicity on the station-side	1	3	3	5
<b>Total score:</b>	<b>12</b>	<b>12</b>	<b>14</b>	<b>15</b>

The solution with the highest score is the Cartesian charging station with a handle that makes use of a spring to correct itself in case of any misalignment between the carrier and the station. The second highest score is the Cartesian charging station with a handle that uses servo motors to correct any misalignment between the carrier and the charging station. The final concept will consist of the former mentioned. Figure 32 shows the inside of the charging station (no charging cable has been added). Figure 33 shows the complete design of the charging station with the handle attached. Figure 34 shows the entire system, with the lid mounted on the carrier and the charging station beside it.

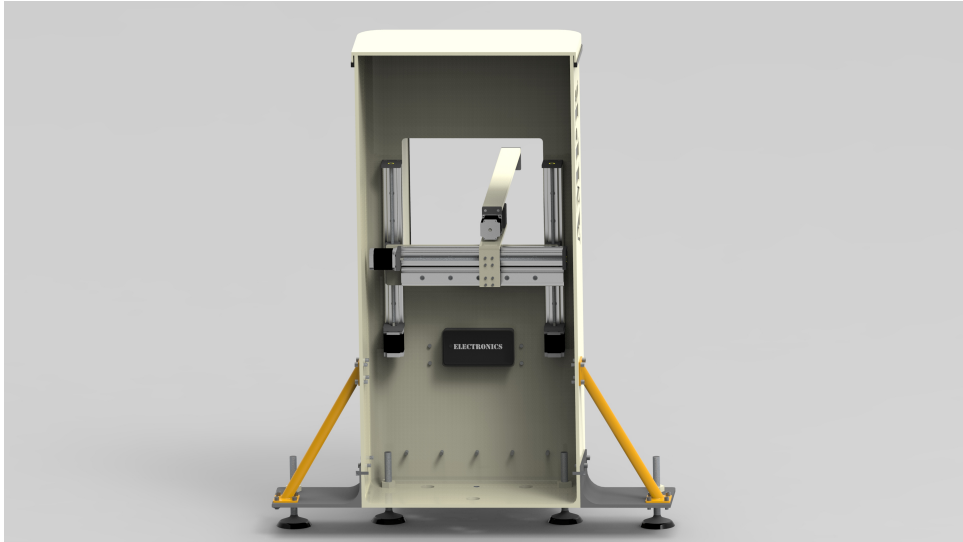


Figure 32: Inside of the charging station. Made in Solid Edge.



Figure 33: Complete design of the system. Made in Solid Edge.



Figure 34: Complete design of the system. Charging station and lid made in Solid Edge. Carrier is from Volvo [23].



## 4.2 Final Design

This chapter describes the design of the proposed concept in detail and its subsystems. The final design of AMPR can be seen in figure 35.

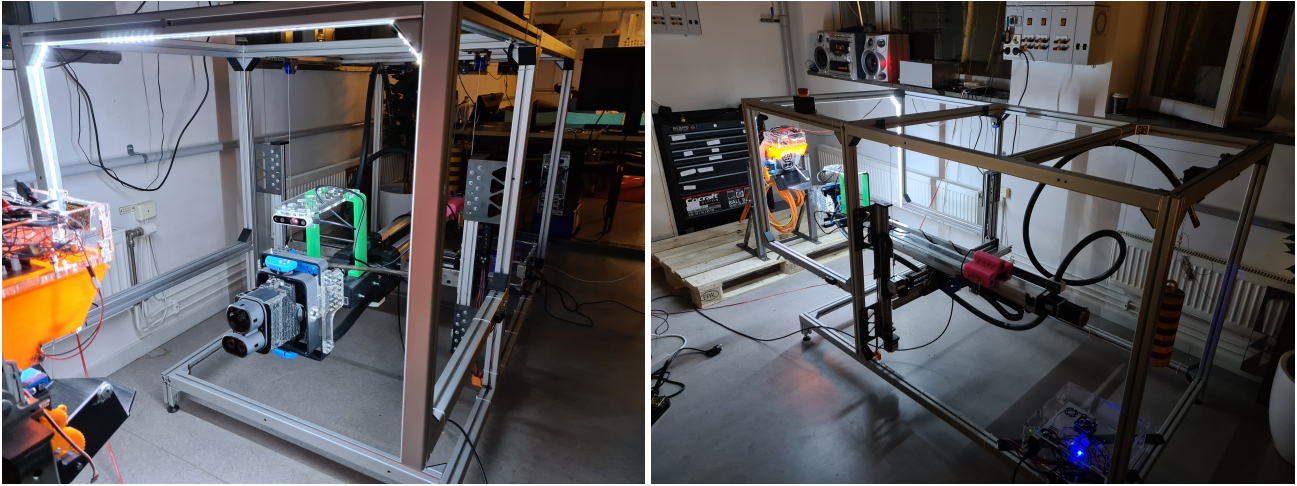


Figure 35: Left: AMPR seen from the front. Right: AMPR seen from a sideways rear view.

### 4.2.1 Mechanical Design

The general layout of the mechanical design is similar to the Cartesian charging station presented in section 4.1. Although several changes have been made throughout the project, resulting in redesigns to be able to fully realise the concept. The availability of machines and tools has been a deciding factor of what is possible to manufacture.

### 4.2.2 Charging cable

The charging contact used is of type CCS combo 2 and was acquired through a sponsorship with Phoenix Contacts. It can support up to 125 A at 1000 V, meaning it has a charging capacity of 125 kW. The charging cable needs to have a large diameter in order to support that amount of power and is thus difficult to manage. Due to the mass of the cable, the amount of torque on the handle will vary depending on the position of the arm. Therefore, an attachment is made halfway out on the aluminium square pipes to clamp the cable in place, see figure 36.

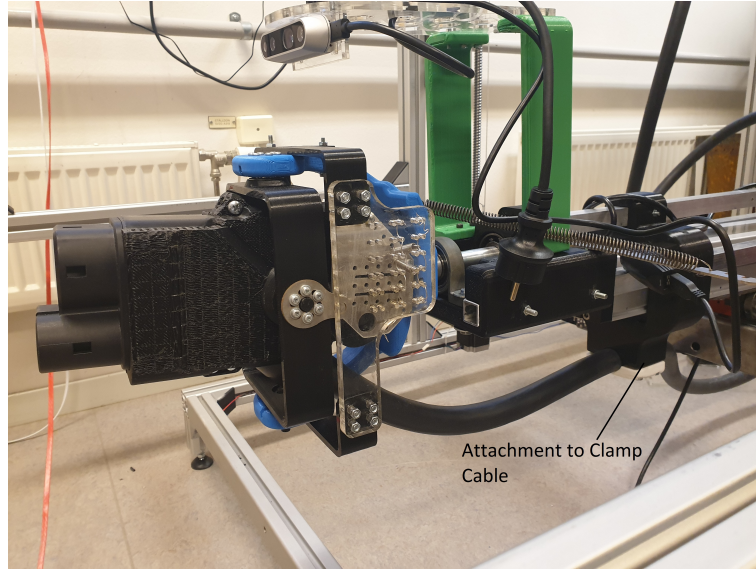


Figure 36: Picture showing the attachment for cable clamping.

By utilizing this method, the cable has a rigid base and will not affect the handle's rotation in any axes based on the position of the arm. An additional attachment was made to fix the cable to the rear of the aluminium square pipes to prevent the cable of dragging along the ground surface. The cable was thereafter attached to the station, see figure 37.

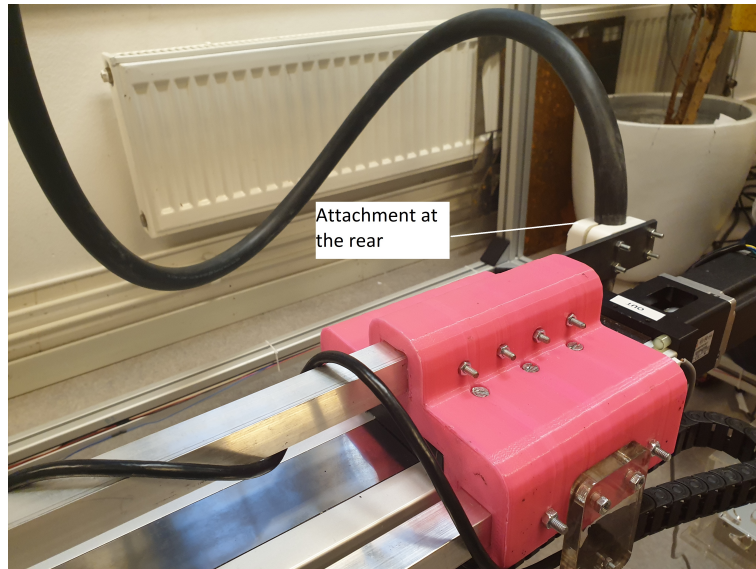


Figure 37: Picture showing the rear attachment for cable clamping.

#### 4.2.3 Motor setup

The motor setup is similar to what is presented in section 4.1, where four linear ball screw actuators with stepper motors are used. The setup consist of two linear ball screw actuators for the vertical motion (Y-axis), one linear ball screw actuator for the sideways movement (X-axis) and one linear ball screw actuator for the outwards motion (Z-axis). The vertical and sideways-moving actuators are the same, however the outwards-moving actuator is larger with a dust-proof casing. Two actuators are used instead of one for the vertical motion to increase both the torque available and also the stability. Since the vertical motion has to lift all other actuators

up and down they are the ones that require the most torque. A counterweight was additionally added to assist the motors in the vertical motion. The counterweight also adds the benefit of the motors not falling freely downwards destroying the station if the motors lose power, which would happen without the counter weight. The counterweight is attached by a steel wire running through a pulley system. The steel wires from each side are bundled together and attached to the counterweight in the back of the station to allow for even pulling force and avoid risks of the stepper motors skipping steps. The full setup of the counterweight can be seen in figure 38.

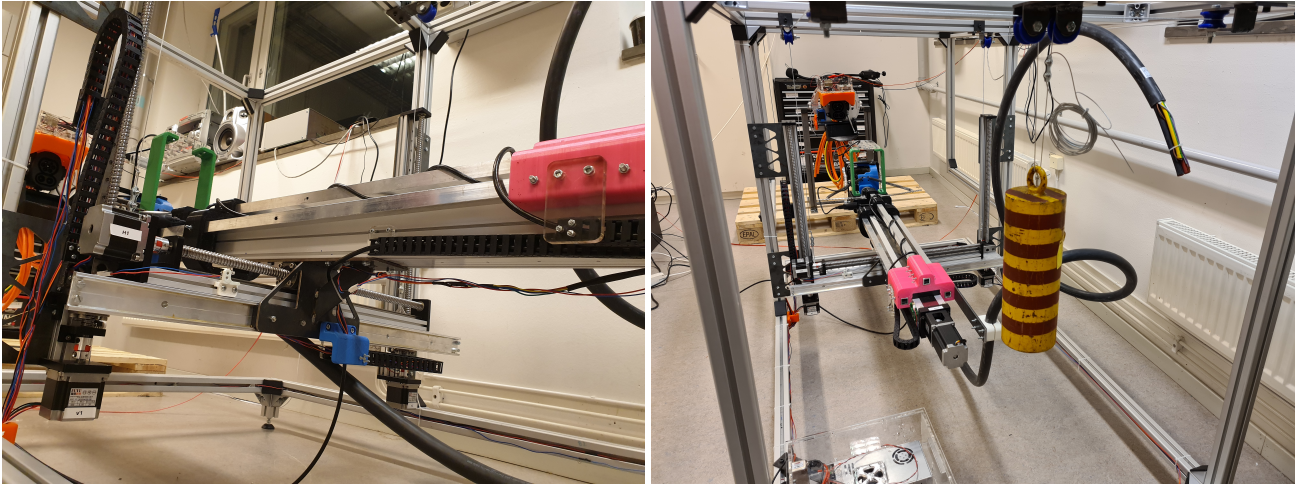


Figure 38: Left: The motor setup. Right: The pulley system and counterweight

For the sideways motion one linear ball screw actuator is used. The outwards going actuator is mounted on top of this one. The center of mass of the outwards-going actuator with the large charging cable attached will differ when going back and forth and thus create a torque around the sideways horizontal axis. Therefore, an additional guide rail is used in order to increase the support. The outwards going actuator sits on top of a steel plate that is connected together with both the sideways actuator and the extra guide rail. In figure 39, the sideways linear actuator together with the steel plate and the extra guide rail is visible.



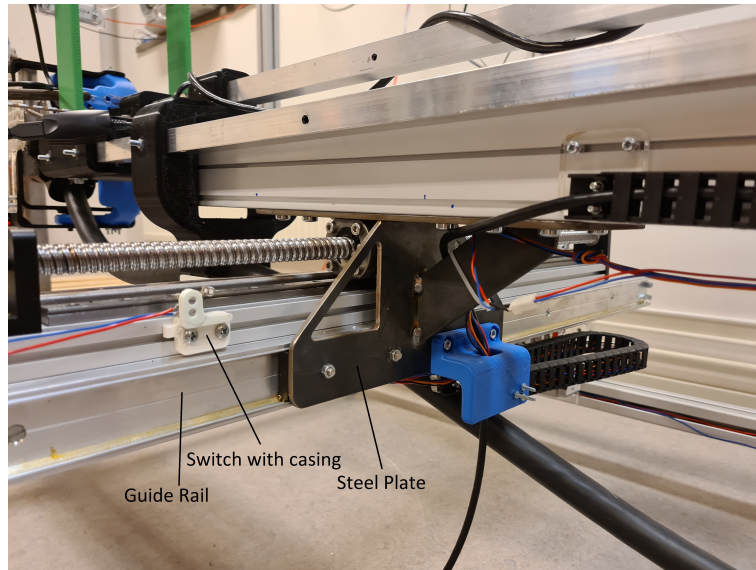


Figure 39: The steel plate and the extra guide rail

The outwards going actuator is a different type of actuator that has the ball screw in a casing to allow for extra protection against dust and water. The casing is necessary because the actuator will be more prone to ingress from rain and dust. This is due to its location near the opening of the station. One large change from section 4.1.2 is that the outwards going actuator is fixed with the carrier on top instead of underneath. In the concept idea the whole actuator would move back and forth which would not be optimal since then the whole weight of the actuator would cause a large bending torque. Therefore, the actuator is fixed in position and only the carrier on the ball screw moves. An attachment fastened on the carrier is made and three aluminum square pipes extend forward. The charging handle solution is attached to the aluminum square pipes. The actuator achieving the outwards movement with the aluminum square pipes can be seen in figure 40.

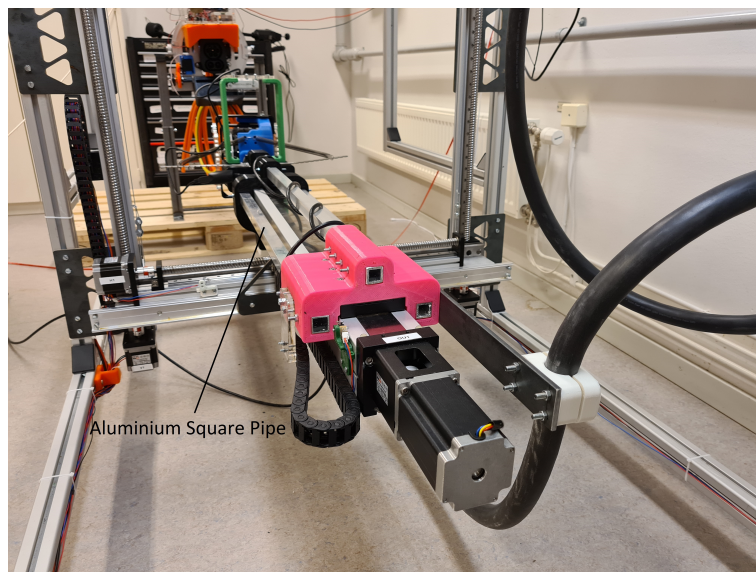


Figure 40: The outwards-going actuator with aluminium pipes attached.

Since the linear ball screw actuators are moved by stepper motors they lack feedback. Stepper motors rotate based on the amount of steps, the stepper motors step based on input from the

MCU to the drivers. Therefore it is possible to calculate their position based on amount of steps it has taken with respect to a reference position. A micro switch is attached to the bottom of each linear actuator to solve this problem. The position of the switch represents the actuators home position and reference point. When booting up the station it moves all actuators to their home positions and thus activating the switches to get the reference point, thereafter the MCU can keep track of its position. One of the four switches with attachment is visible in figure 39.

#### 4.2.4 Handle solution

The spring based concept developed during the spring, see section 4.1.5, was not found sufficient because:

- Its point of rotation is far from the actual handle.
- It is developed for the type 2 plug, however the plug type chosen is CCS combo 2.
- It only utilizes one spring, which might not give satisfactory results.
- It is not compact enough.

The servo motor concept was not found sufficient either because of its bulky appearance and incompatibility with the chosen cable type. A new spring based was therefore developed. The servo based concept was neglected.

Firstly, to be able to fulfill the following requirement, *"The charging station shall be able to handle a 3° deviation around all axes"*, rotation of the handle around x, y and z is required. To achieve these rotations a gimbal system inspired solution was developed. A gimbal system is, in most cases, a system containing three concentric rings, where each ring is capable of pivoting around one axis, see figure 41. The system allows an object attached to the central ring to face any direction at any time [29].

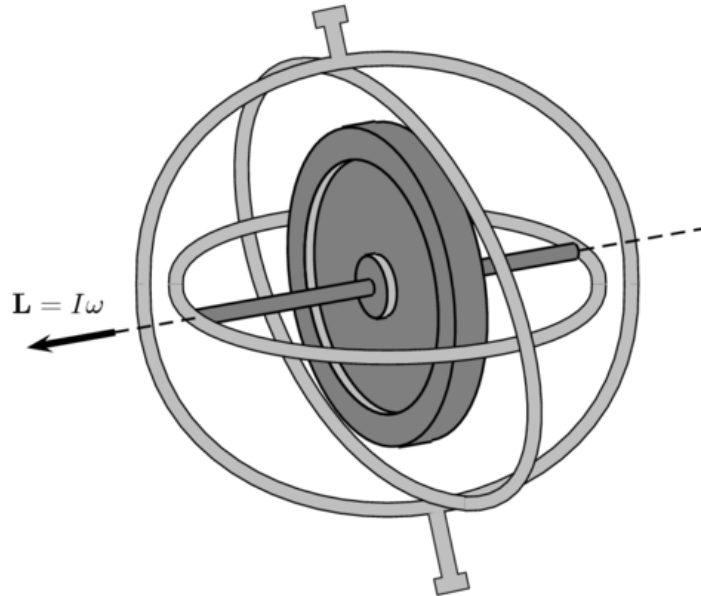


Figure 41: A gimbal system [11].

Attaching the handle to the central ring will therefore result in the rotation capabilities necessary. Secondly, to fulfill the requirement, *"The charging station shall be able to connect to the charging port supplied by Volvo CE"*, an investigation of the port supplied by Volvo is needed, a picture of the simplified Volvo carrier is given in figure 42.

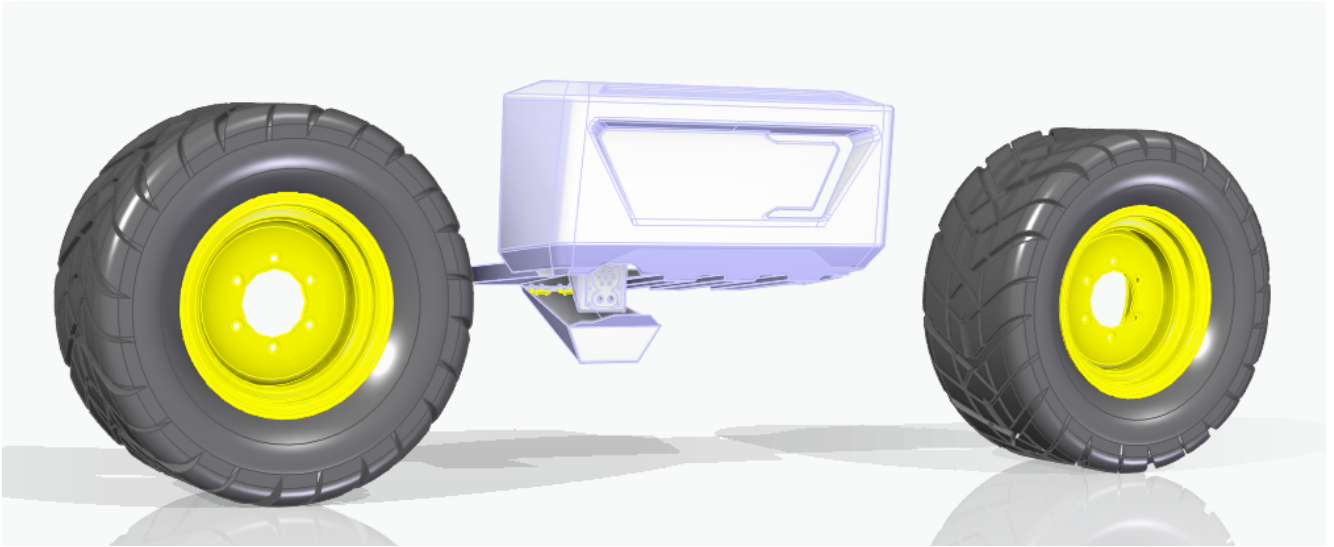


Figure 42: The simplified carrier given by Volvo CE.

The following figure furthermore presents an inserted handle in the Volvo port, see figure 43. Critical dimensions for handle insertion are further presented in the figure.

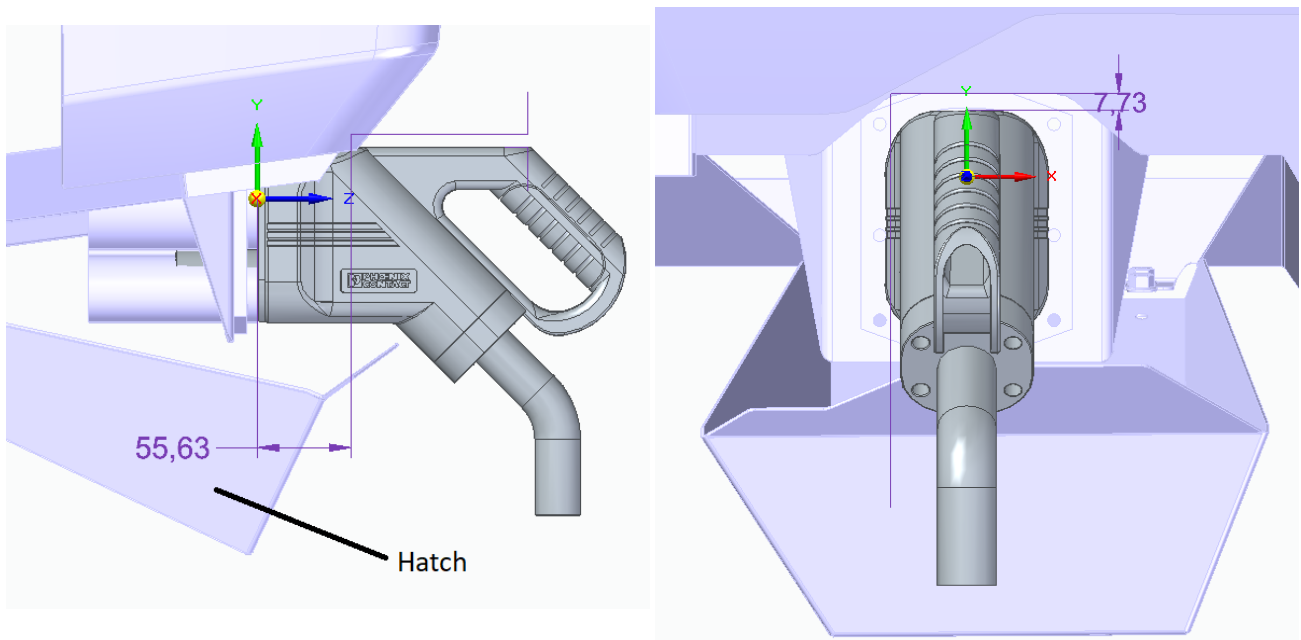


Figure 43: Left: Side view of the port and handle assembly. Right: Behind view of port and handle

Note that the hatch placement in the figure is unnecessarily close to the handle and can be moved further downward. The critical dimensions are thus the distance between the handle head and the carrier with value of 7.73 mm and the distance between the port front and the

previous used point on the carrier with a value of 55.63 mm. These dimensions have to be taken into account when generating a handle solution. A concept was generated utilizing three springs.

The gimbal inspired assembly of the spring based solution can be seen in figure 44.



Figure 44: Spring based solution. Springs not included.

The mechanism mainly consists of a case for the plug and two attachments for rotational movement, which combined, allow for rotation along all axes. All axes are mounted on bearings to minimize friction. To aid the port in adjusting itself to correct for angle deviations there are springs attached in between the casing and the attachments. The springs also help in centering the connector. There is a great surplus of holes in order to easily loosen and tension the springs if needed. The full-scale handle concept showing the placement of the springs is given in figure 45.

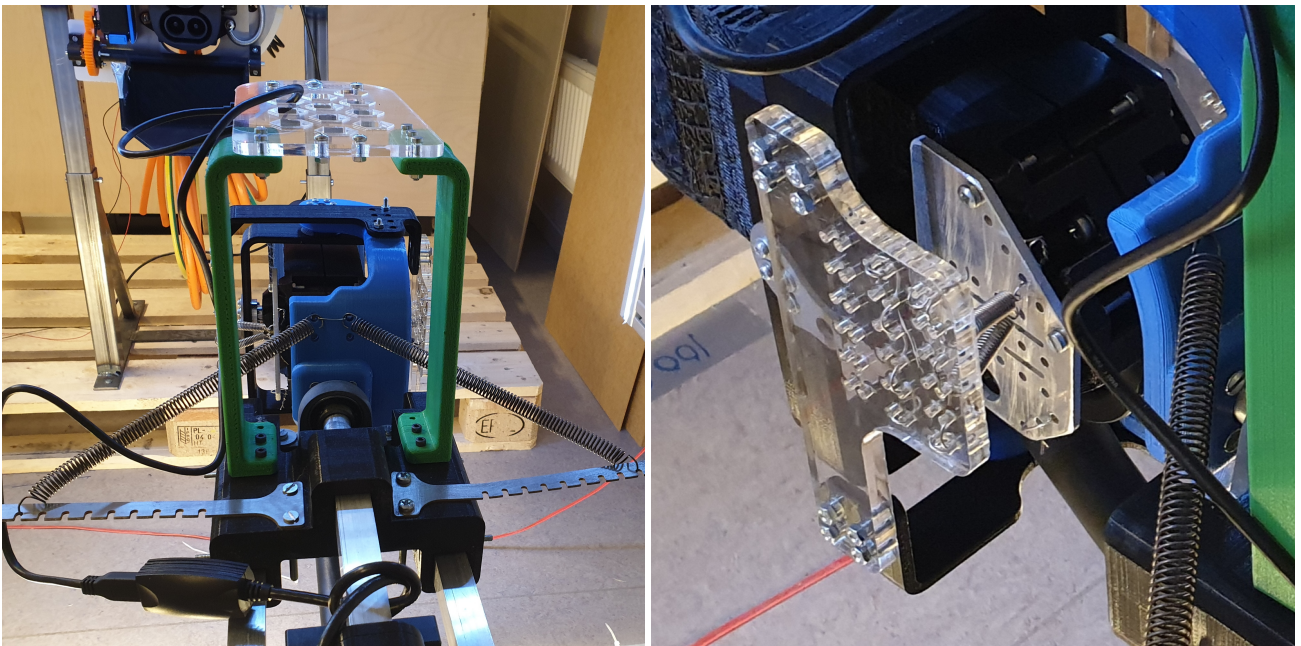


Figure 45: The full-scale handle concept with springs.



#### 4.2.5 Carrier

The lid solution proposed during spring, see section 4.1.6, was, after a discussion with the stakeholders, not found sufficient. Volvo CE had during the summer developed a port assembly on their own. An additional thought during the spring term was to utilize a funnel for easy insertion of the connector, see section 4.1.10. This idea was also aborted due to unclear instructions from the stakeholder. A carrier replica, based on this assembly and the overall appearance of the carrier without a funnel, was therefore developed. The port assembly supplied by Volvo CE is given in figure 43 including the crucial dimensions for handle insertion. It is furthermore advantageous to design the replica allowing it to rotate around all axes to investigate the following requirement, *"The charging station shall be able to handle a 3 ° deviation around all axes"*. The replica shall, moreover, have adjustable height for validity. The resulting carrier prototype is given in figure 46.

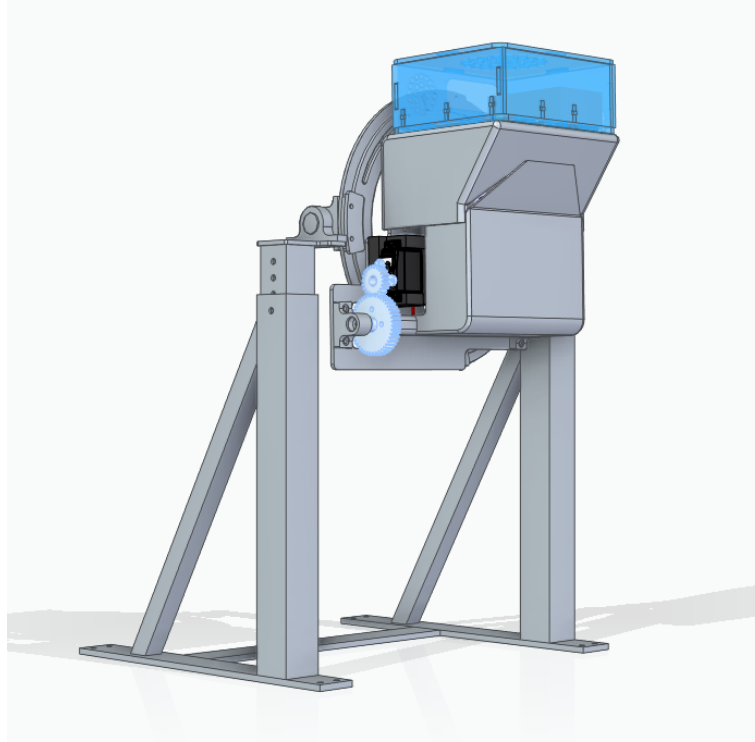


Figure 46: The carrier replica. Made in Solid Edge.

The solution uses the critical dimensions for plug insertion and the overall appearance of the Volvo CE port assembly to create a carrier prototype as close to Volvo's own port assembly. The critical dimensions as well as the appearance are approximately the same in the generated prototype as the supplied port assembly, see figure 47. The carrier replica is furthermore equipped with a servo-motor, a spring and 2 gears. The servo-motor is limited to 180 ° and has an built-in potentiometer for easy control and feedback. The motor is, however, quite weak which is why a transmission ratio,  $i$ , of 2 and an extension spring is necessary to enable seamless movement of the motor. A transmission ratio of 2 will increase the torque of the driven gear by a factor of two with respect to the motor torque according to equation 1 [32].

$$i = \frac{T_{out}}{T_{in}} \quad (1)$$

Where  $T_{in}$  is the motor torque and  $T_{out}$  is the resulting torque of the driven gear. The extension



spring is furthermore attached to the inside of hatch of the carrier replica and is pre-tightened to help the motor in critical situations.

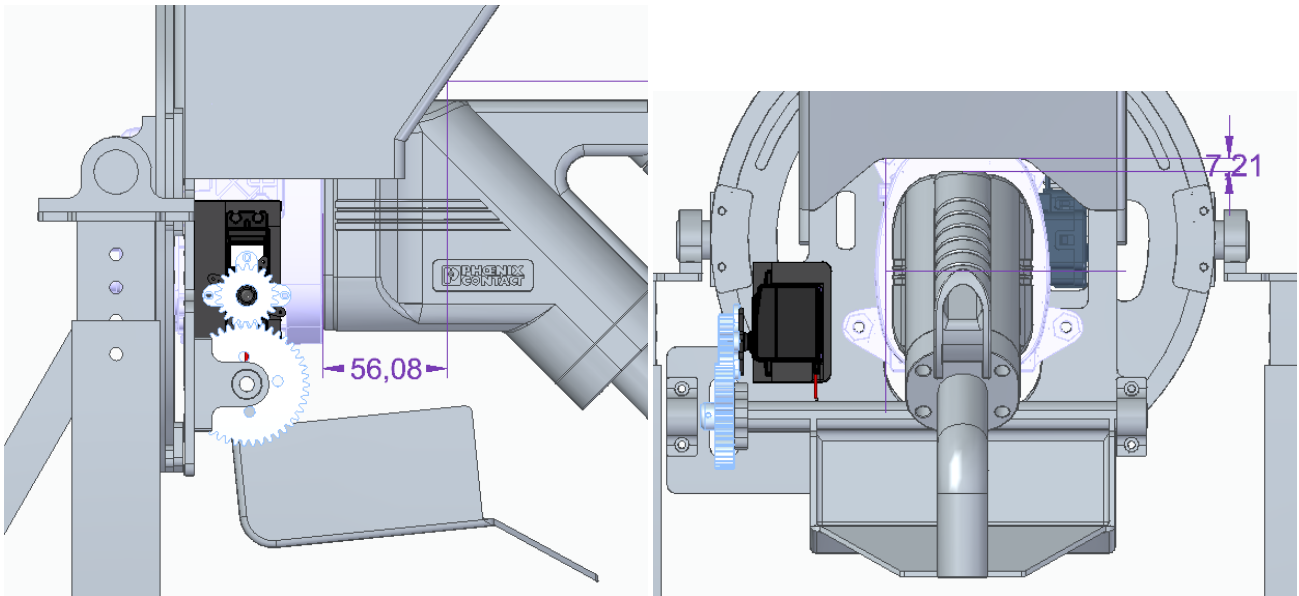


Figure 47: Left: Side view of the carrier replica. Right: Behind view of the carrier replica. Made in Solid Edge.

One can see, from figure 47 that the critical dimension in the solution assumes values lesser than the critical dimension in the supplied port assembly. The prototype can therefore be assumed to be a valid replica of the supplied Volvo port. The port in the carrier prototype is furthermore attached to a circular back plate which is guided by three plates in a track allowing it to rotate when you loosen two screws, see figure 48 and 49.

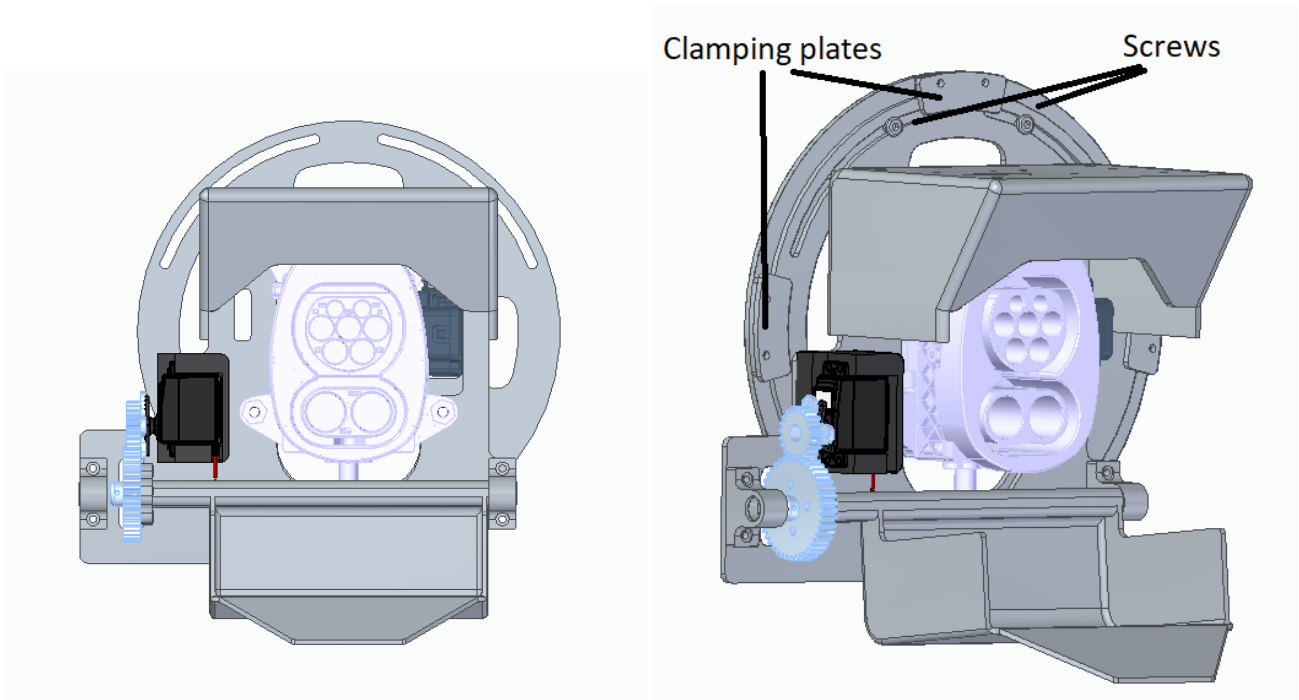


Figure 48: Left: A sub assembly of the carrier replica including the port and circular back plate etc. Right: A sub assembly showing the guide plates and screws used for clamping. Made in Solid Edge.

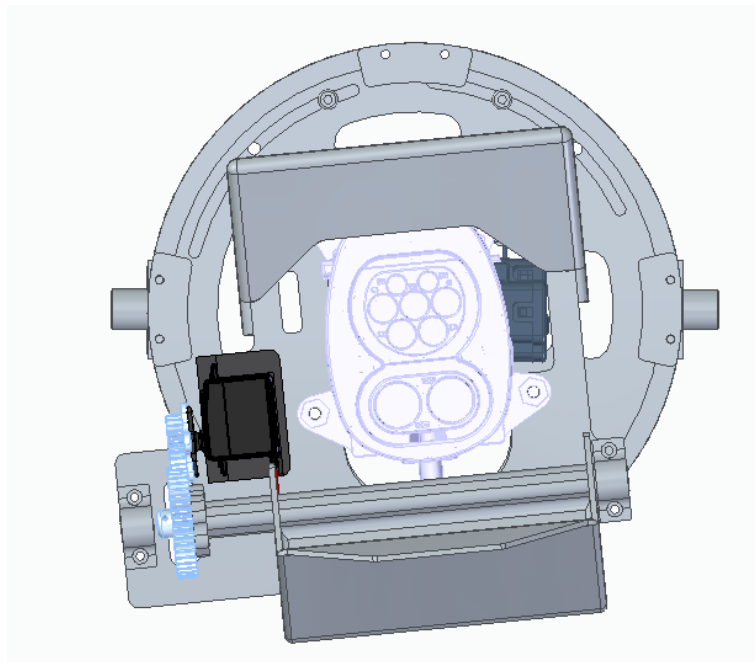


Figure 49: The port solution rotated (roll). Made in Solid Edge.

The construction is moreover attached to two joints also allowing rotation. These joints consist of two partly beveled axles, four axle-holders and two stop screws, see figure 50.

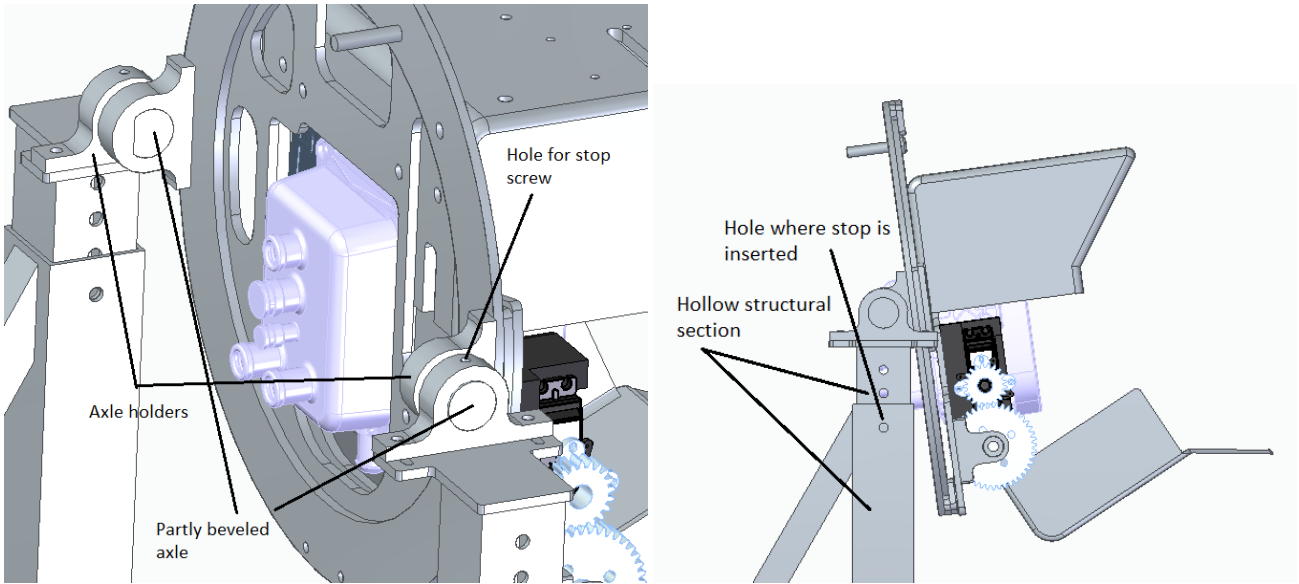


Figure 50: Left: The joint specifications. Right: Rotated view of the port solution (Pitch). Made in Solid Edge.

Observe that rotation around the last axis (yaw) is solved by rotating the entire construction. Adjusting the height of the port solution is accomplished by a square pipe with holes moving in a bigger section with one hole, allowing the height to be adjusted by inserting a stop in the hole desired, see figure 50.

The solution was produced in PLA-plastic, aluminium and steel. The full-scale prototype is shown in figure 51.

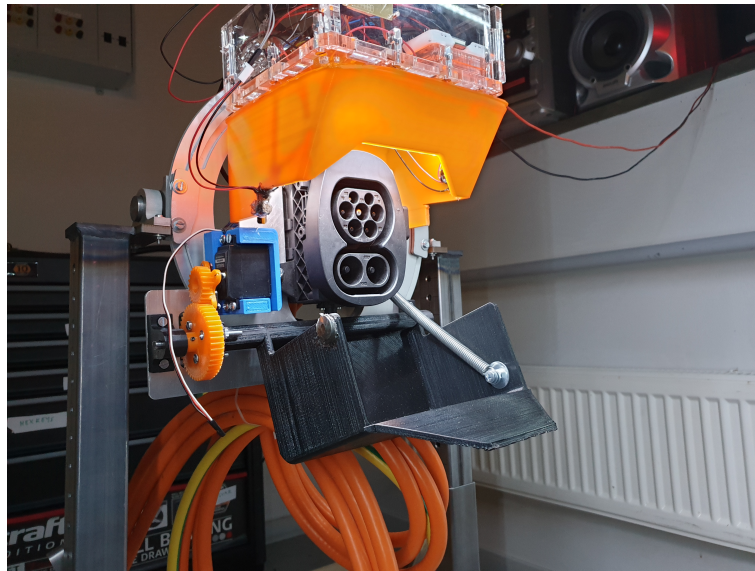


Figure 51: The full scale prototype.

#### 4.2.6 Station

Some kind of frame is necessary to mount all the components onto. Therefore, an aluminium frame was built. The geometry of the frame is designed in a box-like manner which is were

the components (actuators, switches etc.) are attached, with a rain cover extending out of the front where the charging handle is located. It is made out of several *Item* aluminium profiles, which are a type of aluminium profiles with pre-made grooves on the sides. The advantage of constructing the station with the *Item* aluminium profiles is that there are several pre-made attachments made for easy assembly of the profiles. This results in a robust and lightweight construction. Observe that the roof and sides of the frame have the possibility to be covered for weather protection. Since the prototype is made to show the concept the sides are left open for visibility of components and functionality. The frame of the station with everything attached can be seen in figure 52.

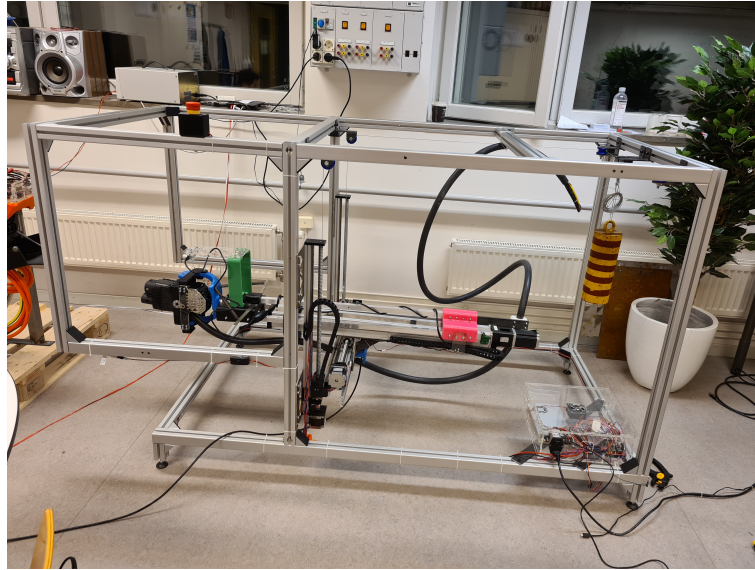


Figure 52: The charging station.

## 4.3 Electrical Design

This chapter goes into depth regarding the electrical design of the station as well as the carrier replica.

### 4.3.1 Station

The four stepper motors which are a part of the linear actuators were controlled using TMC2208 SilentStepSticks that were connected to an Arduino Uno via a GRBL stepper shield. The shield was originally designed to control CNC machines but it is also suited our purposes. Originally it was planned to make a purpose built PCB for the drivers, mostly to protect them against reverse-polarity, flyback-voltage and other design quirks of the SilentStepStick.

Most of these problems could be solved using diodes, a vast number of them. Reverse-polarity was not a major problem since it was caused by operator error and for now there would be no operators.

The flyback-voltage was the biggest worry and is believed to have killed a *few* drivers in the early prototyping period. Mainly before the counter-weight was added, when the drivers lost power the whole linear-actuator assembly would drop and introduce a large reverse-polarity on the driver output pins.

The design quirk of the SilentStepStick is that if the motors are powered but the digital I/O is not, a current will flow from the motor supply through the digital circuit, destroying the IC. By the recommendation of the manufacturer a diode is placed between the positive sides of the motor supply and digital I/O.

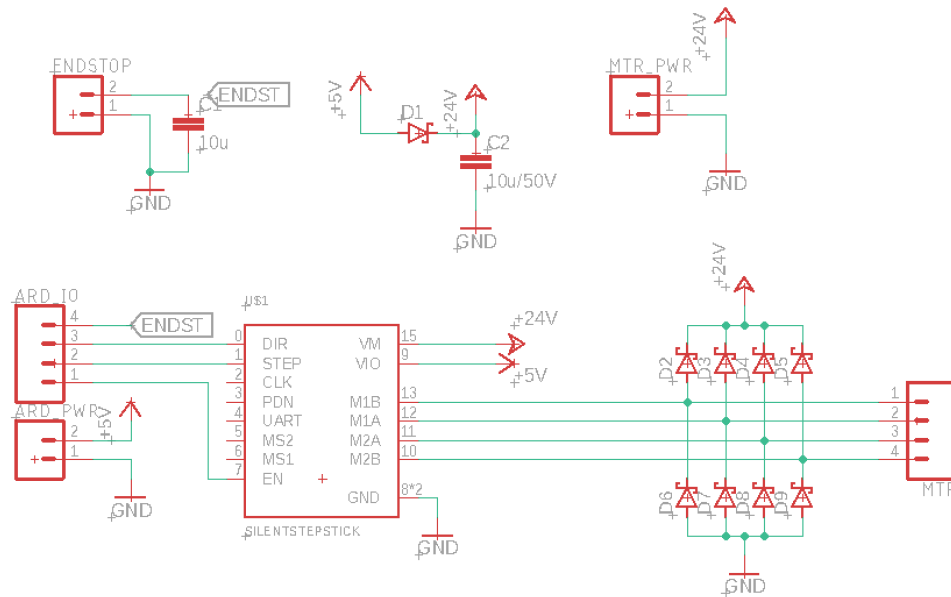


Figure 53: Planned stepper-driver protection circuit

The circuit was sadly never built since we did not really need the flyback-diodes after we added counter weights and the available PCB-machine was out of commission. Thus, the shield was utilized throughout the project.

There were also several *endstop* switches to set the zero position of the steppers. The long wires picked up a lot of noise so we added a low-pass filter on their inputs to clear up the signal and also to debounce the switches.

All these cables needed some appropriate cable-management so wire-guides were added to prevent wear and tear of the cables and our sanity.

Finally a PSU was purchased to power the stepper-driver circuitry, Raspberry Pi and the LED lighting that was added to the front. Since the PSU requires a connection to the main power line, some safety precautions were necessary. A fused inlet was added to the power supply, a big red button was attached to the frame to quickly cut power to the machine if something went wrong and finally since the station is almost entirely made from metal; an earth connection was attached to the entire structure to trigger the GFCI (Ground-fault circuit interrupter) on the outlet should there be a fault.

### 4.3.2 Carrier

To succeed with the assignment for the station and the carrier replica, the carrier replica has to have certain functions. The lid that covers the port has to be able to move up and down, which is achieved with a servo motor. The port and the handle has to be able to be locked

together before the charging starts in order to assure a solid connection is made. This is done by an internal piston on the port that is controlled by a DC-motor. It is also important to have information about whether the piston is locked or not. This is done with utilizing an internal potentiometer in the port. The camera that identifies the port must be able to see the port regardless of whether it is day or night. This is made possible with help of LED strips.

The actuators and sensors are controlled using an Arduino Nano. When the handle inserts into the port it can be known if the handle and port is connected. This is done by sending current through control pins on the port. When the port and handle are fully connected, a circuit is established, and the Arduino can read the current that is going through the wires. The station and the carrier furthermore have to communicate wirelessly with each other, this is done with help of a Raspberry Pi. The Raspberry Pi is connected to the Arduino Nano with a USB cable and the Raspberry Pi communicates with the station via WiFi.

To be able to connect everything together, a PCB was made. The PCB connects all the actuators and potentiometers to the Arduino Nano, it also makes it possible to detect when the plug and handle are connected. A separate PCB was made for six LED's. The PCB for the LED's can be connected to the main PCB with help of cables. In figure 54 a sketch of the main PCB can be seen and in figure 55 the PCB for the LEDs can be seen.

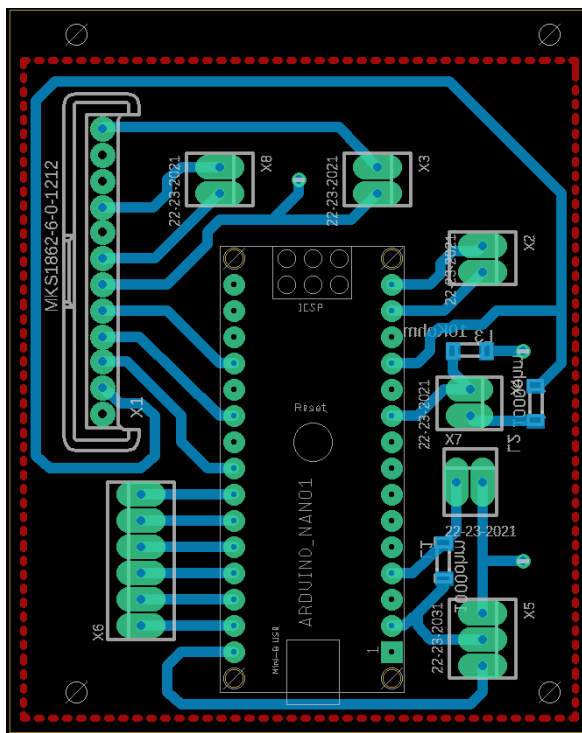


Figure 54: Sketch of the main PCB, Made in Eagle

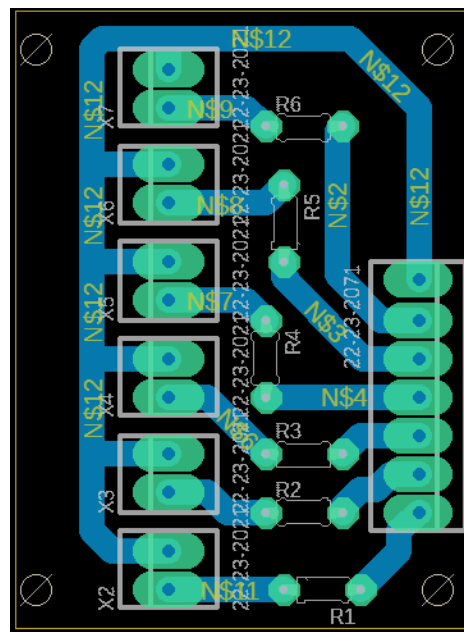


Figure 55: Sketch of the LED PCB, Made in Eagle

Volvo's load carrier supplies 24 V DC. The Arduino Nano needs 7 V - 12 V, the Raspberry Pi needs 5.1 V and 3 A. The motor that controls the piston needs 12 V - 24 V, the voltage and current for the potentiometer will be directly supplied from the Arduino. The LED strip needs 12 V and the servo motor needs 9 V. To be able to lower the voltage to the specific limit, three DC-DC converters are used. The three converters are connected in parallel from the 24 V



source.

## 4.4 Software Design

This chapter goes into depth regarding the software design of the station as well as the carrier replica.

In order to enable the full charging sequence, the system was divided into four subsystems - nodes - each consisting of software, electronics and mechanical hardware. The system consisted of the hauler node, the main node, the camera node and the arm node. The first node is located on the hauler, and the three remaining nodes make up the station.

### 4.4.1 Station

The main node was implemented as a high-level control software for the other nodes, running on the station PC. It implements the full sequence needed for AMPR to connect the charging plug to the charging port, and to later unplug the cable. It communicates with all other nodes, reading their respective statuses and outputs, while giving them instructions based on what is needed to continue whatever part of the sequence is in progress. It is written in *Python*, and makes use of *Rospy* to enable communication with the other nodes via ROS. A simplified description of the full sequence that was implemented is shown below.

- Hauler node opens the port lid, communicates this to main node
- Arm node connects plug to port using feedback from modified camera node data
- Hauler node detects and communicates that cable is fully connected
- Hauler node locks cable into place, communicates this to station
- Main node waits for set amount of time to simulate charging
- Main node asks hauler to unlock handle
- Arm node unplugs charging cable by retracting arm
- Main node communicates to hauler that arm is retracted
- Hauler node closes lid

The code and a more detailed algorithm description is located in Appendix B.

The Arm node is the node for the Arduino controlling the stepper drivers. It communicates via serial to the station's Raspberry Pi using ROS Serial.

The control of the Stepper drivers is mostly achieved by using a library called AccelSteppers that is wrapped into a class that handles the simultaneous control of all axes. The class also handles unit conversions from steps to millimeters, endstop switches, state-management and various capabilities. The idea was to isolate the stepper control from the ROS communications to prevent any user input from damaging it. Unfortunately the ROS serial package did function with it as desired so it had to be deconstructed and hot-fixed to run as intended.

The camera node consists of a camera, recognition software and a PC. The purpose of the camera node is to detect the position of the port relative to the position of the camera mounted on the arm, and to communicate this to the main node. An in-depth description about the recognition software and hardware is provided in section 4.5. It utilized *Python* and *Rospy* to communicate via ROS.

#### 4.4.2 Hauler

The code for the hauler is written in C++ and is executed on an Arduino Nano. The communication between the hauler and station is done via WiFi, which the Arduino Nano does not have. Because of this, a Raspberry Pi is used for communication between the hauler and the station. The communication between the Arduino and the Raspberry Pi is done with a USB cable and with the help of Rosserial communication as mentioned in the previous section.

The arduino code has two overall purposes. The first purpose is regarding the communication. It publishes and subscribes to specified topics, the code sends and receives data to these topics. The second purpose for the code is to use the electronics to help succeed with charging the hauler. In figure 56 a flowchart can be seen that explains what the hauler will have to do, to make the charging succeed. The actual code can be found in appendix: 9.1

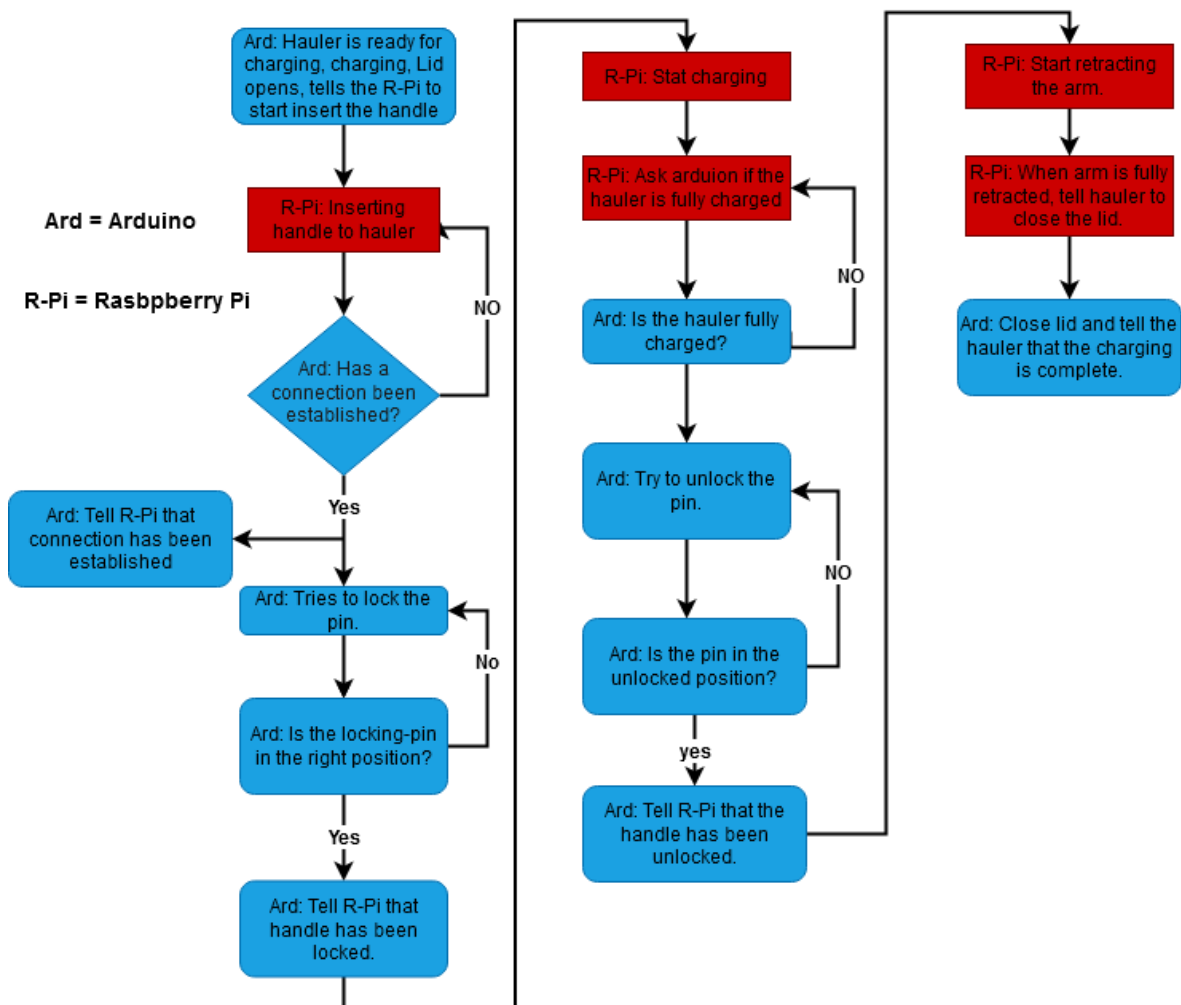


Figure 56: Flowchart of the code for the hauler, Made in Drawio



To make it easier to follow the different steps and to detect when an error has occurred, six LEDs were programmed to light up in a certain way depending on the steps and possible errors. The list below shows how these LEDs will light up and what they will indicate:

- LED 1: Lid has been opened.
- LED 2: Connection has been established between the haulers port and the stations handle.
- LED 3: Locking with the pin has been succeeded.
- LED 4: The unlocking has been succeeded.
- LED 5: Lid is closed.
- LED 1-6 lights up with 0.5 seconds apart, over and over again: This indicates that the hauler is charging.
- LED 1-3 blinks: The pin has not been able to lock.
- LED 1-3 is on and LED 4-6 blinks: The pin has not been able to unlock.

## 4.5 Recognition

The evolution of the recognition program began with a model of template matching, with a pattern created from the contours of the port. This method was rejected since the process was relatively slow and with low accuracy. A model of HAAR cascade classification was tested which reached a higher accuracy but had a standing spread of position, and therefore it was difficult to find a definite pixel coordinate of the port. Furthermore, a CNN model was created which reached a 95% accuracy in tests but still had a long processing time, and was dependent on high computer power.

The recognition of the port ended up with a faster-RCNN model based on TensorFlow model **faster\_rcnn\_inception\_v2\_pets** [42]. A total of 2840 images were used in the training process, with varying backgrounds and brightness. It also includes images of objects that were not the actual charging port. This was done to enhance the knowledge of the model to distinguish between background objects and the port. The model was trained to identify seven different classes, and the actual port was classed as "plug". The distribution was approximately 20% of the data in the test directory, respective 80% in the train directory.

Table 3: The distribution of images for each classification

Class	Test images	Train images
Plug	417	1512
person	132	375
Incorrect Plug	44	183
Raspberry_Pi <sub>3</sub>	9	34
Arduino_Nano	9	34
ESP8266	9	38
Heltec_ESP32_Lora	9	35
<b>Total</b>	<b>593</b>	<b>2247</b>

The training process iterated with 200 000 steps until it reached its final form.

## 4.6 Code Optimization

The aim was to run the detection code on a Raspberry Pi which does not have a separate GPU and has limited CPU power. Even with overclocking, the original code was unable to function at sufficient speeds. The main delay in processing is when the captured frame from the video stream is passed through the layers of the neural network, also called inference. To be able to reduce this time delay the model was converted to an OpenCV format which offers more functionality and optimization options [41]. Once the model was converted, the detection process could be split into two parts: loading the model and inference. The loading of the model was placed outside of the detection loop, as the model only needed to be loaded once. This meant that during the real time detection only the inference was occurring each time a new frame was used. This sped up the process and the inference per second rate was increased from around one to over five.

The OpenCV format also allowed for the use of multiple detection models to be used at the same time. This meant we could load both a short and long range model, then infer the image through whichever model was most appropriate. Through this method we could optimise the model used and create a more reliable and faster detection system.

The model was also converted to the OpenVino format. However, the RCNN model used was too complex and had too many layers to be able to fully optimise and compress to a faster format. Using the neural compute stick only yielded an inference per second rate of 0.65 which was insufficient. The stick could still be used to carry out the neural network back-end calculations however, so it can still help to reduce the load on the CPU.

When the port is identified, the function outputs the center point of a box that encapsulates the port. The red square and red dot shows this process during a live stream from the camera in Figure 57.

The red dot represents the center pixel point of the square, its 2D coordinates are used together with the function `rs2_deproject_pixel_to_point()` from Intel Pyrealsense2 library. This converts the 2D pixel point to a 3D coordinate system with the origin in the center of left imager, that is positioned on the right side of the camera. The output can be shown in the Figure 57 with the green text in the upper left corner which corresponds to the distance in millimeters from the camera to the port.

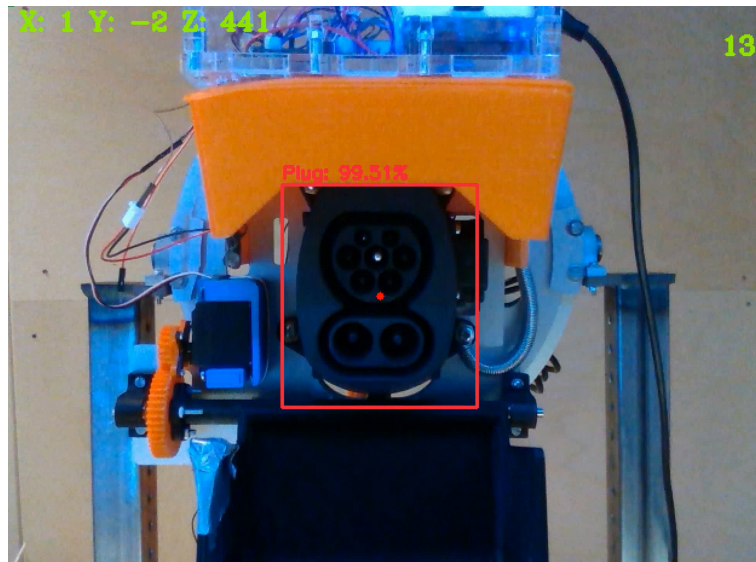


Figure 57: Camera view with recognition of the port.

## 5 Verification and Validation

This section presents how verification and validation methods were applied throughout the project to make sure that the goals were accomplished. The results of the system level tests are presented in the next section.

In order to enable the full sequence, the system was divided into four subsystems - nodes - each consisting of software, electronics and mechanical hardware. The system consisted of the hauler node, the main node, the camera node and the arm node. The first node is located on the hauler, and the three remaining nodes made up the station. Each node was individually tested to verify their respective functionality, before they could be integrated together.

The individual nodes were verified first by testing that their corresponding software behaved as expected, while faking any electrical connections, sensor readings and communication from other nodes. In parallel, the electronics and mechanical solutions were tested to verify that they functioned as expected. Once the software, electronics and mechanical solutions (e.g. the lid servo in the case of the hauler node, the camera in the case of the camera node etc.) for each node was verified, they were integrated together. Once this was done, the nodes could be integrated with main node.

The main node, as described in section 4.4.1, communicated with the hauler, camera and arm nodes, and acted as a high-level control software for these nodes. The node was first specified in terms of what functionality it needed to have to enable the full sequence to take place. The node then used test stubs (fake nodes to emulate the responses from other nodes), so that the software design could be verified in isolation. Once the main node software was verified, partial integration and full integration could be carried out.

The partial integration took place by integrating one of the other nodes (hauler/camera/arm) with the main node, while using the test stubs to simulate the behaviour of the non-integrated nodes. Once each node was verified with the main node, the integration was expanded to include all nodes. Once the behaviour of all nodes together was verified, the full system could be verified and finally validated in regards to the set requirements. The validation of the full system was organized as a set of tests to validate that the stakeholder and technical requirements were met. The tests and results of these are detailed in section 6.

To validate the solution generated, a carrier replica was furthermore constructed, since the HX02 load carrier was not available. The carrier replica was designed using the critical dimensions of the HX02 load carrier port affecting plug insertion. A successful sequence using this carrier replica would thus validate our results and would imply success when implementing the solution on the HX02.

## 6 Results

This chapter presents the results of the project at hand. The chapter states whether the project requirements are fulfilled or not as well as articulate the capabilities of the prototypes produced.

### 6.1 Requirements

The stakeholder and technical requirements were tested in order to validate the prototype. The subsequent sections detail whether or not the requirements were fulfilled, and how this was determined.

#### 6.1.1 Stakeholder requirements

Below is a table listing the stakeholder requirements, and whether they were fulfilled or not with a Pass (P) or Fail (F) rating.

Table 4: Table declaring whether stakeholder requirements were met

Stakeholder requirement	P/F	Comments
SR1: The structure shall manage to operate in harsh weather conditions and harsh environments	-	Stakeholder removed requirement in late stages of project, but prototype was designed with this requirement in mind.
SR2: The charger plug should be based on an existing connector	P	Based on CCS-combo 2
SR3: Project cost should be minimized	P	Cost was factored in and discussed with stakeholder during entire project
SR4: A charging port should be positioned on the HX02 load carrier	P/-	Did not have access to HX02, but solution was based on HX02 design
SR5: The structure shall have support for a charger with a power level of at least 150 kW (for 650 V system)	P	CCS-combo 2 supports >150 kW, but the cable provided by sponsor supported 125 kW. Simple to exchange however.
SR6: The structure shall be able to charge a slightly misaligned HX02	P	See technical requirements tests below
SR7: The structure shall be able to align to the port enough create a solid connection	P	See technical requirements tests below
SR8: The station shall be able to communicate with the HX02	P/-	Did not have access to HX02 proprietary communications system, but communicated wirelessly, like said system
SR9: Aesthetics shall be taken into consideration when designing the structure	-	Stakeholder removed requirement
SR10: The connector shall be interchangeable	P	Connector is replaceable as long as there are adapter parts manufactured for other connectors.

### 6.1.2 Technical requirements

The following table declares whether the technical requirements were fulfilled or not, see table 5. Observe that some the requirements are left out. These requirements are in need of additional investigating and will be examined in section 6.2.

Table 5: Table declaring whether the technical requirements were met.

Technical requirement	P/F	Comments
TR1: The charging station shall be able to carry the weight of the cable + own weight + safety margins	P	
TR2: The charging station shall be able to extend at least 500 mm from its starting position	P	The charging station can extend 500 mm
TR3: The charging station shall be able to move in a minimum range of 200 mm horizontally	P	The charging station has a range of 430 mm horizontally
TR4: The charging station shall be able to move in a minimum range of 100 mm vertically	P	The charging station has a range of 475 mm vertically
TR6: The charging station shall not take up more than $2.5 m^3$ and not weigh more than 300 kg	P	The charging station has a volume of $2.2 m^3$ and weighs approx. 100 kg
TR7: The charging station shall have a capability to charge up to 150 kW	F	The plug used can only supply up to 125 kW, but the station allows change of the plug to higher power
TR9: The charging station shall be equipped with at least one camera	P	
TR10: The charging station shall be able to communicate with the main server host	P	
TR11: The charging station shall have protection, at least of standard IP13 with possibility for improvement	F/-	The prototype produced does not supply any weather proofing, it is however easy to apply a casing to the prototype

The technical requirements regarding the carrier were not taken into consideration due to the stakeholder being in charge of the carrier.

## 6.2 Testing

All tests regarding the rotational positions of the port were made with the charging port at the same distance from the station, which was 600 mm . Pitch is defined as rotation around the X-axis, yaw is defined as rotation around the Y-axis and roll is defined as rotation around the Z-axis. Positive angle is clockwise around respective axis and negative angle is counter-clockwise around respective axis.

### 6.2.1 Image Recognition

The following requirements for image recognition are:

- TR5: The charging station shall be able to handle a  $3^\circ$  deviation around all axes
- TR8: The charging station shall be able to detect the port from a distance of 600 mm and through iterations, obtain its coordinates within  $\pm 10$  mm.

were investigated further. The table below, see table 6, analyses the requirements given. A number 1 through 5 was given to each scenario. The following rating was given:

- 5 : Detection without any difficulties
- 4 : Detection with minor difficulties
- 3 : Detection with major difficulties
- 2 : Spontaneous detection, not reliable
- 1 : No detection at all

Table 6: Table presenting the results when investigating the angular deviation capabilities of the camera detection

Requirement	1-5	Comments
Pitch angle deviation of $+3^\circ$	1	Probably not able see the full outline of the port, and thus fails
Pitch angle deviation of $-3^\circ$	4	Took a while to detect the port but eventually it did
Roll angle deviation of $+3^\circ$	4	
Roll angle deviation of $-3^\circ$	4	
Yaw angle deviation of $+3^\circ$	3	There was a good detection but it was slightly offset, leading to a more difficult mechanical alignment. Yaw rotation resulted in faster dead on alignment than straight forward
Yaw angle deviation of $-3^\circ$	3	Yaw rotation resulted in faster dead on alignment than straight forward

### 6.2.2 Lighting

Lighting tests were made in order to find what the capabilities of the camera were. The findings are presented in figure 58. The lighting tests were made at the same distance between the port and the plug. Different lighting conditions were made in the room. Lights in the ceiling were turned on, halfway on, and off. The lights on the frame were turned on and off. The lights in the port were turned off and on. A pink square around the port means that it is detected.

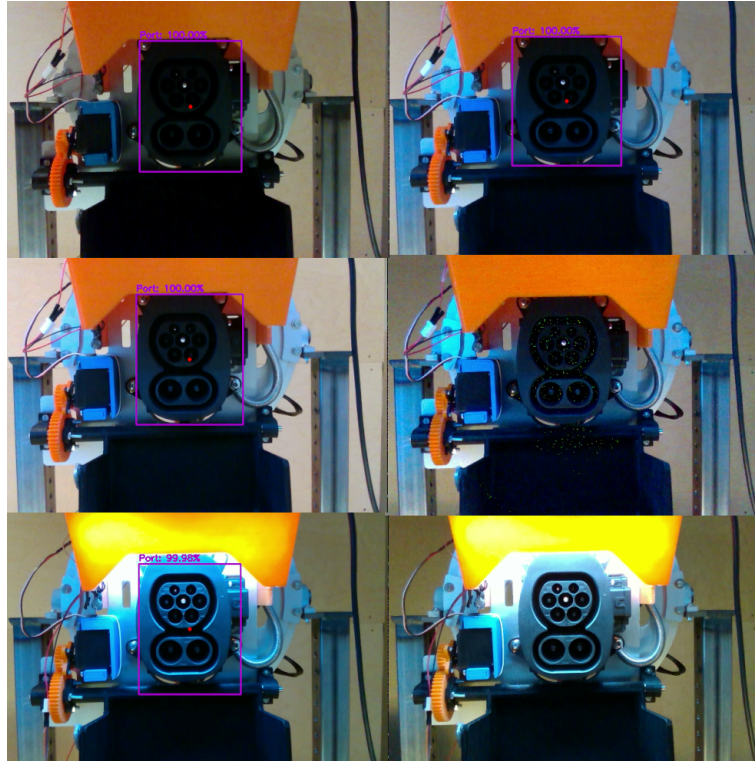


Figure 58: Port detection tests with different lighting.

### 6.2.3 Connector Angle Deviation

The connector is, as mentioned, equipped with springs to handle angle deviations of the port. The following requirement

- The charging station shall be able to handle a  $3^\circ$  deviation around all axes

was investigated to analyse the degree of success regarding the connector construction. A number 1 through 5 was given to each scenario. The following rating was given:

- 5 : Fully inserted without any difficulties
- 4 : Inserted with minor difficulties
- 3 : Inserted with major difficulties
- 2 : Not inserted but made contact with the outside of the port
- 1 : Missing the port completely



Table 7 presents the results of tests made for the connection of different angular deviations.

Table 7: Table presenting the results when investigating the angular deviation capabilities of the connector solution

Requirement	1-5	Comments
Pitch angle deviation of $+3^\circ$	5	
Pitch angle deviation of $-3^\circ$	3	Inserted but not all the way
Roll angle deviation of $+3^\circ$	4	
Roll angle deviation of $-3^\circ$	3	Got stuck but snapped into place
Yaw angle deviation of $+3^\circ$	5	
Yaw angle deviation of $-3^\circ$	3	Got stuck but snapped into place

#### 6.2.4 Accuracy

The same test was done five times when evaluating the accuracy. This was done by placing the port in a fixed position with no rotational deviation around any axes. This is to ensure that it could complete the same task over and over again, so it was no coincidence.

Table 8: Table presenting the results when investigating the accuracy of a connection when the port is facing straight forward

Run	Success (S) or Fail (F)	Number of adjustments	Comment
1.	S	8	
2.	S	14	
3.	S	8	
4.	S	10	Bumped into the bottom and then snapped up
5.	S	21	

The number of adjustments means that each time it has found the port and moves to a position that it thinks is more suitable in order to get a better reading of the port. Fewer amount of adjustments means that it finds good readings of the port quicker. It is satisfied with its reading once the port is in the center of view and a certain distance from the camera, which it adjusts by sending coordinates to the motors.

## 7 Discussions and Conclusions

All technical requirements, except one, were accomplished. The requirement that was not met was regarding weather proofing. This requirement was dropped quite early in the development process, with Volvo's consent, since it was determined that this was not as important as the rest of the requirements. The focus was on achieving a well functioning prototype rather than one that would be weather proof but not reaching the other requirements. Developing a weather proof prototype would be more expensive and time consuming. Although, the station is made for easy fitting of weather protection if that would be desired at a later stage.

Most technical requirements were met with a slight margin in order to satisfy the stakeholder. Two out of the three translation-range requirements were met with a really large margin. The reasoning behind this was that it was better to build a prototype capable of achieving more than needed, since the hauler was still not developed by Volvo and the port location was therefore still not determined.

The two requirements for image recognition are fulfilled to some extent. With an angle deviation, the system could recognize the port in all cases except at a pitch of +3 degrees. But none of the cases could detect without some difficulties. Mostly the difficulties consisted of miscalculations in the form of distance assessments. Which then furthermore affected the interconnection itself. The other requirement of detecting the port from a distance of 600 mm  $\pm$  10 mm, was also fulfilled, but it was noticeable that once it detected the port it soon after lost the detection. So one could say that even though detection occurred it was not a stable detection.

Though the system only shows a proof of concept and to get a more robust system, the model needs to be trained with thousands more images in more varied conditions in the form of environment and light saturation. It is noteworthy that the system has a certain deadband at the moment where the effects of twilight or certain distances can limit the detection.

Throughout the project the desired angular deviations have been unknown since, as mentioned, Volvos hauler was under construction. Therefore, the angle of 3 degrees back and forth in roll, yaw, pitch has been estimated. In both pitch and yaw the plug inserts quite well, especially in positive angles. This could be due to how the charging cable is attached. It could also be due to the twist of the cable in one direction while clamping the cable down, this could cause uneven torque while rotating in the different directions. The cable as mentioned, is thick, it is also difficult to twist and spring back, therefore a completely equal and balanced attachment is hard to obtain. Another fact causing this could be that the springs are not attached symmetrically. The springs are attached asymmetrically to compensate for the cable to get the handle to rest completely straight. The station had a harder time to insert the plug with a 3 degree roll in both directions. It could be caused by several factors; from how the cable is attached to the bearing and springs for that direction. One factor that could play a major role is the design of the plug and connector since it might cause less torque for self correction in different directions.

## 8 Future Work

### 8.1 Plug

The motion of the plug is quite limited due to the weight and stiffness of the cable. This issue could be handled by using a smaller cable. However, this would reduce the amount of power that the station could deliver. Adding a funnel mechanism could also be a possibility, making it easier for the plug to adjust itself to the port. The effects of the plug tending to rotate more to one side due to the stiffness of the cable could be minimized by having a mechanism that could adjust the tension of the springs. In this way it would be even easier to compensate for the stiffness of the cable.

### 8.2 Construction of station

One way to make the whole construction more stable and robust would be to replace plastic parts to metal parts. Although, most parts made in plastic work well in the current configuration.

One part that is especially important to remake in metal is the part that connects the square pipes for the handle to the moving carrier on the outwards going actuator. The weight of the handle and cable made the outwards arm sag a little bit, which could be caused by the flexibility in plastic. The part can be seen in figure 37. Another factor that might cause the outwards arm to sag is the rigidity in the carrier itself. This could be improved by having an actuator with a more rigid carrier or one with a double carrier setup.

For weather protection, walls around the casing would be needed. This has intentionally been left aside primarily due to the fact that it is good to be able to see the inside of the station while demonstrating. The prototype will not be used anywhere else than inside and thus no casing on the sides is needed. The station is instead made in a way that allows for easy fitting of a casing around it, if it would be required at a later stage.

Finally the aesthetics of the station is an area for improvement. During the project the function of the station has been the main focus and thus the aesthetics have been left aside. For example, some plastic parts are made out of different colors due to the availability of colors for the 3D-printers at that moment. The design could be improved on, when or if, version two of the prototype is manufactured.

### 8.3 Camera

The camera produced a sufficiently clear image in most scenarios. The depth also allowed for the accurate measuring of the port position. A more advanced camera with higher accuracy and better low light features could be used for more precise positioning.

If the whole station was to be made weatherproof then there would need to be modifications made to the camera setup. Currently the camera is also not weatherproof so more robust option could be found or a case would need to be built to house it.

There is also the possibility of using two or more depth cameras. If there were additional cameras positioned at an outside vantage point then it would be possible to see the plug as it

enters the port and verify that all parts are aligned or adjust, if necessary.

More cameras could also be used to create a complex 3D textured point cloud. From which the port position and orientation could be found. This would allow for precise corrections when inserting which are tailored to the port orientation. However additional cameras would require a lot more computational power and neural network training to be able to achieve an accurate result.

## 8.4 Neural network

The neural network used for object detection was a faster RCNN. However there are other alternatives which could yield better results. One such alternative is a Mask RCNN [5]. The desired object is identified in each image, by using a blob which defines the outlines. This method is a lot more accurate and faster than the current method used. However it requires a lot more time to prepare each image for the model. Another alternative is the "You Only Look Once" method or YOLO [37]. This is a method which is a regression-based one stage method [28]. It is faster than the method used so more appropriate for the real time applications. However it is harder to train and can have reduced accuracy. Future work would be to test out these other object detection neural network alternatives and see which is the fastest with an acceptable accuracy.

Retraining of the model would also improve the current accuracy. The current model is sensitive to variations in lighting and coloured objects around the port. With further input images and training of the model, a more robust model could be created. There is also the option to use the 3D point cloud image from the camera for the detection algorithm. This would completely eliminate the light variance factors but could introduce some new problems when trying to obtain a clear depth image. Further research into the depth image would be required to determine if it could be implemented with one of the previously mentioned neural networks.

## 8.5 Micro-controllers

The aim was to use a Raspberry Pi with the Intel Neural Compute Stick as the main micro-controller to run the detection. However, when a faster RCNN is used for the object detection, there is not enough processing power for real time detection of the port. Therefore, a more powerful micro-controller is needed. Most likely one with a separate GPU or VPU which is capable of running the inference from the video stream in real time.

Possible solutions are one of the Nvidia Jetson boards which utilise CUDA to increase performance. Another solution is to use an Intel board which already has multiple Myriad X VPUs, from the stick, built into the board. These board alternatives are far more expensive than a Raspberry Pi but would give greater possibilities for the detection and would be beneficial if using some of the other neural networks mentioned above, such as YOLO or mask RCNN.

## 8.6 Implementation for Other Vehicles

This station is designed for the HX02 from Volvo CE. But it can be used for other vehicles, if the port is at approximately the same height and they use the same kind connector as the hauler. Future work could be to add adjustable feet. The current handle solution makes it hard

to switch out the handle. Further work could also be to implement an easy way to switch out the handle to make the station function with many more types of vehicles.

## References

- [1] A.Hosseini and A.Karlsson et. al. *Ericsson A.R.M.S.* 2018. URL: <https://www.kth.se/social/files/5e15ba3356be5b267b3d522c/ericsson-arms-finalreport.pdf> (visited on 05/05/2020).
- [2] A.Mordvintsev et. al. *Hough Line Transform.* 2013. URL: [https://opencv-python-tutroals.readthedocs.io/en/latest/py\\_tutorials/py\\_imgproc/py\\_houghlines/py\\_houghlines.html](https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_houghlines/py_houghlines.html) (visited on 05/05/2020).
- [3] H. Brunner B. Walzel and M.Hirz et. al. *Robot-Based Fast Charging of Electric Vehicles.* 2019. URL: [https://www.researchgate.net/publication/332156996\\_Robot-Based\\_Fast\\_Charging\\_of\\_Electric\\_Vehicles](https://www.researchgate.net/publication/332156996_Robot-Based_Fast_Charging_of_Electric_Vehicles) (visited on 04/30/2020).
- [4] *Canny Edge Detection.* 2020. URL: [https://docs.opencv.org/trunk/da/d22/tutorial\\_py\\_canny.html](https://docs.opencv.org/trunk/da/d22/tutorial_py_canny.html) (visited on 05/05/2020).
- [5] "Computer Vision Tutorial: Implementing Mask R-CNN for Image Segmentation (with Python Code)".
- [6] *Computer Vision: What it is and why it matters.* en. Library Catalog: [www.sas.com](http://www.sas.com). URL: [https://www.sas.com/en\\_us/insights/analytics/computer-vision.html](https://www.sas.com/en_us/insights/analytics/computer-vision.html) (visited on 05/05/2020).
- [7] Intel Corporation. *Intel RealSense Depth Camera D435.* 2020. URL: <https://www.intelrealsense.com/depth-camera-d435>.
- [8] C.Steitz. *Plug wars: the battle for EV charging supremacy.* Fig. 2018. URL: [https://www.autoblog.com/2018/01/24/plug-wars-the-battle-for-ev-charging-supremacy/?guccounter=2&guce\\_referrer=aHR0cHM6Ly93d3cucGludGVyZXN0LmNvbS8&guce\\_referrer\\_sig=AQAAALy2h4EUD56Etbh83h-2ruJr\\_mncEwfFt6xXE07-IR8aC4ydZ4I5zjPRXR4Bvh6IepkY9L1nCqbju6vTFDekerAbozY0ZRe0VV8T3iYUzMi4g1ma\\_2Hc2-g4bzkYAJJ\\_I2b5uTjxPz7ZQgYZvYZzGML0eY](https://www.autoblog.com/2018/01/24/plug-wars-the-battle-for-ev-charging-supremacy/?guccounter=2&guce_referrer=aHR0cHM6Ly93d3cucGludGVyZXN0LmNvbS8&guce_referrer_sig=AQAAALy2h4EUD56Etbh83h-2ruJr_mncEwfFt6xXE07-IR8aC4ydZ4I5zjPRXR4Bvh6IepkY9L1nCqbju6vTFDekerAbozY0ZRe0VV8T3iYUzMi4g1ma_2Hc2-g4bzkYAJJ_I2b5uTjxPz7ZQgYZvYZzGML0eY).
- [9] Dejan. *How a Stepper Motor Works.* 2015. URL: <https://howtomechatronics.com/how-it-works/electrical-engineering/stepper-motor/>.
- [10] Engineering on a desk. *Industrial robots.* n.d. URL: [http://engineeronadisk.com/V2/book\\_integration/engineeronadisk-14.html](http://engineeronadisk.com/V2/book_integration/engineeronadisk-14.html).
- [11] Dhanak. *CC BY-SA 3.0.* URL: <http://creativecommons.org/licenses/by-sa/3.0/viaWikimedia>.
- [12] *Differences Between Geometric Matching and Pattern Matching - National Instruments.* URL: <https://knowledge.ni.com/KnowledgeArticleDetails?id=kA00Z000000kGRjSAM&l=sv-SE> (visited on 05/05/2020).
- [13] Dorodnic. *Depth from Stereo.* 2018. URL: <https://github.com/IntelRealSense/librealsense/blob/master/doc/depth-from-stereo.md>.
- [14] Electrokit. *Ultrasonic Transducer receiver.* <https://www.electrokit.com/en/product/ultrasonic-transducer-reciever/>. Retrieved 2020-05-13.
- [15] E. Elisabeth. *Basics of Rotary Encoders: Overview and New Technologies.* 2015. URL: <https://www.machinedesign.com/automation-iiot/sensors/article/21831757/basics-of-rotary-encoders-overview-and-new-technologies>.
- [16] Elprocus. *Servo Motor – Working, Advantages Disadvantages.* n.d. URL: <https://www.elprocus.com/servo-motor/>.

- [17] Opto Engineering. *Introduction to infrared vision*. 2020. URL: <https://www.opto-e.com/resources/infrared-theory>.
- [18] B. Frank. *Meeting 2020-04-17*. Stockholm, 2020.
- [19] B. Frank. *Power Electronics in Future Construction Machines*. 2019. URL: [https://www.iea.lth.se/kel/2019/Lectures2019/L22\\_Power\\_Electronics\\_GuestAutomotive.pdf](https://www.iea.lth.se/kel/2019/Lectures2019/L22_Power_Electronics_GuestAutomotive.pdf) (visited on 04/27/2020).
- [20] B. Frank and J. Unneback. *The Mechatronics HK 2020 Project Descriptions*. 2020. URL: <https://www.kth.se/social/files/5e6b926e44aaf033eb4cdde7/hk2020-projectdescriptions.pdf> (visited on 04/27/2020).
- [21] H. Hashizume et al. "Fast and Accurate Positioning Technique Using Ultrasonic Phase Accordance Method". In: *TENCON 2005 - 2005 IEEE Region 10 Conference*. 2005, pp. 1–6.
- [22] The mobility house. *All the relevant charge cable and plug types*. 2020. URL: [https://www.mobilityhouse.com/int\\_en/knowledge-center/charging-cable-and-plug-types](https://www.mobilityhouse.com/int_en/knowledge-center/charging-cable-and-plug-types).
- [23] Volvo Press Information. *VOLVO CE UNVEILS THE NEXT GENERATION OF ITS ELECTRIC LOAD CARRIER CONCEPT*. 2017. URL: <https://www.volvoce.com/global/en/news-and-events/press-releases/2017/conexpo-vegas-2017/volvoce-unveils-the-next-generation-of-its-electric-load-carrier-concept/>.
- [24] "Intel Movidius Myriad X Vision Processing Unit".
- [25] *Introduction - NI Vision 2017 for LabVIEW Help - National Instruments*. URL: [http://zone.ni.com/reference/en-XX/help/370281AD-01/nivisionconcepts/pattern\\_matching\\_introduction/](http://zone.ni.com/reference/en-XX/help/370281AD-01/nivisionconcepts/pattern_matching_introduction/) (visited on 05/05/2020).
- [26] *Introduction - NI Vision 2017 for LabVIEW Help - National Instruments*. URL: [http://zone.ni.com/reference/en-XX/help/370281AD-01/nivisionconcepts/geometric\\_matching\\_introduction/](http://zone.ni.com/reference/en-XX/help/370281AD-01/nivisionconcepts/geometric_matching_introduction/) (visited on 05/05/2020).
- [27] A. Jain, Aishwarya, and G. Garg. "Gun Detection with Model and Type Recognition using Haar Cascade Classifier". In: *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*. Aug. 2020, pp. 419–423. DOI: 10.1109/ICSSIT48917.2020.9214211.
- [28] Zicong Jiang et al. *Real-time object detection method based on improved YOLOv4-tiny*. 2011.
- [29] J.Strickland. *What is a gimbal – and what does it have to do with NASA?* 2008. URL: <https://science.howstuffworks.com/gimbal.htm>.
- [30] M. O. Khyam et al. "High precision ultrasonic positioning using a robust optimization approach". In: *2013 18th International Conference on Digital Signal Processing (DSP)*. 2013, pp. 1–6.
- [31] Shaolong Ma et al. "Faster RCNN-based detection of cervical spinal cord injury and disc degeneration". en. In: *Journal of Applied Clinical Medical Physics* 21.9 (2020). \_eprint: <https://aapm.onlinelibrary.wiley.com/doi/pdf/10.1002/acm2.13001>, pp. 235–243. ISSN: 1526-9914. DOI: <https://doi.org/10.1002/acm2.13001>. URL: <https://aapm.onlinelibrary.wiley.com/doi/abs/10.1002/acm2.13001> (visited on 12/10/2020).
- [32] *Maskinelement Handbok*. KTH, 2008.

- [33] Justinas Miseikis et al. “3D Vision Guided Robotic Charging Station for Electric and Plug-in Hybrid Vehicles”. In: *arXiv:1703.05381 [cs]* (Mar. 2017). arXiv: 1703.05381. URL: <http://arxiv.org/abs/1703.05381> (visited on 05/05/2020).
- [34] OpenVino. *OpenCv About*. 2020. URL: [https://docs.openvinotoolkit.org/latest/openvino\\_docs\\_M0\\_DG\\_Deep\\_Learning\\_Model\\_Optimizer\\_DevGuide.html](https://docs.openvinotoolkit.org/latest/openvino_docs_M0_DG_Deep_Learning_Model_Optimizer_DevGuide.html) (visited on 12/09/2020).
- [35] *Picture of the Carla\_connect robot*. Fig. URL: [https://www.kuka.com/en-us/products/robotics-systems/kuka-ladeassistent-carla\\_connect](https://www.kuka.com/en-us/products/robotics-systems/kuka-ladeassistent-carla_connect) (visited on 05/14/2020).
- [36] Richard J. Radke. *Computer Vision for Visual Effects*. en. Google-Books-ID: MgIVyAWf9VUC. Cambridge University Press, 2013. ISBN: 978-0-521-76687-6.
- [37] Adrian Rosebrock. “YOLO and Tiny-YOLO object detection on the Raspberry Pi and Movidius NCS”. 2020. URL: <https://www.pyimagesearch.com/2020/01/27/yolo-and-tiny-yolo-object-detection-on-the-raspberry-pi-and-movidius-ncs/> (visited on 12/10/2020).
- [38] Azriel Rosenfeld. *Digital Picture Processing*. en. Google-Books-ID: FcXSBQAAQBAJ. Elsevier, June 2014. ISBN: 978-1-4832-9478-0.
- [39] G. V. Sivanarayana et al. “Review on the Methodologies for Image Segmentation Based on CNN”. en. In: *Communication Software and Networks*. Ed. by Suresh Chandra Satapathy et al. Lecture Notes in Networks and Systems. Singapore: Springer, 2021, pp. 165–175. ISBN: 9789811553974. DOI: 10.1007/978-981-15-5397-4\_18.
- [40] C. Sun et al. “Method for Electric Vehicle Charging Port Recognition in Complicated Environment based on CNN”. In: *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*. Nov. 2018, pp. 597–602. DOI: 10.1109/ICARCV.2018.8581197.
- [41] OpenCV Team. *OpenCv About*. 2020. URL: <https://opencv.org/about/> (visited on 05/08/2020).
- [42] *tensorflow/models*. en. URL: <https://github.com/tensorflow/models> (visited on 12/11/2020).
- [43] Random Nerd Tutorials. *Electronics Basics – How a Potentiometer Works*. 2016. URL: <https://randomnerdtutorials.com/electronics-basics-how-a-potentiometer-works/>.
- [44] *Understanding SFP Ports on SFP Switch*. URL: <https://blogg.improveme.se/felali/2018/05/25/understanding-sfp-ports-on-sfp-switch/> (visited on 05/18/2020).
- [45] P. Viola and M. Jones. “Rapid object detection using a boosted cascade of simple features”. In: *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*. Vol. 1. ISSN: 1063-6919. Dec. 2001, pp. I–I. DOI: 10.1109/CVPR.2001.990517.
- [46] *What is Pattern Matching? - Definition from Techopedia*. en. Library Catalog: [www.techopedia.com](http://www.techopedia.com). URL: <https://www.techopedia.com/definition/8801/pattern-matching> (visited on 05/05/2020).
- [47] *Why TensorFlow*. en. URL: <https://www.tensorflow.org/about> (visited on 12/10/2020).



## 9 Appendix

### 9.1 Appendix A - Code Hauler in C++

Hauler node on Github: [Hauler code](#)

## 9.2 Appendix B - Main Node Code

Main node code on GitHub: [\(link\)](#)