Bachelor Degree Project

HÖGSKOLAN I SKÖVDE

Build of an open source AGV

Bachelor Degree Project in Industrial Engineering
30 ECTS
Spring term 2022

Roberto Carlos BARNHARDT
Juan Carlos NÚÑEZ HERGUETA

Supervisor: Richard SENINGTON
Examiner: Stefan ERICSON

Certificate of Authenticity

This thesis has been submitted by Roberto Carlos Barnhardt and Juan Carlos Nuñez Hergueta to the University of Skövde as a requirement for the degree of Bachelor of Science in Industrial/Automation Engineering.

The undersigned certifies that all the material in this thesis that is not my own has been properly acknowledged using accepted referencing practices and, further, that the thesis includes no material for which I have previously received academic credit.

Place and date: Skövde, Sweden. 17/08/2022

Signature: Roberto Carlos Barnhardt Firmado

Signature: Juan Carlos Núñez Hergueta Firmado por NUÑEZ HERGUETA JUAN CARLOS - 26258835B el día 17/08/2022 con un

Högskolan i Skövde

Abstract

School of Engineering Science

Industrial Engineering

Build of an open source AGV

by Roberto Carlos BARNHARDT
Juan Carlos NÚÑEZ HERGUETA

Logistics has become one of the most important aspects in industry. In modern production, factories are dealing with throughputs which were unthinkable barely half a century ago. Having the required parts delivered to workers on time to continue assembling a product is a key factor for a company to maintain a production rate which can keep up with the market demand. In big factories, there are plenty of facilities where such AGVs help the workers with this challenging task.

Commercial **AGV** solutions are very costly, ranging from 200.000 SEK upwards of 1M SEK. The final aim of the overall project is studying the challenges associated **to providing an Automated Guided Vehicle (AGV) to Small and Medium-sized Enterprises (SME)** for a reasonable price using **O**pen **S**ource **H**ardware (**OSH**).

This is the continuation of a project where theoretical foundation and initial design was developed (Gámez and Martínez, 2021). Adapting it to the provided materials and budget was necessary, which was the reason to start designing it from scratch but maintaining the initial concept.

The final outcome of this part of the project was the construction of the physical **AGV** unit, arrangement of electronics and basic programming for functional testing purposes have been included.

(*According to European **SME** definition Commission, Internal Market, and SMEs, 2017*)

Acknowledgements

We would like to thank our supervisor Richard Senington for his advice and being the nexus to other professors who helped us with many parts of the project and our examiner Stefan Ericson for his understanding and patience with all the problems that we had during the project realization. We would also like to give special thanks to Zimon, without his help this project would have never happened. We have to mention Rikard Ed, who opened an special safety course date in order to be able to use the University workshop.

Finally, we have to thank the organizations which made possible this project, which are the University of Skövde, ASSAR Industrial Innovation Arena and Nystedts.

Roberto Carlos: To start off, I would like to thank my family, without their support I wouldn't be where I am today. I am also grateful for all of the new relationships that I have made this year, who have helped me to be a better person and I am sure they will be with me for a long time to come, they have helped to make this Erasmus the experience of a lifetime. This project has had its ups and downs, so finally I would like to thank Juan Carlos, you have taught me so much and always stayed positive in spite of adversities, while bringing "Zimon Muchacho" to life you have been a great partner and a better friend.

Juan Carlos: First of all, I have to thank more than my partner, my friend Roberto. He shared his knowledge with me along the process which helped me both to design and construct the "Zimon muchacho" **AGV** (our child).

This experience would not have been the same without the people that I have met during the Erasmus, specially "Xenia First Floor" group . They made my experience unique and unforgettable. My friends in Spain are not lagging behind, they have helped me in the distance with everything that they could and even made the effort to come and visit me despite of the weather and leaving their family behind in Christmas. I feel that I don't deserve them.

I could not thank my family enough for the huge support that they have been, not only during this year, but at every step on my journey, always having their hand on my shoulder and giving me the freedom of choice at all time. Thank you for the values that you have stilled in me since I was a child, which helped me to get where I am standing now. Thank you from the bottom of my heart.

No puedo agradecer lo suficiente a mi familia por el gran apoyo que han sido, no solamente este año, sino en cada uno de los pasos que he dado a lo largo de mi vida, siempre teniendo una mano tendida en mi hombro y dandome libertad para tomar mis propias decisiones. Gracias por los valores que me habéis inculcado desde niño, que me han ayudado a llegar donde estoy hoy. Gracias desde lo más profundo de mi corazón.

Contents

List of Figures

Acronyms

SBC **S**ingle-**B**oard **c**omputer. viii

SCADA **S**upervisory **C**ontrol **A**nd **D**ata Supervisory Control And Data Acquisitioncquisition. viii, 25

SLAM **S**imultaneous **L**ocation **A**nd **M**apping. viii, 16, 60, 63

SME **S**mall and **M**edium-sized **E**nterprises. ii, viii, 2, 4, 26, 60–62, 69

STL, SLA **ST**ereo**L**ithography. viii, 10

WLAN **W**ireless **L**ocal **A**rea **N**etwork. viii

μC **M**icro **C**ontroller. viii

List of Tables

Chapter 1

Introduction

1.1 Background

Logistics in the industrial environment is the crucial link between the different stages of a manufacturing process. Since the beginning of industry, material handling has always been performed by operators who are in charge of moving parts from one part of the production chain another using transportation devices such as forklifts, pallet trucks, or carts. Apart from being hazardous for the operators when dealing with dangerous or heavy materials, it is an operation which could be optimized. (Sabattini et al., 2018)

FIGURE 1.1: Manual material handling

It was not until The Third Industrial Revolution that the automation seed was planted with the development of control systems, which led to the necessity of changing the transportation methods in order to improve the overall performance of the whole process. One of the solutions given to carry out this task was the development of the concept of Automated Guided Vehicles (**AGV**) (Taalbi, 2017). These devices would avoid using human effort for logistics applications, as they are capable of moving parts with ease following a predefined path, being aware of their surroundings, and leaving the worker free to perform other tasks.

The evolving industry 4.0 demands complete integration of information and communication technology into the already automated processes (Winter, n.d.). Intelligent Logistics takes advantage of this by defining an improved version of **AGV**s: The **A**utonomous **M**obile **R**obot (**AMR**). Creating an **AGV** fleet could get very expensive, considering in the first place that the price per unit of these devices, ranging from 200.000 SEK to 1M SEK, and the necessary external and internal infrastructure that it requires (Sabattini et al., 2018), starting from path creation to device priority, communication, and the biggest problem, the low flexibility and adaptability, as traditional **AGV** systems have a predefined path which is very hard to repurpose for other applications.

AMRs extend the behaviour of **AGV**s but adding key features such as avoiding obstacles, gathering information about its environment, or even optimizing operation paths (Siegwart and Nourbakhsh, 2004). The ultimate result of this project is the development of an **AGV** with the final goal of reducing cost which would open the field to small to medium size companies. It would as well set the basis for implementing a fully operational **AMR** in following projects.

1.2 Purpose

Currently, commercial **AGV** production is led by companies that deliver high-end logistics solutions that comply with industry standards and regulations, but for an elevated cost which can put these tools out of the reach of SME.

Companies that cannot afford to automate some of their processes with these machines can suffer from decreased productivity and cause them to be less competitive. And those who can afford to purchase commercial **AGV**s must rely on the manufacturers for service and maintentance, and any desired modifications to their design for specific tasks could be impossible or very expensive.

Due to these elevated costs and the lack of flexibility, an **O**pen **S**ource **H**ardware alternative could be very interesting, as it could help to reduce the final price per unit, provide complete flexibility in the design, and furthermore the development costs would be shared among those who participate in the project. Nonetheless, there has been very little development in the field of Industrial **OSH AGV**s.

The main objective of this project is to investigate and develop the possibility of industrially capable **O**pen **S**ource solution that can compete with expensive commercial devices.

1.3 Problem statement

The purpose of this thesis is to continue developing upon the work started last year in the project "Open Hardware AGV" (Gámez and Martínez, 2021), where a theoretical basis and design was set to build an affordable **OSH AGV** which could be implemented in **S**mall and **M**edium-sized **E**nterprises. Much of the research carried out in the previous project is going to be implemented in this project which makes it possible to progress faster than if all the research has to be done from scratch.

Thus, the **research question** for this project is: **What are the challenges of developing an Open Source Hardware (OSH) AGV which could be used in industry by SME?**.

1.4 Objectives and limitations

To attempt to answer the research question and evaluate the aim of the project as a whole, the following objectives will be defined.

Objectives

- Manufacture a prototype **AGV**.
  - Build the structure as faithfully as possible to the original design (Gámez and Martínez, 2021).
  - Set up necessary electronics
  - Implement basic software for testing
- Evaluate performance of the prototype.
  - Measure load capacity in tugger configuration.
  - Measure load capacity in underride configuration.
  - Evaluate general usability.
- Evaluate viability of an affordable industrially capable **AGV** using **OSH**.
  - How far is it from being useful within an SME?
  - Does it make sense to pursue this idea further?

There is a series of factors that difficult the process of achieving these objectives.

Limitations

- Budget.
  - Initially set at 2000-3000 SEK, making it necessary to acquire parts and materials by means other than purchasing.
  - The actual cost of a fully functional unit would depend on the requirements of the company whose interest is upgrading its logistic system. A parametric design would help at the time of adapting the parts and dimensions to the company's needs.
- Time.
  - Building a physical artifact is very time-consuming.
  - There is only one semester to adapt design, build and document project.
- Resources.
  - The project is dependent on University staff being able to provide necessary tools for manufacturing on their own time, as students are not permitted to use metalworking tools without supervision. It should be noted that in a real industrial environment, this constraint would not be a problem as there will be full availability of machines and materials necessary.

1.5 Sustainability

This projects can be considered to have two different mechanisms in which it manifests sustainable development. The project itself, as an object, uses multiple repurposed parts, which otherwise may have gone to waste with time. Furthermore, as it is based on **O**pen **S**ource **H**ardware, the plans for the **AGV** will be made public. Therefore any user is able to consult the documentation and obtain necessary information to carry out maintenance or replace any parts that might be faulty by reusing old parts in new devices, supporting circular economy. Making servicing more accessible for users makes it less likely that a faulty unit may be discarded and replaced.

From a global perspective, this project promotes development in **OSH**, which will hopefully extend the use of **OSH** in other projects in the future. This would increase accessibility and flexibility overall for all technological development. If plans for more projects are publicly available, future work can build open what is already there, saving time and resources overall.

Another way this project may be beneficial is if it were actually implemented by an **SME**. By implementing **AGV**s in businesses, hard and monotonous labor is passed from being carried out by a human worker to being done by a machine. This allows workers to dedicate themselves to more elaborate and useful work. Furthermore, this automation increases throughput and reduces the utilization of resources.

1.6 Human, Society and Environment responsibilities

The realization of this project might not have brought huge technological advances to the humankind. What it is really remarkable, is the aim of making it accessible for every individual to adapt it to any application and sharing the knowledge to the world in order to help in the pursuit of improving the design to reach its peak performance, being the unique profit developing an industrial logistic solution that would be available for every company trying to adapt and optimize their processes.

The perpetual intention of the corporations of making the maximum capital gains usually leaves behind the mere intention of developing and researching in benefit of knowledge and technological advance. Keeping the resources unavailable to the **OS** community is not fair in a way that there is not reciprocity at the time of sharing thoughts and possible solutions to certain problems. The companies are always going to have the possibility of reaching open source solutions to get inspired and develop their closed source products, but never in the other way around.

Societal wise, the inclusion of this kind of devices in the industrial processes would suppose the eradication in some cases of precarious jobs that would not exploit the full potential of the worker and could even be hazardous. Logistics tasks usually involves moving high loads from one point to the other of the facilities, with the risk associated to performing this activity.

On the other hand, it could be seen as well in a way of eliminating jobs, because there could be the case of no need of relocating the worker in other part of the plant or the process because the machine is performing the task even more efficiently and with a very low necessary investment. In any case, there would be always a need of supervising the machines that replace workers in a plant.

Finally, the environmental responsibility is a matter that has been considered throughout all the development process of this device. The **O**pen **S**ource **H**ardware nature of the project and the aim of recycling, re-purposing and 3D printing parts helped to obtain a solution that apart from being affordable (prior standard validation), could be fully configurable and adaptable depending on the particularized requirements of any desired application.

1.7 Overview

In the following chapters, the design and manufacturing processes of the **AGV** will be described. In Chapter 2, the theoretical framework will introduce and define the different aspects that are going to be treated along the thesis. The methodology used will be detailed in Chapter 3. The literature review will be exposed in Chapter 4. Chapter 5 will describe the implementation process: from the 3D model to the real world unit. The results of the research will be in Chapter 6

Finally, conclusions of the whole process will be extracted in Chapter 7.

Chapter 2

Theoretical Framework

In this chapter, different aspects that are not covered in the previous **OSH AGV** project (Gámez and Martínez, 2021) will be introduced. Based on the previous design a range of techniques and technologies are needed to construct an **AGV** including:

- **Manufacturing methods**: **C**omputer **A**ided **D**esign and 3D printing will be introduced, taking part in both prototyping and final part possible solutions.

- **Electrical design**: On the first part, crucial parts of the powertrain of an electrically driven vehicle will be briefly described, going from the motor technology and the controllers that would be necessary using.

  The different sensors that would enable the unit to perform autonomous activities will be introduced as well in the next part, including LIDAR, Ultrasonic, Line or encoder sensing devices, apart from the safety actuators that are mandatory for the **AGV** unit to wander around industrial environments.

  Finally, an introduction on different industrial computing devices will be enumerated and described, as well as the programming environment that will be implemented and used.

Although this chapter presents the background on these capabilities along with the broader background on OSH/OSS, the information gathered in the literature review and theoretical framework of the preceding project will be used as the foundation of the project in the following chapters.

2.1 Open Source Initiative

The **"open source movement" (OS movement)** was born with the vision of keeping software advances openly available for everyone to understand (Deek and McHugh, 2008). This would give chance to improve every individual creation, which would be highly transparent throughout the whole development process. The **OS** initiative defines a whole new mindset in the technology research environment since there would not be any barriers in any part of the design or implementation flow in comparison to closed sources as, for example, big companies.

FIGURE 2.1: OS (Santos, n.d.).

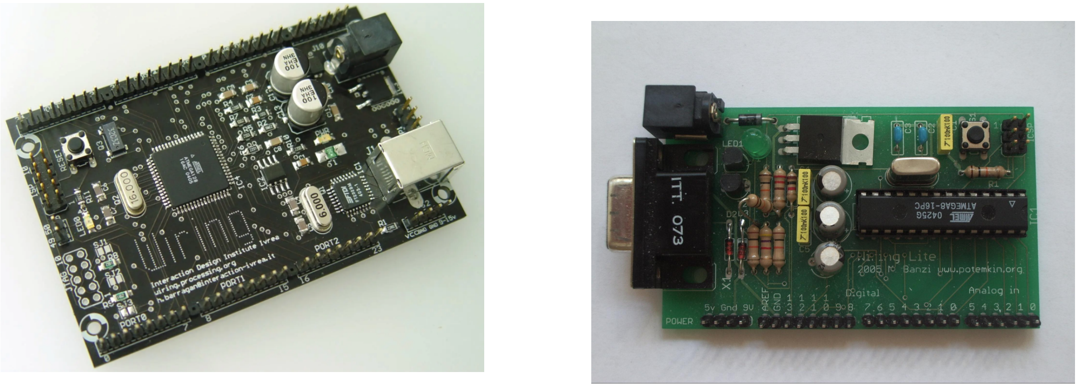
More particularly, this thesis would be more defined in the **O**pen **S**ource **H**ardware (**OSH**) movement. In this case, instead of developing a software solution where anyone could find and edit its source code, the hardware developer would find explicit drawings and plans of the whole project and adapt or reproduce it from scratch (Hannig and Teich, 2021).

Working in a **OSH** environment would be very beneficial in the development of a project in many ways, starting from the economical aspect; as there are many people working "on the same page" and there is constant transparency and communication on the creation process, there could be a very quick detection of any problem and further solution, something that could not be achieved if there were only an individual facing it, wasting more time which is always more expensive.

FIGURE 2.2: **OSH**(Santos, n.d.).

One of the most famous examples of the combination of Open Source Hardware and Software is **Arduino**, which initial idea started as a master's degree project in which **"Wiring"** was developed (Barragán, 2020).

The aim of this work was to make it easier for artists and designers, which are not familiar with electronics, to abstract from the difficult part of it so that they could focus on their own objectives. This idea was so successful that it evolved to what we know today; one of the biggest Open Source victories of the community, which has been widely spread from the point of its origin.



(A) Wiring (*Wiring Was Arduino Before Arduino | Hackaday* n.d.).

(B) Arduino Prototype (Barragán, 2020).

FIGURE 2.3

2.2 Manufacturing methods

This section introduces a brief history of the 3D modeling tool that has been used and the methodology to materialize a project in an accessible way: 3D printing.

2.2.1 Computer Aided Design (CAD)

The origins of **C**omputer **A**ided **D**esign (**CAD**) and 3D modeling could be traced back to the 1960s. At this time, this powerful design tool was very restricted to professionals engineers, basically because of the complexity associated with using the software and the previous knowledge that was compulsory to have. It was not until **Ivan Sutherland** introduced **Sketchpad**, a graphical interface that would be a complete game changer in the **CAD** and 3D modeling environment. (*When Did 3D Modeling Start? A Brief History* n.d.)

FIGURE 2.4: Sketchpad.Ivan Sutherland

Sketchpad was the spark that lit the fire of **CAD**, which later succeeded in the development of **C**omputer-**A**ided **T**hree dimensional **I**nteractive **A**pplication (**CATIA**) and AutoCAD in the 1980s. This would define a new era in the parametric 3D design which may be a very powerful tool but only if it is used by experts in the field, who would be able to correctly embed constraints and relationships within a model.

It was not until the 2000s when **SpaceClaim** integrated a solution for making 3D **CAD** even more intuitive and accessible for experts and curious users: **direct 3D modeling**. By adding this feature, designers could deliver a concept model without knowing anything about its modeling history, which was mandatory in previous parametric design (Tornincasa, Di Monaco, et al., 2010).

In the present day, these design tools combine parametric and direct modeling to conform to a hybrid tool. An example would be **Autodesk Fusion 360**, which in addition to this capability, it includes a design history where the user could edit a previous feature or part and see the changes that have been performed in the design, as it has a system of additive versions of the same file depending on the last time saved.

This leap in **CAD** would accelerate the creation process of developers in the **O**pen **S**ource **H**ardware environment, because these programs have become easy to use for everyone interested in the field.

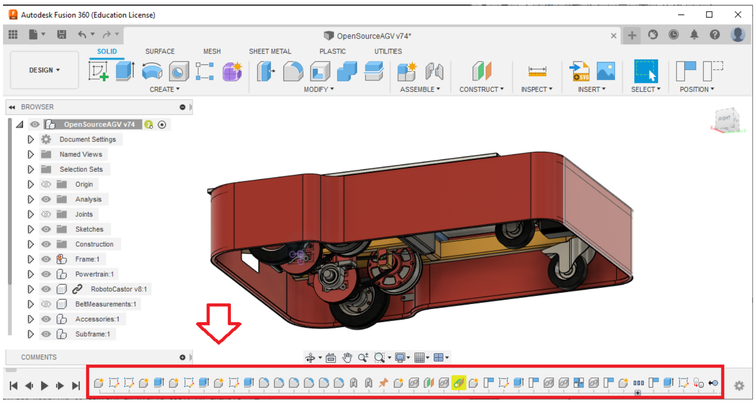


FIGURE 2.5: Autodesk Fusion 360 timeline

2.2.2 3D printing

Subtractive manufacturing was the main method used in industries to make solid 3D objects. This method consists in removing material from a solid block using lathes or milling machines until the desired shape is obtained. During the 1980s, other ways of industrial manufacturing were considered: **additive manufacturing**, which in contrast to subtractive, consisted on depositing material layer-by-layer. (Torta and Torta, 2019)

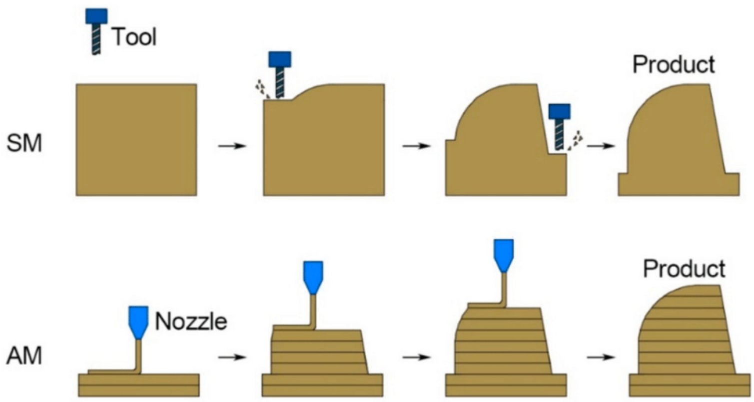
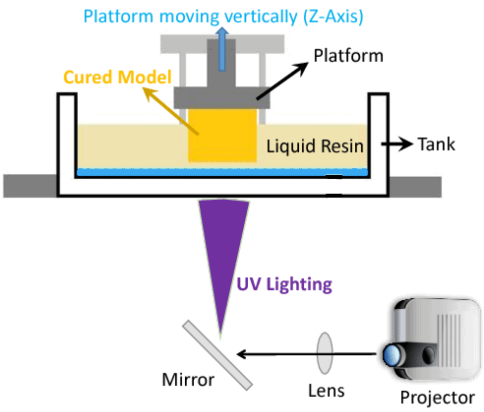


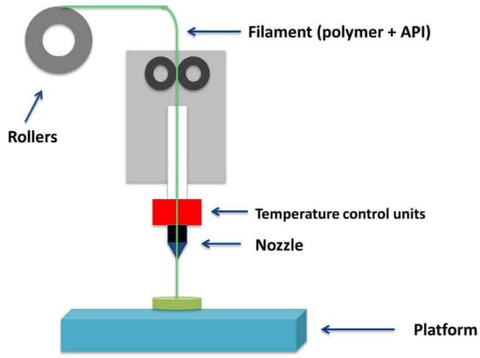
FIGURE 2.6: Subtractive and Aditive manufacturing (Tao, Yin, and Li, 2020)

Different technologies were developed during the same decade to achieve 3D additive manufacturing among which could be highlighted **ST**ereo**L**ithography (**STL, SLA**) and **F**used **D**eposition **M**odeling (**FDM**) (Savini and Savini, 2015).

- **STL, SLA** invented by **Charles Hull** consists on making liquid polymers harden using ultraviolet light. The final shape of the object would be reached by pulling the dried layers from the recipient where the liquid polymer reside.
- **FDM**, developed by **C.S. Crump**, on the other hand, consisted of depositing fused material in a layer-by-layer fashion until the final shape was achieved. There is a wide variety of materials that could be used to 3D print with this method, depending on the required properties of the final object. This fabrication technology is the most used to date for both prototyping and final products.



(A) STL/SLA 3D printing (*Stereolithography (SLA) 3D Printing Overview | Xometry Europe* n.d.).



(B) FDM 3D printing (*What Is FDM 3D Printing? [Quick and Simple Explanation]* n.d., FDM 3D printing).

FIGURE 2.7

These two technologies today have significant relevance due to their accessibility and ease of use, as there are plenty of tutorials and guides of how to set up and print whichever object the user needs.

FDM 3D printing was made affordable for anyone thanks to **Prusa Research** by **Josef Prusa** who shared his designs with the **OS** community and started a company due to the huge interest that this new technology raised. (*How Czech startup Prusa Research hit 17,122 | Sifted* n.d.). Even though his company has become one of the most important in this field, the blueprints and design of each and every part of the 3D printers are available for everyone to manufacture its printing tool by themselves.

OSH developers benefit from this in several ways, starting from prototyping; where they could study the performance of any design in the real world, to final product detailed printing; it is possible delivering a product printed by itself straight from the 3D model design.

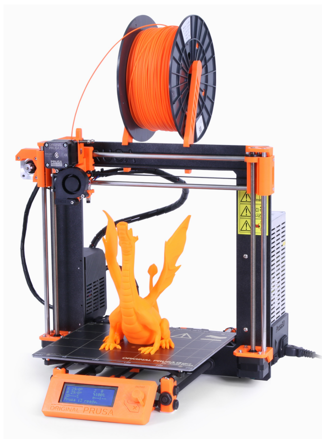


FIGURE 2.8: Prusa i3 3D printer (*Prusa i3 - Wikipedia* n.d.)

2.3 Standardization

Many issues must be considered at the step of moving from a conceptual design to a real-world solution. There is a set of requirements that should be embedded in the design before the product can be released to the market. This set of requirements are gathered in a **Standard**, which are documents released by state and global competent organizations (Guido, 2011). ISO is an example of a global standard, EN/DIN corresponds to European standards, and SS, which would be followed in this project, is the national Swedish standard. The national standard is the most limiting. National standards (e.g. SS) have to be in accordance with the continental (e.g. EN) and worldwide standards (e.g. ISO)

Different standards exist depending on the product family that is evaluated. Industrial standardization becomes the most restrictive due to the safety requirements that any device in the work environment must comply with in order not to cause material damage or personal injury, the main priority. The standard which frames this project is ISO 3691-4:2020 (ISO 3691-4:2020, n.d.), where the requirements that a driverless truck must meet are listed and categorized. In this standard could also be found referenced other standards for safety regulations, electrical design, signalling and all the related issues to this type of vehicles.

This standards are EN ISO 13849-1:2015 (ISO 13849-1:2016, n.d.), safety measure in performance levels and EN 1175:2020 (1175:2020, n.d.), which involves the required electrical safety measures. As this project is the continuation of the theoretical analysis of the possibility of building an Open Hardware **AGV** (Gámez and Martínez, 2021), the standardization part of the design would be hugely based on this previous work, emphasizing in annexes 5 to 7 where most of the requirements have been evaluated in detail for this type of vehicle.

2.4 Electrical Design

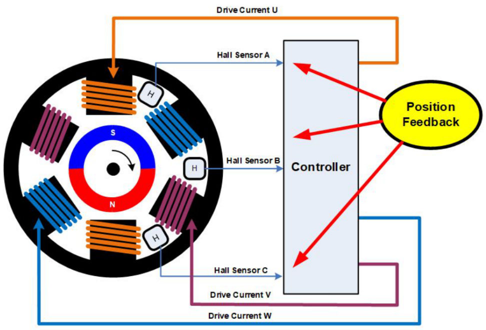
This section covers everything from the necessary parts to build an electric powertrain to the minimum sensors and emergency actuators that should be included in the design to be industrially capable.

2.4.1 Motor and motor controller

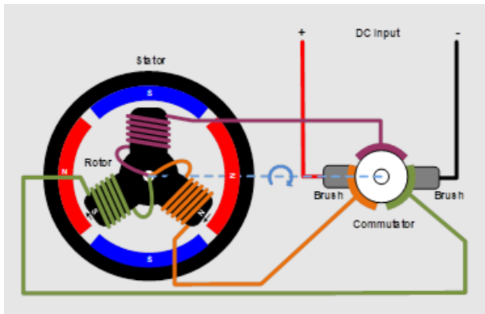
When it comes to the selection of the motor for this **AGV** application, it is necessary to differentiate the two main types of existing DC motors in the market. A brief explanation of each motor type will be given in order to further understand the advantages and disadvantages at the time of choosing one or the other:

- **B**rush**L**ess **D**irect **C**urrent (**BLDC**) motor implements electronic commutation. Attached to the rotor shaft there are magnetized permanent magnets and the coil windings are placed at the stator part. In between the windings, position sensors are installed in order to detect the polarity of the rotor, and the information of the sensor is sent to the controller, which switches the current direction in the coil winding to generate a magnetic force in the desired direction to drive the motor.

- **B**rushed **D**irect **C**urrent (**BDC**) motor coils-magnet configuration is inverted in comparison to **BLDC**. In this case, the coil windings are situated in the rotor and the permanent magnets in the stator, The commutation is produced mechanically using carbon brushes which slide in the commutator to produce a rotating magnetic field.



(A) BLDC motor model (*Brushless DC Motor vs. Brushed DC Motor* n.d.).



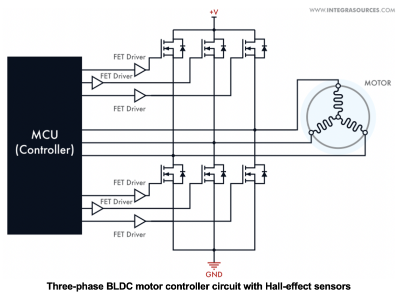
(B) BDC motor model (*Brushless Vs Brushed DC Motors: When and Why to Choose One Over the Other | Article | MPS* n.d.).

FIGURE 2.9: BLDC and BDC motor models

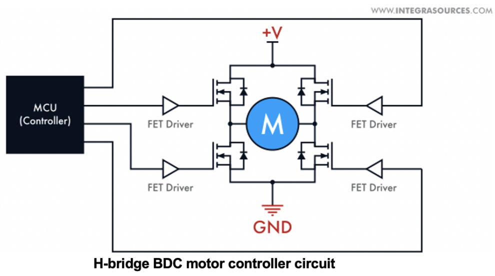
Activities performed by an **AGV** first require a high torque per unit of power and mass because of the high weight it has to transport. Both motor types could achieve this, but it has been demonstrated that **BLDC** perform this task more efficiently. The way that **BDC** motors are structured and the fact that the commutation is performed mechanically, makes them less reliable as there is a high possibility of breakage of the carbon brushes. Heat dissipation is another factor that has to be considered, as higher load involves higher current drain; **BLDC** motors fulfills this task more rapidly compared to **BDC**. (*AGV Brushless DC Motor Benefits* 2020)

Although **BLDC** motor might seem the best option for this application regarding efficiency, reliability and heat dissipation, it is still necessary to study the type of motor controller that would be needed to run the motors. In the case of **BDC**, H bridges with two high-side and two low-side switches (two on each side for forward-backward control) are used. In order to control the speed of the motor, the most extended solution is introducing a varying **PWM** signal to the gate of the drivers that "open" the transistors a period of time. This type allows **open loop** control, which does not require motor feedback in order to work.

On the other hand, **BLDC** motor controllers use half-H bridge circuits to control speed and direction. The number of phases of the motors determines the number of transistors used; for a three phase **BLDC**, the most common configuration for this application, requires three half-H bridges, one pair of high/low side switches per phase. In addition to this, position or hall effect sensors would be needed to run the motor because the controller requires rotor position feedback to fire the switches in a specific period to get the motor running depending on the varying **PWM** signal fed to the drivers. This mode of operation is defined as **closed loop** (Andrey Solovev, 2019).



(A) BLDC motor controller (Andrey Solovev, 2019).



(B) BDC motor controller (Andrey Solovev, 2019).

FIGURE 2.10: BLDC and BDC motor controllers

Again, thanks to the **OS** community there are affordable solutions that could be used to control the motors of this type of vehicles. An example of this is **Odrive robotics**, whose main objective was to provide robot motor controllers and even motor designs to the community (*ODrive* 2019).

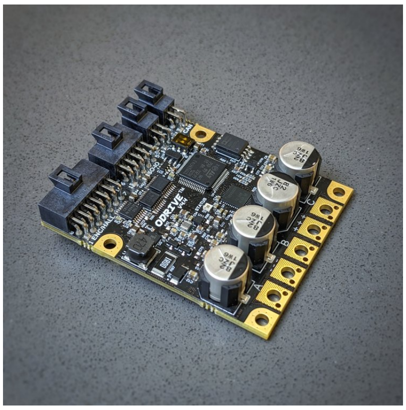


FIGURE 2.11: OS motor controller (*ODrive* 2019)

Motor controllers have always been mounted separately from the motor, but since it is mandatory to have these devices to run the motor, there are companies that have developed a compact and unified motor-motor controller solution that would fit perfectly in the **AGV** or similar vehicles. Maxon group delivers a solution which also includes a gear reduction for very low speed and high torque operations (Maxon House, 2017).

FIGURE 2.12: BLDC compact drive (*Compact drives by maxon maxon group* 2022)

2.4.2 Battery

This section is deeply discussed in the previous Gámez and Martínez, 2021's FYP, which was hugely inspired by the work developed by Ullrich, 2015. In the second appendix, the optimal **D**epth **o**f **D**ischarge (**DoD**) is calculated to find the most profitable battery for this application. The chemistry of the compared battery was lead acid, lithium iron **N**ickel **M**anganese **C**obalt Oxide (**NMC**) and **L**itium **I**ron **P**hosphate (**LFP**). From the analysis, it is extracted that **LFP** would be the best-fit solution for this application in the long run among **NMC**, which would be finally discarded, and Lead Acid.

Although the aim of the continuation of the project is to remain faithful to the theoretical work developed in the previous research project, budget constraints and availability would possibly define the battery type used in the actual project.

2.4.3 Sensors

The most important feature that any unmanned vehicle must have is **awareness of its surroundings**. This is achieved by implementing sensors in the design of the unit. Depending on the level of autonomy, there would be different types and numbers of sensors that would have to be included. For this application, either **AGV** or **AMR** share the same sensing devices, but in the case of the last autonomous vehicle type, cameras should be added for navigation and specific object detection purposes. In the following table, sensors have been classified according to the function that it performs in the **AGV**. A brief description of the most used sensors will be given afterwards.

TABLE 2.1: **AGV** sensor compilation depending on its function (AGVNetwork, 2022).

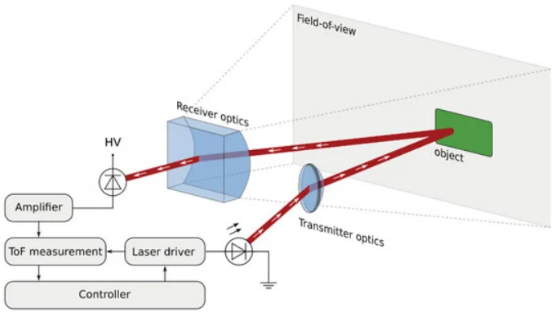
Function	Type of Sensor	Application
Safety Sensors	Safe 2D Lidar Bumper Encoder	Safe personnel detection Vehicle stoppage if contact Vehicle speed and steering detection
Environment Perception	2D & 3D Lidar, Ultrasonic, Camera, Radar	Avoid impacts with objects
Navigation and Localization	2D & 3D Lidar, Ultrasonic, Camera, Radar Line sensors (magentic, inductive, optic sensors)	Mapping, Localization and Navigation
Load Handling and Identification	Optical distance sensors or Wire draw encoders Cameras, 2D or 3D LiDAR, Ultrasonic Photocells, ultrasonic, inductive RFID, Laser or Image based bar code scanners	Fork Height Sensors Pallet pocket detection Ensure the right load positioning Transported material identification

LIDAR

As shown in the table, it is a sensor that could help to perform both the safety and navigation functionalities. Most autonomous vehicles have introduced this technology to create an environment map while the vehicle is moving or set a virtual safety perimeter around the unit to avoid possible material and personal hazards (Behroozpour et al., 2017).

The working principle of **LiDAR** systems is very simple; it is a remote sensing technology which uses rapid pulses of laser light to measure distances between the emitter and objects or targets. The scattered beam is then collected by photodetectors. The processing unit then calculates the time it takes for each pulse to scatter back. Repeating this in short intervals and using algorithms such as **S**imultaneous **L**ocation **A**nd **M**apping (**SLAM**) allows the generation of a map (*RPLidar A1M8 : MDROBOT* 2020). One of the most widely used and affordable **LiDAR** sensors for **AGV** applications is RPLIDAR.

With this sensor, it is possible to generate a 2D map which could be stored in the memory of the **AGV** main processing unit for learning purposes, path planning, and more. **AMR** are sometimes required to implement 3D **LiDAR** systems which could allow the unit to define a larger working perimeter and increase the safety around the robot (AGVNetwork, 2022).



(A) Lidar working principle (*Sensors, Special Issue : LiDAR for Autonomous Vehicles* 2021)

(B) Rplidar (*RPLidar A1M8 : MDROBOT* 2020)

FIGURE 2.13: **L**ight **D**etection and **R**anging (**LiDAR**)

Ultrasonic sensor

The working principle of these sensors is very similar to the **LiDAR** previously defined, but with a lower distance range: 2-4 cm (*Sparkfun Electronics HC-SR04* 2022). In this case, the emitter sends a pulse of very high frequency sound waves that collides with the obstacle or object and then come back reflected to the detector. Again, the time in between the emission of the sound wave and the reception would help to define the distance of the object (Budisusila et al., 2021).

The main use of this sensor in **AGV** is safety, as it allows for collision avoidance. As these are proximity sensors, they could also be used to determine very close distances for example when reaching a package or piece in a particular process. These sensors are usually distributed around the vehicle, 2 per side, to ensure maximum coverage of the closer surroundings.

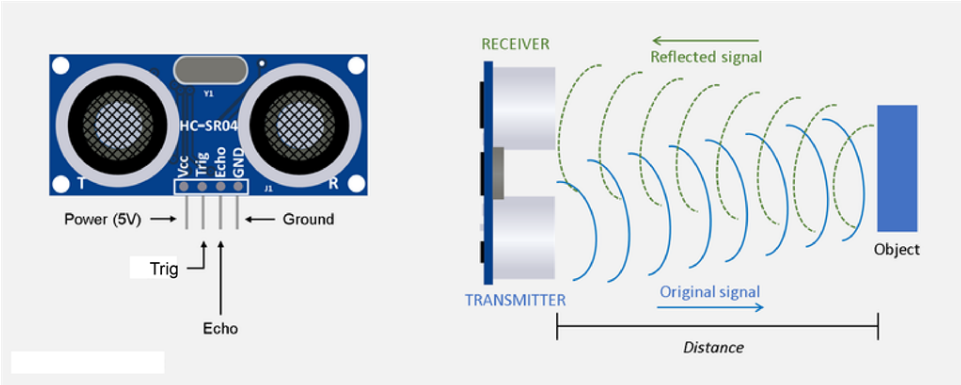


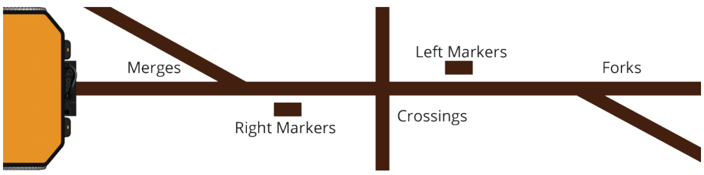
FIGURE 2.14: Ultrasonic sensor (Paramasivam, 2019)

Line sensors

The functionality of these sensors is purely navigational. They are frequently implemented in **AGV**s. These sensors in combination with a line path located in the working environment help the vehicle move in a restricted direction, without giving the opportunity to cause a hazardous situation to its surroundings. The problem with this method of guidance is the adaptability of the path to the needs of the current process.

This makes this method obsolete nowadays, as it is being substituted by **LiDAR** and ultrasonic sensors (Jiang, Xu, and Sun, 2021). The sensor used would vary depending on the material used for the path making. If metallic strips are used, the sensor mounted on the vehicle would be magnetic, if it is only paint, in that case the sensor would be optical. This method has been the most reliable from the beginning of industrial autonomous driven vehicles.

(A)



(B)

FIGURE 2.15: Line magnetic sensor (*Magnetic Guide Sensors* 2021)

Encoder

Navigation wise, encoders and **LiDAR** are the most important sensing devices that an autonomous driven vehicle must have in order to determine its exact position at all times with the less error possible. These devices are usually placed in the main shaft of the rotor situated in the motor and are already included in **BLDC** motors (hall effect sensors), because of the need of the motor controller to know the position of the magnets at all times. For some applications, these sensors would be more than sufficient to know the position of the vehicle, but some applications require the best precision possible (Cho et al., 2017).

There are two main types of encoders, optical and magnetic. The last type are the ones used in **BLDC** motors in order to sense the position of the magnet poles. In the picture below the working principle of magnetic and optical encoders is shown.

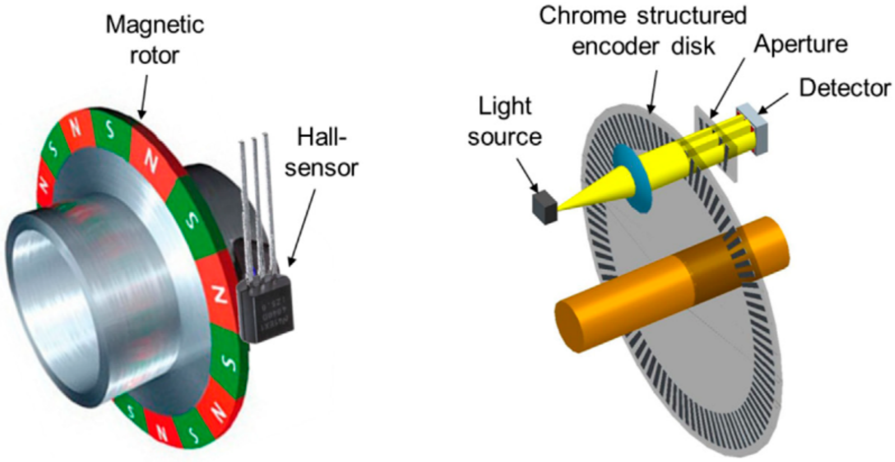


FIGURE 2.16: Magnetic and optical encoders (Seybold et al., 2019)

2.4.4 Emergency and warning actuators

These elements are mandatory in order to have the possibility to disengage all the systems of any autonomous vehicle in case of failure and alert the people around the unit about the state of the vehicle at all times. The type of actuator could go from warning actuators, emergency stop buttons, pin brakes or sirens. Information on emergency situations and safety concerns is gathered in ISO 3691-4:2020, n.d., and although this topic was explained in Gámez and Martínez, 2021's work, it will be developed in this section. A brief description of the minimum emergency actuators will be given.

Electromechanical Pin Brake

The working principle of this device is to stop the vehicle immediately in case of emergency by inserting a metal pin in the rim. This system is triggered if any emergency button is pressed or if the power supply is compromised by any failure.

FIGURE 2.17: Electromechanical Pin Brake (*Electromechanic Pin Brake for AGVs and Robots Kendrion* 2021)

In Gámez and Martínez, 2021's work, a magnetic emergency brake is described, the working principle is similar, but in this case it would stop the movement of the main shaft instead of inserting a pin into the wheel hub of the vehicle.

Emergency buttons

It should be easily accessible for the workers to stop the unit in case of a failure. In some cases, they are distributed in the front and back part of the vehicle.

FIGURE 2.18: Emergency button (*AGV Design and Component Considerations OEM Automatic Ltd* 2020)

Beacons and sirens

Light systems and sirens are very useful in industrial environments to warn people around the machines about the performed operation at a particular moment. If these elements are noticed at first, it is easier to be cautious when wandering around. Sometimes beacons and sirens are merged into one element, as is shown below.

FIGURE 2.19: Emergency beacon and siren (*AGV Design and Component Considerations OEM Automatic Ltd* 2020)

2.5 Single Board Computer (SBC)

In the history of industrial control and automation, the main and most reliable logical unit that has been used is the **P**rogrammable **L**ogic **C**ontroller (**PLC**), a robust system that enables the correct operation of a particular process always prioritizing the safety of the workers and then the expensive machines and materials involved. For many years, this has been the one and only solution for global industrial process control, but the development of industry and the incorporation of information and communication technologies have brought to the stage new requirements (Vieira et al., 2020).

In this section the incorporation of internet connected devices that could perform control operations will be introduced. **PLC**s are usually in charge of controlling any process in any industrial environment, but they are lacking "Internet of Things" capabilities which are required by Industry 4.0 standards. This task along others such as, path planning or map creation could be performed by newer and more powerful computers which could be introduced in this section.

2.5.1 Main processing unit

There is a remarkable fact about this section: the **OS** founding nature of the devices that will be described. It has been a very long journey before being able to use devices such as **Raspberry Pi** and **Jetson Nano** in industrial environments due to the difficulties encountered when trying to standardize them. Thanks to the new industry interconnection requirements, some companies have taken a step towards the inclusion of this devices in industrial processes by making them as reliable as the known **PLC** but adding uncountable powerful functionalities.

Thanks to being **OS**, there is more availability of similar devices that could be used for testing purposes in the development of any project before an industrial rated product is implemented in the design. This could significantly lower the overall cost of the project compared to using industrially capable products from the beginning of the project.

Industrial Raspberry Pi: Revolution Pi

Revolution Pi is an industrial PC, which is **OS** and modular. It is purely based on the Raspberry Pi 3 module, featuring a 1.2 GHz quad-core processor with 1GB RAM memory and up to 32 GB of eMMC flash memory (*3 Best Industrial Raspberry Pi Solutions in 2021* 2021).

FIGURE 2.20: Industrial Raspberry Pi (*3 Best Industrial Raspberry Pi Solutions in 2021* 2021)

Industrial Jetson Nano: Aetina AN110-XNX-EN70

AN110-XNX-EN70 is based purely on the Nvidia Jetson Nano module. It is an even more powerful machine, **AI** enabled which allows the implementation of algorithms such as autonomous path planning with object detection. Although NVIDIA company, which is the manufacturer, is not open-source based, they released this product along a github repository where troubleshooting and development would be open for everyone.

The performance of its quad-core processor is 0.5 TFLOPs with 4GB RAM memory and up to 16 GB of eMMC flash memory (*Aetina AN110-XNX-EN70 NVIDIA Jetson Nano.* 2020).

FIGURE 2.21: Industrial Jetson Nano (*Aetina AN110-XNX-EN70 NVIDIA Jetson Nano.* 2020)

2.5.2 Motor controller unit

The motor controller in combination with an Arduino-based PLC would be more than sufficient to drive the vehicle. As it was mentioned before, the fact that these components are available in similar shapes and configurations allows the community to test the developed systems without the need of using industrial capable electronic equipment from early stages of the projects.

FIGURE 2.22: Industrial Arduino (*Automation, Monitoring and Control with Industrial PLC Arduino* 2022)

2.6 Programming

Although this is not the main focus of the project, some basic programs have been implemented for testing purposes and the software described here would be relevant for future work applications.

2.6.1 Robot Operating System (ROS)

Every robot is made up of many parts including sensors, motors, vision systems, batteries, or emergency actuators. In order to perform a task, every part has to be controlled effectively and communicate with the main control unit, which will be in charge of running all the system simultaneously by receiving and delivering the information back and forth with the devices of the machine.

This services have to be provided by an operating system, similar to the ones used in regular computers; that is the role of **ROS**.

FIGURE 2.23: **R**obot **O**perating **S**ystem (**ROS**) (*ROS Wiki* 2022)

The goal of this **O**pen **S**ource operating system is to support code reuse in robotics research and development instead of "reinventing the wheel" each time a problem needs to be solved. **ROS** is a distributed framework of processes (Nodes) that enables executables to be individually designed and coupled at runtime. The processes could be grouped into Packages which are easily shared and distributed in the existing repositories (Koubâa, 2016). The system is running in Unix-based platforms such as linux or ubuntu.

It uses a publisher-subscriber pattern to receive and deliver information to the parts of the robot, which is similar to the **MQTT** protocol. Some remarkable features in **ROS** are: **language independence**; it is possible to implement any modern language as **Python** or **C++**, **low performance impact**; it is designed to be as thin as possible so it could be integrated with other robot frameworks. **ROS** environment allows the user to test robot programs in the virtual world thank to **Gazebo** and **Rviz** integrations before moving to the real world, helping in the debugging step of the process and preventing any possible hazardous operations.

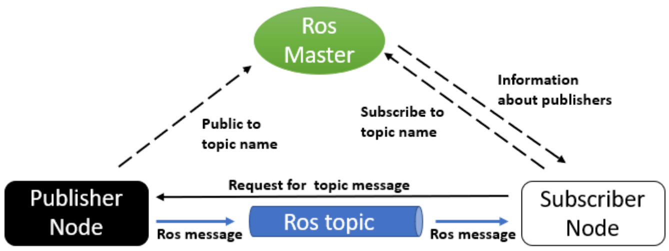


FIGURE 2.24: **ROS** communcation (*ROS Wiki* 2022)

Chapter 3

Literature review

The aim of this chapter is reviewing articles of interest which could help in the task of providing a basis for the realization of this project. The utilization of **O**pen **S**ource **H**ardware (**OSH**) and **D**esign **S**cience **R**esearch (**DSR**) in industrial applications will be highlighted, as well as the incorporation of 3D printing technology (additive manufacturing) in industry.

3.1 Open Source initiative: Article review

OS benefits and limitations

The **O**pen **S**ource initiative is, unarguably, one of the biggest knowledge sharing movements of all time. The contribution of the **OS** community at the time of solving problems is the most valuable aspect, as there is a considerable reduction of the time spent finalizing a particular task, compared to completing the same activity individually or within an small work group. As it was stated in the article: **Open Source and Accessibility: Advantages and Limitations (Heron, Hanson, and Ricketts, 2013)**, there are major advantages at the time of developing an open project that should be noted, for instance, in the **OS** Software environment.

Usually, the authors who release their software as open source have the expectation of increasing the market share of the project being developed along the maintenance of itself by the community working on it. At the same time, there is a potential increase of the employability of the main developer who achieves a remarkable personal reputation.

Corporations or individuals who want to develop and commercialise their software products also utilize **O**pen **S**ource to speed up the progress or seeking particular creative outlets. The final result is obtaining successful achievements and instant revenue in their own self-interest but both the community and the corporations benefit from the development process.

The previously enumerated advantages unfortunately come with limitations of the usefulness of open source applications in some environments. Poor quality end-user documentation is not always a concern of the open source community, as it does not give any major value to the developed software per se, which leads to leaving the project always in a draft state.

Another big drawback could be the minimum regard for the impact that could cause any big update in a particular software version. Sometimes the users could find it very difficult to adapt to the newest version of a program or environment just because the existing documentation is not suited for inexperienced community members.

Although there are limiting factors to this **OS** initiative, it is still an indispensable research tool which could help to solve major global development problems and challenges by merging a large amount of people working on the same task in order to give the most optimal solution and in the quickest way possible.

The same advantages, limitations and indisputable benefits that **OS** have, could be associated to other initiative which is the **O**pen **S**ource **H**ardware (**OSH**) movement (Rakitin and Markova, 2022). In this case, the final outcome of the developed project would be the possibility of sharing blueprints, models, knowledge and experience associated to designing and/or materializing a particular invention.

There is a relatively recent and very important project which merge both the **OS** Software and **OSH** initiatives. During the COVID19 pandemic, there was a huge need of releasing a reliable, cheap and fast-manufacturing ventilating devices to supply every hospital having patients which precised assisted respiration, as there was an extreme global necessity and shortage of these medical resources (*MPS Open-Source Ventilator* 2021).

Thanks to the existing **OS** community and the people who joined this global development race to deliver this device as soon as possible, a huge part of the infected population could recover. This revealed the importance of sharing knowledge and combining forces to fight any possible adversity, and as well demonstrated how powerful the **O**pen **S**ource Software and **O**pen **S**ource **H**ardware initiative were.

3.2 Open Source Hardware applications

Industrial robot control

Collaborative robots have been recently introduced in industry in order to optimize the processes and ease the job of the operators. Non-collaborative robots needed to be equipped with sensors and systems to ensure the safety at the time of working or wandering around the units. The implementation of control systems associated to fast real-time response to cope with the human unpredictable interaction was deemed necessary. **OS** and **OSH** were involved in the origination of the control system described in the article **An open embedded hardware and software architecture applied to industrial robot control (Zhang, Slaets, and Bruyninckx, 2012)**.

Commercial industrial robot controllers were based on modular hardware architectures with integrated software that could ensure robot reliability and maintainability, but did not take into account possible system extendibility for example with the addition of other sensing devices. The closed source nature of the software industrial solutions did not help as well in the attempt of implementing more sophisticated motion control algorithms.

The proposed solution to this problem was the creation of an open embedded hardware and software controller architecture which would be applied to an industrial robot platform, using flexible hardware such as field programmable gate array (FPGA) running real-time Linux and Open RObot COntrol Software (OROCOS). The outcome of this successful approach was the demonstration of flexibility and reliability of **OSH** FPGAs implementation in comparison to regular PC-based controllers. One big advantage of the system being **OS** is the possibility of controlling **any general multi-axis robot**, as long as the motors were equipped with incremental position decoders and driven by PWM.

Industrial Programmable Logic Controller (PLC)

Cyberattacks have become an actual hazard in every internet connected device. The inclusion of information and communication technologies in the industrial environment has brought to light the high grade of vulnerability of **PLC** and **S**upervisory **C**ontrol **A**nd **D**ata Supervisory Control And Data Acquisitioncquisition (**SCADA**) systems which are not only used in production lines; they are responsible of running electric and nuclear energy systems, transportation, or even hospitals.

Since the vendors of these industrial control systems do not make available hardware and software related information of the devices, the clients have to rely on the security updates which are not usually up to date with the type of hazards and cyber-attacks that the systems may suffer. In the article: **OpenPLC: An IEC 61,131–3 compliant open source industrial controller for cyber security research (Alves and Morris, 2018)**, an **OSH** and **O**pen **S**oftware alternative is given trying to face the vulnerabilities that the commercial available systems may have.

The **OSH PLC** used in the study was the "OpenPLC Neo", which was indeed developed by the authors of the article in a previous project. The "OpenPLC Project" was created in accordance with the IEC 61131-3 standard in the attempt of being able to compete with the commercial available solutions. OpenPLC Neo runs Armbian Linux, a modified Debian distribution which targets embedded devices. OpenPLC software runs on top of this operating system. The development environment is PLCOpen Editor, a graphical interface intended to ease the creation process of a PLC project. All of the features, programs and characteristics are build in the **OS** environment, available for the community to adapt the PLC solution to a particular application.

The outcome of the project was the demonstration of similar overall system performance of the OpenPLC system compared to other commercial PLC solutions, but with the advantage of being able to modify the code in order to defeat a specific attack and avoid vulnerability issues. This is a missing feature in commercial solutions that enhances the open source alternatives at the time of adapting the systems when trying to face cybersecurity hazards.

Autonomous Mobile Robot (AMR)

The use of **OSH** in some industrial companies as for example **RoboteQ (a NIDEC brand) roboamr** is becoming usual. This industrial component supplier had the idea of showing their product catalogue by creating an **OSH AMR** available for their possible clients to develop their own autonomous logistic solution.

The project was done with the aim of providing the possibility of having this kind of technologies in any company. As the design is completely open sourced, the costs related to this part of the realization of the project would be none, allowing for example **S**mall and **M**edium-sized **E**nterprises to use this devices to optimize their processes.

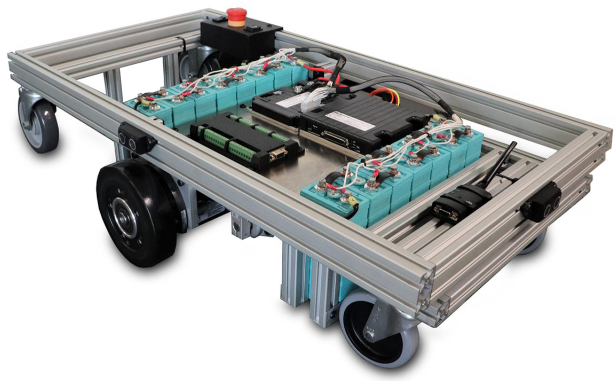


FIGURE 3.1: **AMR** RoboteQ (RoboteQ, 2021)

Each client could adapt it to any desired application by modifying its size and components, having in mind that the resulting operating unit would not adhere to any functional safety standards; in case that the company would like to verify the device, it would be necessary to run the tests and validation according to the rules or restrictions associated to the application that it would be used for.

The resulting device would be fully configurable and adaptable by the client, who would have been the one in charge of manufacturing the **AMR** from scratch, using just the required parts to run the device for a particular use or even increase the safety measures implemented compared to commercially available solutions. Although the sold components may not be fully open source yet, this is a huge advance in **O**pen **S**ource **H**ardware manufacturing and allows the community to improve and re design the unit as desired.

The major drawback of the **AMR** project reproduction would be the elevated costs of the industrial rated parts suggested by RoboteQ, which could get the resulting device being quite expensive. Changing the parts used in the design for others which could be **OSH** would lower the final bill of materials and thus the overall price of the construction of the unit.

3.3 3D printing industrial applications

Aditive Manufacturing (AM): 3D printing

In the article: **Additive manufacturing: expanding 3D printing horizon in industry 4.0 (Prashar, Vasudev, and Bhuddhi, 2022)**, the definition, history and applications of **AM** in industry is reviewed, specially focusing in the benefits of 3D printing and manufacturing in industrial processes.

3D printing has been associated to reduced cost and fast development both in the open source community and companies, but the wide variety of applications and materials that could be used to manufacture any type of component has drawn the attention even of aerospace or health care fields to the point that resulting 3D printed pieces are being used in finished products. **A**ditive **M**anufacturing, specially 3D printing, is going to be consolidated as a leading technology in this fourth industrial revolution, due to efficiency, reduced development costs, shorter lead times and decreased material waste among other factors.

In some cases, a particular design requires special pieces that are very difficult and expensive to manufacture because of the material used or the laborious process to produce it. Although 3D printing has been associated to plastic material component manufacturing, thanks to the evolving robotic technology and material science, it is possible printing even with metal wire. That is one of the cases of study presented in the article.

There are businesses that are specialized in Robotic Wire Arc Additive Manufacturing (WAAM) technology, which consists on metal deposition using a robotic arm with an arc welder. With this manufacturing method, a computerized 3D model could be turn into a real metal piece, as if was plastic 3D printing, widening the possibilities of developing any part that would not be easy to be manufactured with usual methods.

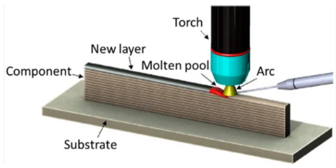


FIGURE 3.2: Robotic Wire Arc Additive Manufacturing (WAAM) (Rodrigues et al., 2019)

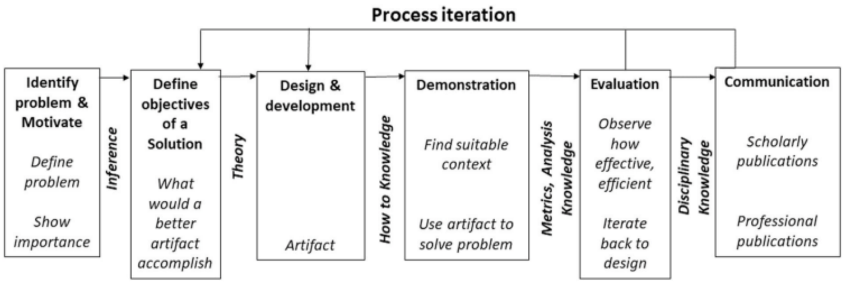
3.4 Design Science Research methodology

DSR pharmaceutical industry

In the article: **The role of Design Science Research Methodology in developing pharmacy eHealth services (Gregório et al., 2021)**, **DSR** methodology is presented to develop and implement innovative solutions to the health systems that yield maximum value for care.

The integration of technology in health care is one of the most promising upcoming solutions of the next decades. The arrival of "eHealth" concept to the pharmaceutical environment has created an actual need of updating the health services to adapt to the coming technological era.

The article proposes a series of case studies are used to explain how **D**esign **S**cience **R**esearch **M**ethodology would be applied to improve some parts of the health care system. An explanation of the method is resumed in an illustrative scheme, with the steps that would be necessary to follow and the tasks that would be done on each step. Iterations are contemplated after each run of this process and even during the realization of the research.



– DSRM cycle (adapted from (Peffers et al., 2007)[48]).

Short description of each DSRM activity

Activity 1 – Problem identification and motivation: Define the specific research problem and justify the value of a solution.

Activity 2 – Define the objectives for a solution: Infer the objectives of a solution from the problem definition and knowledge of what is possible and feasible.

Activity 3 – Design and development: Create the artefact. This activity includes determining the artefact's desired functionality, its architecture and then creating the actual artefact.

Activity 4 – Demonstration: Demonstrate the use of the artefact to solve the problem.

Activity 5 – Evaluation: Observe and measure how well the artefact supports a solution to the problem.

Activity 6 – Communication.

FIGURE 3.3: **DSR** methodology scheme (Gregório et al., 2021)

Chapter 4

Methodology

4.1 Introduction

Before carrying out research, it is first vital to define the method which is going to be followed. The chosen methodology defines where information comes from, how it is going to be processed and what is done with the results of the study, it also provides credibility to an article because the reader can know how and why decisions have been made. Without specifying a research method it is possible for researchers to waste time and resources by focusing on tasks that may not contribute to answering the research question that is trying to be answered.
Due to the nature of this project, where an artifact will be built to attempt to provide the knowledge necessary to answer a research question, it is thought that the most fitting method is **DSR**, or the **D**esign **S**cience **R**esearch methodology.

4.2 Design Science Research

The **DSR** paradigm has its roots in engineering and the sciences of the artificial (Simon, 1996). It is fundamentally a problem-solving paradigm. **DSR** seeks to enhance human knowledge with the creation of innovative artifacts and the generation of **D**esign **K**nowledge (**DK**) via innovative solutions to real-world problems (Hevner et al., 2004). As such, this research paradigm has generated a surge of interest in the past twenty years, specifically due to its potential to contribute to fostering the innovation capabilities of organizations as well as contributing to the much needed sustainability transformation of society (Brocke et al., 2012).
The **DSR** methodology consists of a multiple step iterative process which leads the researcher from the identification of the problem to the evaluation of the proposed solutions and communication of results.

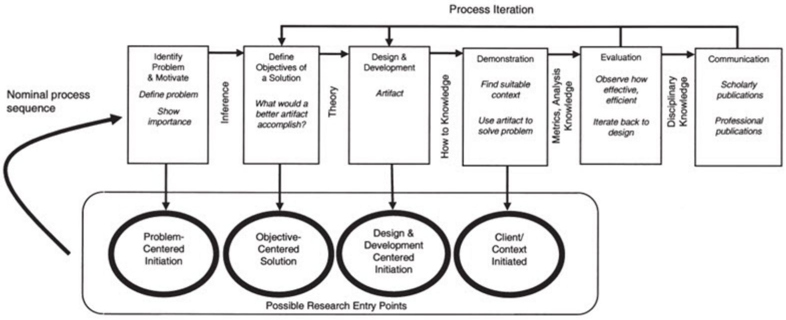


FIGURE 4.1: **DSR** methodology schema. (Peffers et al., 2007)

1. Problem identification and motivation

This activity defines the specific research problem and justifies the value of a solution. Justifying the value of a solution accomplishes two things: it motivates the researcher and the audience of the research to pursue the solution and it helps the audience to appreciate the researcher's understanding of the problem. Resources required for this activity include knowledge of the state of the problem and the importance of its solution. To obtain this knowledge it is necessary to carry out research on the topic at hand to determine whether the problem has been faced previously, if it has been overcome in the past it may not me necessary to carry out the project. If another researcher has unsuccessfully tried to solve the problem it can be insightful to take into consideration their experience and what challenges they faced, in this way it is possible to save time and resources.

2. Defining the objectives for a solution

The objectives of a solution can be inferred from the problem definition and knowledge of what is possible and feasible. The objectives can be quantitative, e.g., terms in which a desirable solution would be better than current ones, or qualitative, e.g., a description of how a new artifact is expected to support solutions to problems not hitherto addressed. The objectives should be inferred rationally from the problem specification.

3. Design and development

An artifact is created. Conceptually, a DSR artifact can be any designed object in which a research contribution is embedded in the design. This activity includes determining the artifact's desired functionality and its architecture and then creating the actual artifact.

4. Demonstration

This activity demonstrates the use of the artifact to solve one or more instances of the problem. This could involve its use in experimentation, simulation, case study, proof, or other appropriate activity.

5. Evaluation.

The evaluation measures how well the artifact supports a solution to the problem. This activity involves comparing the objectives of a solution to actual observed results from use of the artifact in context. Depending on the nature of the problem venue and the artifact, evaluation could take many forms. At the end of this activity the researchers can decide whether to iterate back to step three to try to improve the effectiveness of the artifact or to continue on to communication and leave further improvement to subsequent projects.

6. Communication

Here all aspects of the problem and the designed artifact are communicated to the relevant stakeholders. Appropriate forms of communication are employed depending upon the research goals and the audience, such as practicing professionals.

4.2.1 DSR in terms of cycles

As seen in "A Three Cycle View of Design Science Research" (Hevner, 2007), it can be considered that the iterative process is performed in three inherent research cycles: Relevance, Design and Rigor. These will be described in the next subsections, and to properly understand the correlations in between each cycle, a scheme will be provided below.

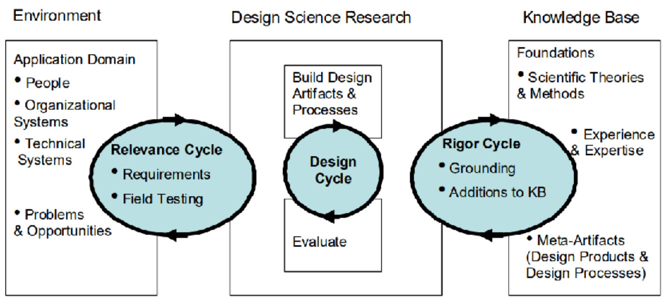


FIGURE 4.2: **D**esign **S**cience **R**esearch (**DSR**) (Hevner, 2007)

Relevance Cycle

This first step of the project is related to the motivation and context of the realization of a possible project to improve the environment by developing a tangible artifact or anything else that could be profitable. The relevance cycle would set the basis of the project, gathering the context in order to study its feasibility, the materials needed, people, knowledge and all the parts that would set the foundation of the design.An iteration process would be necessary evaluating both the requirements and field testing of the artifact or improvement that is studied.

Rigor Cycle

This cycle provides past knowledge to the research project to ensure innovation. The researchers will have to ensure that the development that is being carried out is referenced appropriately enough to base the new knowledge that would be generated after the **DSR**. This rigor is predicated on the researcher's skilled selection and application of the correct methods and theories to construct and evaluate the artifact or improvement.

Design Cycle

The heart of any **DSR**. This cycle iterates more rapidly between the construction of the artifact, evaluation and feedback to refine the final design. In this part, there is a continuous evaluation contrasted with alternatives and the requirements of the artifact until a satisfactory design is achieved. In this cycle, it is important to maintain a balance between the effort constructing and evaluating the artifact in order to waste the least time possible debugging the design. Artifacts must be thoroughly tested in the laboratory before being released for field testing. An early step in the process instead of performing iterations in this stage would imply the failure of the project.

4.3 DSR applied to this project

In this section there will be discussion about how **DSR** methodology is applied in this project, step by step.

1. Problem identification and motivation

This step is reflected in chapter 1 of this document, specifically in Background, Problem statement and Purpose sections.In these sections the problem is presented, the value of a solution is explained and the research question is defined. Subsequently the problem is deeply understood by carrying out a review of relevant literature and similar projects.

2. Defining the objectives for a solution

Also reflected in chapter 1, in the Objectives and limitations section. In this section the main goals for the project are presented, as well as the main obstacles that may prevent all the goals from being fulfilled.

3. Design and development

Design and development represents almost all of the work carried out in this project. Depending on the perspective from which you observe the project, the concepts of problem that needs to be solved, and artifact, may change. From a global point of view, the problem is the lack of **OSH AGV** development, and a possible artifact to solve this could be a prototype **OSH AGV**. However, at a smaller scale, a problem that needs to be evaluated could be that the motors need to somehow be attached to the frame, and a possible artifact could be a 3D printed part that interfaces with both the motor and the frame. As seen, the design and development step is applied at many different levels of the project.

4. Demonstration

In this project, demonstration occurs at two main phases. On one hand, due to how true to real life the CAD model can be, some parts can be considered to be demonstrated by simply placing them in the correct position. On the other hand sometimes the demonstration can only be carried out in real life, so it is necessary to acquire or fabricate the part to test it.

5. Evaluation

Just as mentioned in Step 3, evaluation occurs frequently and at different parts and scales. Every component must be evaluated individually, in other words, how well it performs its specific function; and as a whole within the project, which means how it benefits the project globally, for example, it may be interesting for a part to fulfil more than one function. It is important in this step to keep in mind the research question, the evaluation should be focused on trying to develop a viable solution for the project problem, and creating knowledge in this field. Practically every part used in this project has been evaluated, and then taken back to the Design and development step multiple times, and then re-evaluated, until it was deemed that it fulfilled its function adequately.

6. Communication

In this project the communication step happens when the findings of this research that are in this report are published. The work that has been done and the conclusions that have been drawn will be available on the Internet, for anyone who wishes to learn about development in the field of **OSH AGV**s, or perhaps another researcher, to read and apply this knowledge to their work.

Chapter 5

Implementation

5.1 Introduction

As mentioned in Chapter 1, a prototype **AGV** is going to be manufactured to try to answer the research question: **What are the challenges of developing an Open Source Hardware (OSH) AGV which could be used in industry by SME?**. As also exposed in Chapter 1, a major limitation in this project is budget. At the time of planning the construction, the budget was said to be in the region of 2000-3000 SEK. If we compare this budget to that of a similar project (Dzezhyts, 2020), where the budget was 50000 SEK, of which 20595 was spent, it is apparent that it wouldn't be possible to purchase all new parts and materials to strictly follow the initial design (Gámez and Martínez, 2021). Due to this, it was necessary to adapt components that were extracted from other projects, such as roboto (Dzezhyts, 2020), as to follow the initial design as closely as possible.

5.2 Design

5.2.1 Previous design evaluation

As mentioned in the introduction, due to budget limitations, it was not possible to purchase the materials and parts recommended in (Gámez and Martínez, 2021). To make the build possible it was necessary to make modifications to the design and adapt parts, keeping the key points of the structure as unchanged as possible.

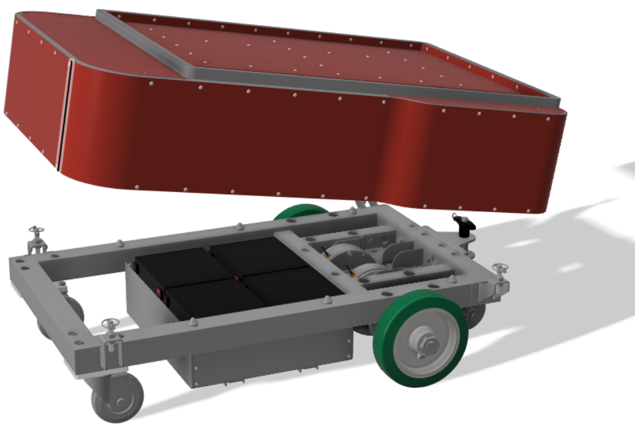


FIGURE 5.1: CAD design from OPEN HARDWARE AGV.(Gámez and Martínez, 2021)

Initially the idea was to take the design and only modify the parts that were going to be different, but this proved to be difficult because the dimensions of the materials that were available for use were initially unclear and it would be cumbersome to design the necessary parts and modify them again at a later date. To solve this, it was decided that a new design would be created from scratch in CAD, copying the fundamental design features from the previous design to preserve the structural calculations that were made, but using parametric design where possible, so that any changes in the materials that were available could be dealt with by modifying the dimension parameters, and the model would update automatically.

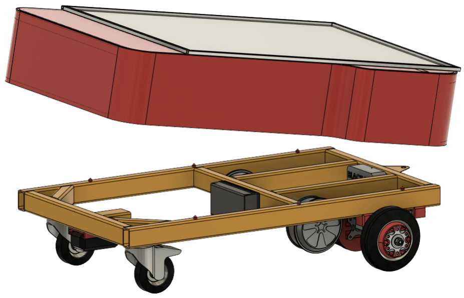


FIGURE 5.2: The current design.

5.2.2 Parametric CAD model

As mentioned in the previous section, it was decided that the best option given the circumstances was to create a new CAD model, maintaining key features but replacing the parts that cannot be purchased, with the new, adapted parts. The model was created using the Autodesk Fusion 360 software, which supports the use of parameters in models, in other words, instead of defining geometry in terms of dimensions, they are defined by a placeholder parameter, which in practical terms acts as a variable whose value can be modified at any point in time, which in turn modifies the parts associated to the parameter.

Parameters

Parameter	Name	Unit	Expression	Value
Favorites				
User Parameters				
User Parameter	tube_width	mm	30 mm	30.00
User Parameter	tube_height	mm	50 mm	50.00
User Parameter	tube_thickness	mm	3 mm	3.00
User Parameter	frame_length	mm	900 mm	900.00
User Parameter	frame_width	mm	515 mm	515.00
User Parameter	frame_middletube_...	mm	380 mm	380.00
User Parameter	frame_shortertube_...	mm	150 mm	150.00
User Parameter	shaft_dia	mm	20 mm	20.00
User Parameter	shaft_len	mm	250 mm	250.00
User Parameter	thread_depth	mm	10 mm	10.00
User Parameter	tube_fillet_r	mm	5 mm	5.00
User Parameter	Notchdepth	mm	4 mm	4.00
User Parameter	washer_dia	mm	35 mm	35.00
User Parameter	tow_hook_width	mm	150 mm	150.00
User Parameter	tow_hook_dia	mm	8 mm	8.00
User Parameter	subframe_pin_sepa...	mm	300 mm	300.00
User Parameter	subframe_pin_dia	mm	8 mm	8.00

FIGURE 5.3: Some of the parameters used in the design.

Due to the parameterized nature of the model, it is fast and simple to make modifications to the dimensions of the **AGV**. If a customer were to have access to a more polished version of this design, they could simply adjust the measurements of the model to their needs and would then have the means to manufacture it or seek an external company to build it.

5.2.3 Frame

The proposed design for the chassis of the AGV is a rectangular frame built from square steel tubes, with some reinforcement tubes in the center that fulfill various objectives, such as acting as a base for storing the bearings for the main axles, and being a solid attachment point for multiple components, such as the motors and the brakes. This frame design features some conical steel pins which are used to align the frame with the subframe, which will be described in the following section.

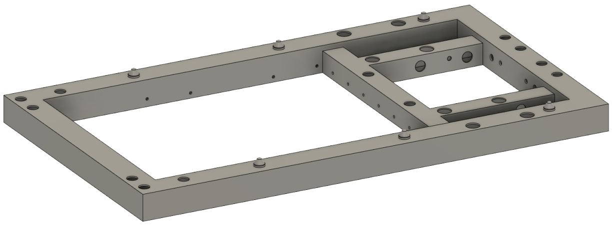


FIGURE 5.4: Original frame design.(Gámez and Martínez, 2021)

Due to the fact that the materials that would be available to build the frame were unknown, the design was made parametric to accept any dimensions of tube when this information was available. The frame features the same structure and layout as the original design, with some slight modifications to the dimensions of the reinforcement tubes to accommodate the motors that are being used. The updated design includes a small section of tube underneath the frame which is used to anchor the motors. The conical pin alignment system was also included, with the alteration that they are 3D printed, instead of metal. Two diagonal tubes are included to offset the caster wheels from the edges so to increase the clearance between them and the outer sheet metal skirt. A further tube is added in the front to support the battery.

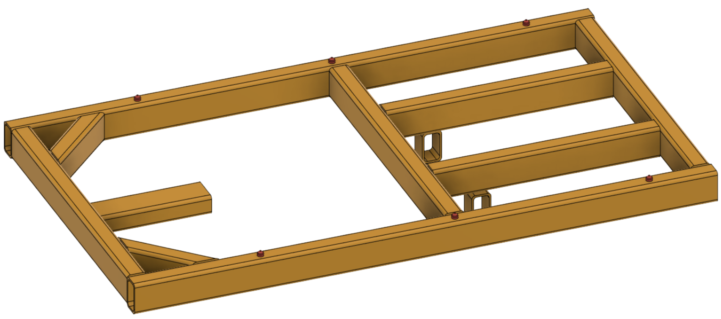
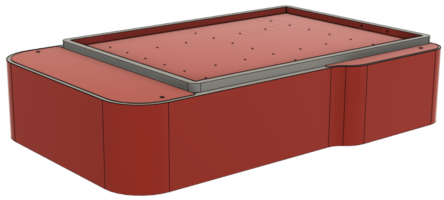


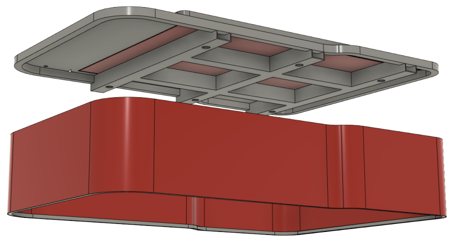
FIGURE 5.5: Current frame design.

5.2.4 Subframe

The design includes a component called the subframe, it consists of a smaller section welded steel tube frame which is placed on top of the main frame to evenly spread the load over the frame, and a sheet metal skirt which surrounds the edge of the AGV and encloses all of the moving parts and electronics, creating a physical barrrier for safety purposes.



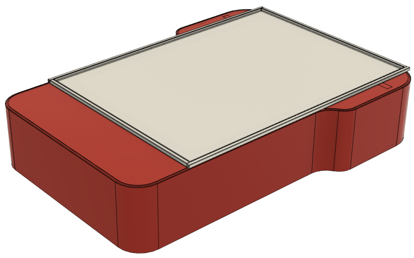
(A) Top view of subframe.



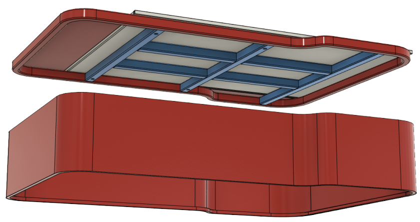
(B) Underneath view of subframe.

FIGURE 5.6: Different views of initial subframe model.(Gámez and Martínez, 2021)

Not many changes were made here, primarily the sections and dimensions of the tubes, to make the fabrication process simpler. In the initial design the central frame tubes had a recess cut on the ends, and the edge tube fit inside this recess, this was modified so everything was flush, which requires less cutting, also the load platform wall was extended to reach the edges. Furthermore, the load-bearing sheet metal platform will be attached by welding instead of rivets.



(A) Top view of subframe.



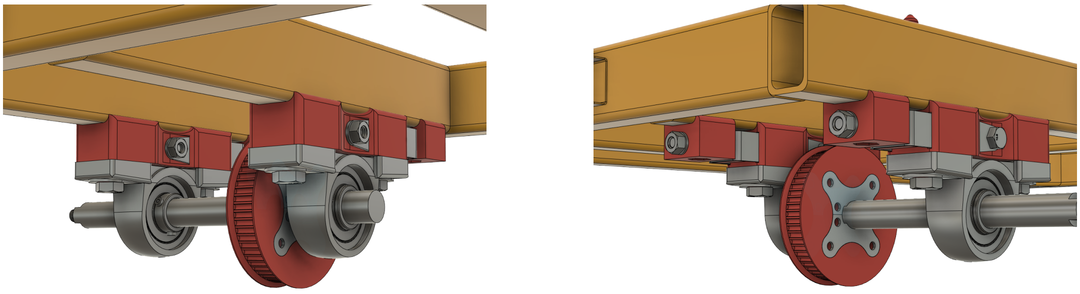
(B) Underneath view of subframe.

FIGURE 5.7: Different views of updated subframe model.

5.2.5 Powertrain

Axles and bearings

The base of the powertrain to which the rest of the components are attached are the axles. The design uses two round steel shafts, one for each side, as axles to which the wheels and pulleys are attached. It is vital that the drive shafts are strong, because most of the weight of the unit is transferred through them. The shafts are attached to the frame via industrial pillow block bearings. If the bearings were attached directly to the frame, the top surface of the chassis would not be parallel to the ground, due to the dimensions of the front caster wheels that are used. To solve this there are some solid plastic spacer blocks that set the correct height for the axles and keep the unit level. These blocks also serve the purpose of being part of the belt tension adjustment system.



(A) Front view of axle and bearings.

(B) Rear view of axle and bearings.

FIGURE 5.8: Different views of driveshaft mechanism.

As stated previously, the drive wheels are not designed to be fixed rigidly to a shaft and 3D printed parts will be used to overcome this. To carry this out, the shafts require some features so that they can be attached to the pulleys and wheels, and reliably transmit torque. In the previous design, the axles used a machined keyway and key system to attach the drive pulley and wheel to the driveshaft.

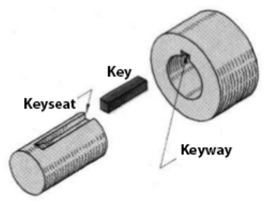


FIGURE 5.9: Example of a key and keyway coupling system (*Cutting A Keyway In A Shaft* 2022).

This option can be very robust and makes maintenance simple, but in the scope of this project it has various drawbacks. Adding a keyway to a shaft requires precision machining, and it would be necessary to use a well-calibrated tool, such as a milling machine, and the knowledge of how to operate it. If the tolerances are not correct, the drive system could develop backlash, leading to imprecise positioning of the AGV in space and would quickly wear different components of the powertrain. Furthermore, if this system were used, it would be necessary to have some metal parts that couple with the key to transmit torque to the 3D printed plastic parts, and these would be quite complicated to manufacture accurately. In view of these factors, a different coupling method was developed, the axles would have some parallel flat features on the top and bottom. It would still be necessary to employ some metal parts to transmit torque from the shaft to 3D printed parts and vice versa, due to the fact that prolonged repetitive pressure from a sharp edge on the metal shaft would wear the plastic and create backlash. Although, due to the simplicity of the parts, it would be simple to design metal parts to couple with. It seems that it might be possible to manufacture some parts from metal using a powerful laser cutting machine at ASSAR.

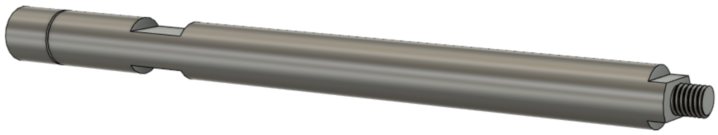


FIGURE 5.10: Flat features on driveshaft for coupling.

Wheels

There are different factors that need to be taken into account when choosing a wheel type. As was shown in the predecessor thesis (Gámez and Martínez, 2021), the ideal type is industrial wheels, due to their high weight capacity, temperature resistance, and durability. For the rear wheels, industrial wheels using some kind of keyway system would be necessary for the wheels to be fixed to and rotate with the axles. Nevertheless, these wheels have a big drawback, which is their price, starting in the region of 2000 SEK, two industrial grade wheels could cost over 5000 SEK, which is over a quarter of the expected budget for this project. As these wheels are very expensive just for development and testing purposes, the solution given to this problem is to adapt some generic trolley wheels, reused from Roboto (Dzezhyts, 2020), costing around 150 SEK, using 3D printed and metal parts to couple them to the driveshafts.

(A) Commercial industrial wheel with keyway.(*Keyed Drive Wheel* 2022)

(B) Trolley wheel and 3D printed part.

FIGURE 5.11: Industrial wheel vs budget adaptation.

In the front, the vehicle is going to use caster wheels, these are wheels that due to their design, self-orient in the direction of movement.

Industrial caster wheels are optimal because of their high load rating, durability and high traction finish. But, as before, their price makes them inadequate for use in this project, as a pair of them could cost over 6000 SEK, more expensive than the drive wheels due to having high load bearings built in. To stay within budget it was decided that some generic casters would be used, reused from Roboto (Dzezhyts, 2020), strong enough so that they won't fail under the loads that the unit will be able to carry.

FIGURE 5.12: Example of industrial caster wheel (*Hamilton Caster* 2022)

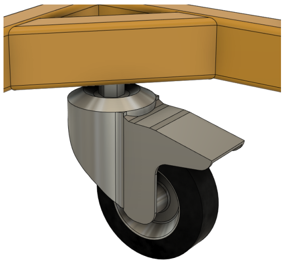


FIGURE 5.13: Generic caster that will be used.

Motors

There are many factors to consider when selecting a motor for this project. In the original design, some generic 500W **BDC** motors were used, with built-in reduction gearing, and external encoders were used to gather information regarding the position and velocity. This configuration provides several benefits, the first of which is simplicity, brushed DC motors do not require complex motor drivers to be used, and a simple power modulation circuit is sufficient, as seen in 2.4.1. The built-in gearing removes the need for additional reduction gears, reducing additional cost and weight. These motors are also relatively affordable, depending on the source, they can be found in the 300 to 500 SEK range, though they can be found for up to 3500 SEK if they are from a well-reputed manufacturer. The addition of an encoder would be mandatory for this type of motor, as it does not provide any kind of feedback, and this is required to be able to calculate the position and velocity of the **AGV**.

FIGURE 5.14: Brushed DC motor with reduction gearing. (*Reduction Electric Sprocket Brushed Reductor* 2022)

Finally, to reduce costs and take advantage of components that were already available, the motors that were chosen are those extracted from a used hoverboard. These motors were previously used in a similar AGV construction project (Dzezhyts, 2020).

These motors have properties that are quite interesting for this application, such as their price, due to the surge in popularity of hoverboards in recent years and the relatively short lifespans of lithium batteries, there is a large used hoverboard market, many of them with faulty batteries but with functional motors. Because of this, it is possible to obtain these motors cheaply. The hoverboard motors also have a low kV rating, which is a measurement of **RPM** per volt applied to the motor, of approximately $16kV$ (*Project HoverArm - Projects - ODrive Community* n.d.). The maximum angular velocity of a motor can be calculated using the formula.

$$MaxRPM = Voltage * kV = 36V * 16kV = 576RPM \tag{5.1}$$

In the context of **BLDC** motors this is a relatively low maximum RPM, thanks to this it is not necessary to introduce a large gear reduction to obtain the final linear velocity of 1 m/s, around 3:1 would be sufficient.

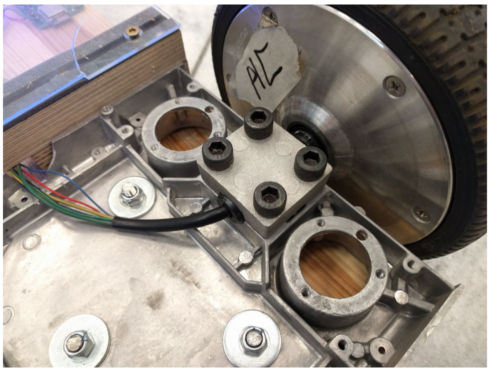
Pulleys

In the AGV design, the proposed method for transmitting torque from motors to wheels is a toothed belt and pulley system. This belt and pulley system consists of an initial pulley that is attached to the motor, a belt, and a second pulley that is attached to the driveshaft.

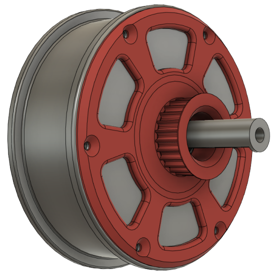
FIGURE 5.15: Generic belt and metal pulley combination

As a first step, it was necessary to decide what kind of pulley was to be used with the motor and how it would be attached. It would be difficult and expensive to use an off-the-shelf metal pulley for the motor for various reasons.

Firstly, the actual pulley costs in the region of 200-300SEK, and two of them are required, one for each motor. Secondly, to fix a metal pulley to the hoverboard motors that we are using it would be necessary to fabricate some kind of adapter plate from metal, which would then have to be attached to the motor using bolts or welds, furthermore the casing of the hoverboard motors are made from aluminium, which is notoriously difficult to weld. Due to these factors, it was decided that the most appropriate manufacturing method for this part would be 3D printing.



(A) Typical hoverboard motor use case (*Build Instruction: TranspOtter* 2022)



(B) Motor with 3D printed pulley model

FIGURE 5.16: Typical hoverboard motor use vs adaptation

3D printing the pulley has multiple benefits, as well as multiple drawbacks, which can be dealt with to some degree. One important advantage of this manufacturing method is the flexibility it provides, it makes it possible to experiment with different numbers of teeth, which lead to different gear ratios.Another benefit is that it does not require hardly any human intervention; the metal alternative would have required multiple hours of work to build, whereas with the chosen method other work can be carried out while the 3D printer makes the part.

Arguably, the main benefit of 3D printing the pulley in the scope of this project is cost, each motor pulley uses less than 70g of filament, if the chosen material is PLA, as it has been for the prototype parts, the price for each pulley is around 14SEK. The final version of the part will be printed in **P**oly**C**arbonate **C**arbon **F**iber filament, making the price for the part around 49SEK. The drawback of using a 3D printed pulley is the reduced strength compared to a metal alternative. The two most likely failures that could happen with a 3D printed pulley are failure due to delamination and failure due to deformation. Both of these problems are significantly mitigated by making use of the **PCCF** filament, due to its great layer adhesion, tensile strength, and temperature resistance.

Subsequently, the pulley that was to be attached to the drive shaft had to be analyzed. In this case it would be easier to make use of and off-the-shelf pulley, due to it being a standard arrangement where a pulley is linked to a round shaft of a common diameter. Nevertheless there are multiple arguments against employing a commercially available metal pulley. The first of which, again, is cost. A metallic pulley with the required dimensions has a price of around 300 SEK, and two of them are needed.

Another reason is the lack of flexibility; if it became necessary to experiment with different gear ratios, inevitably another set of pulleys would have to be purchased. Additionally, at this time it has already been decided that the motor pulley will be manufactured using 3D printing, and if there is already one plastic pulley in the drivetrain it would not make sense for this one to be metal, as it would not increase the overall reliability, due to the first plastic pulley already being the weakest link. After taking all these factors into consideration it was decided that the driveshaft pulley would also be 3D printed.

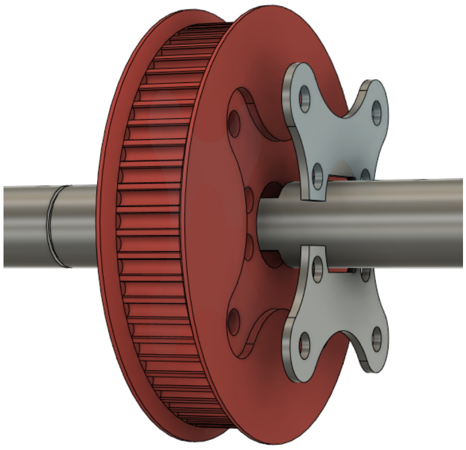


FIGURE 5.17: Driveshaft pulley with metal plates

Emergency brake

It is imperative that a vehicle of this type only moves when it is intended, in order to fulfil this the initial design included a pair of electromagnetic brakes, one for each side. When these electromagnetic brakes are activated the AGV is immobilized, creating a stable base on which to place material without risk of unintentional movement. In the scope of this build, due to budget constraints, these brakes have not been included.

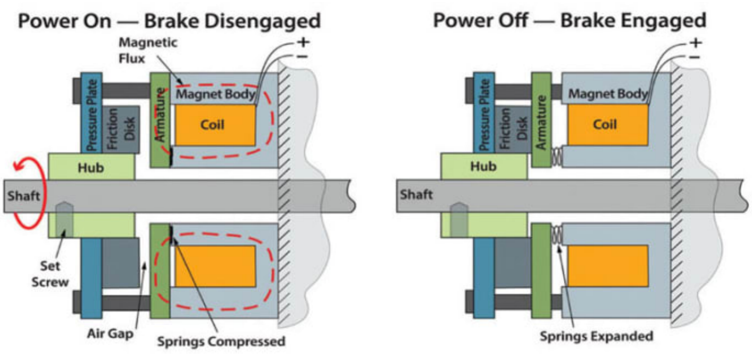


FIGURE 5.18: Emergency brake (Dragone,2018)

To include two electromagnetic brakes in this project would have a cost of approximately 5000 to 6000 SEK, though it may be possible to use a single brake with some kind of mechanism. Nevertheless, it is an important safety factor that the unit has the ability to lock its position, in order to achieve this the "brake" function of the motor controllers will be utilized. When the motor controller is in "brake" mode, it will apply a resistance to the motor phases, locking its position.

5.2.6 Power and Electronics

Once the mechanical design has been carried out, it is necessary to work on the electronics that the AGV will use to power and control it. The components and connections can be seen in the diagram at the end of this section (5.2.6).

Battery

For the unit to be independent from the power grid, it is necessary for it to carry its own power supply. Batteries have multiple characteristics to consider to choose the most appropriate one for this application. The first decision to be made is what battery technology will be used. In 2.4.2 the theoretical basis for the decisions made about batteries is presented, and for this particular use case it was decided that a battery based on lithium cells would be the most appropriate. Now, the type of battery has been chosen. Another defining specification for a battery is its nominal voltage, in this case it is given by the working voltage of the motors that are going to be used. The hoverboard motors have a nominal voltage of 36V, so a battery of this voltage is required.

The next specification of the battery to be decided is the capacity. The capacity of the battery defines the time the AGV will be able to operate between charges, at the expense of weight and cost (5.2). During the design phase, the current consumption of the AGV is unknown, so it is necessary to make an estimation to be able to select the optimal capacity of battery for the unit to be able to operate for an adequate amount of time. Consumption depends mainly on weight, the resistance in the powertrain, and rpm at which the motors will operate. Motors are estimated to consume a current of around $0.5A$ each when there is no extra weight being carried, and the electronics should consume a current of approximately $1A$. This would put the total consumption of the **AGV** carrying only its own weight at $2A$ continuously when in motion. Knowing this, it is possible to decide the capacity of the battery as a function of the desired run time, which could be for example, 6 hours (5.3).

$$Operating\ time\ (h) = \frac{Capacity\ (Ah)}{Current\ (A)} \tag{5.2}$$

$$Capacity\ (Ah) = Operating\ time\ (h) * Current\ (A) = 6\ h * 2\ A = 12\ Ah \tag{5.3}$$

The real current that the final unit will consume is currently unknown, but a $12Ah$ lithium battery is a good compromise between price, weight and capacity which will make the testing process easier due to now having to use a fixed power supply connected to a wall outlet or having to frequently stop to charge a smaller battery.

FIGURE 5.19: Commercial 12Ah 36V battery. (*36V high energy 12.0Ah battery* $|36VToolBatteries|RyobiSpain$ 2022)

Motor controllers

As seen in 2.4.1, the type of motor controller needed depends on the motor that will be used. In this case the motors that have been chosen, as seen in 5.2.5 are hoverboard motors, which are $350W$ **BLDC** motors with built-in hall effect sensors for feedback. There are many options on the market to control this type of motor. An affordable open source option could be that offered by ODrive robotics (2.11), but there are many more. Finally, to reduce costs, as with other components in this project, the motor controllers are also going to be repurposed from roboto (Dzezhyts, 2020). The controllers that will be used are manufactured by Maxon Group, specifically the DEC 50/5.

FIGURE 5.20: Maxon motor controller

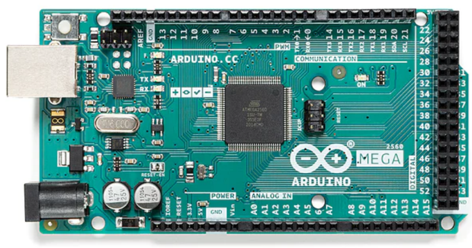
These motor controllers can output up to $250W$, which means that they will not take advantage of the full power of the motors, but this has a benefit which is that the temperature will not rise above the manufacturer recommended limits. These controllers have inputs for the motors hall effect sensors, meaning that they can provide closed loop velocity control, taking as a reference an input signal between 0 and $5V$.

Logic Boards

For this project to function as a fully capable project **AGV**, it is necessary to have a certain amount of processing power, read, store and process all the data from multiple sensors, create maps, calculate trajectories and motor velocities, and possibly send data to some kind of cloud-based platform.

Furthermore, the unit is expected to run **ROS** (2.6.1), which requires some kind of PC running the Linux operating system. An affordable, *mostly* **OS** option for this could be a Raspberry Pi 4, though in practice it seems that they are somewhat lacking in computational power, and it would be required to heavily optimize the software for it to run smoothly, if it could run at all. To avoid this problem and reuse parts that were made available, a Nvidia Jetson Nano, taken from roboto (Dzezhyts, 2020) is going to be used. This board offers enough resources to carry out the necessary tasks without externalizing any processes. This computer is not strictly speaking open source, but it is readily available and easy to obtain. Apart from the Nvidia board, an Arduino Mega will be included to deal with interfacing with the motor controllers, connected to the Nvidia board via a serial connection.

(A) Nvidia Jetson Nano (*NVIDIA Jetson Nano Developer Kit |NVIDIADeveloper* 2022)



(B) Arduino Mega. (*Arduino Mega 2560 Rev3 – Arduino Official Store* 2022)

FIGURE 5.21: Logic boards used in the project

Circuit scheme

Due to the re-purposing of the materials of other projects such as **Roboto** (Dzezhyts, 2020), the electric circuit would be almost the same as the one described in the previous project. In this case, the parts included in the scheme refer to the minimum required to achieve driving functionality. To feed the Jetson Nano with the required voltage and current, a step-down component is included in the design. It is omitted from the next scheme for simplification purposes.

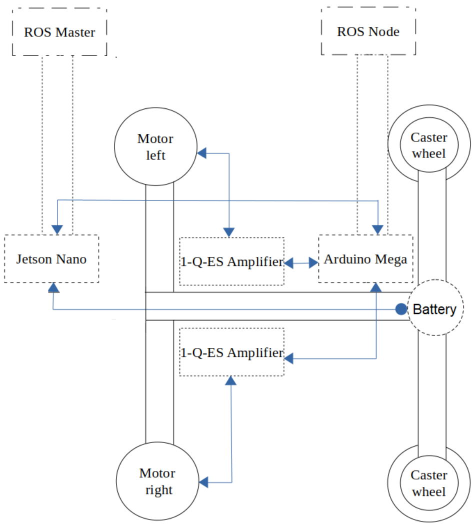


FIGURE 5.22: Basic electric scheme (Dzezhyts, 2020)

5.3 Manufacturing and Assembly

5.3.1 Introduction

Once the design phase is advanced enough, it is time to start construction. The design phase can't be said to be completely finalized at any point during a project which involves a prototype, due to the fact that there are constant revisions.

Having a physical artifact gives a much more intuitive view of what has to be done to make progress; this leads to the need to correct possible oversights from the design phase that would go unnoticed unless somebody were to attempt to build the unit.

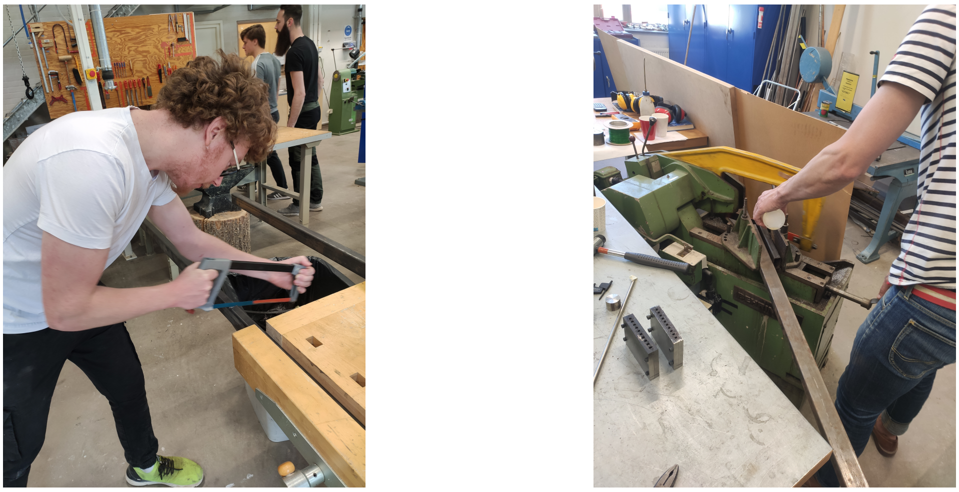
5.3.2 Materials

Due to the Open Source nature of this project and parametrized CAD files, materials and dimensions can be quite flexible. In AppendixB there is a complete **BOM** with the parts that were used in this project, along with the approximate cost of the final unit.

It is important to note that attempting to build this requires a diverse selection of tools and manual skills, such as 3D printing, cutting, drilling, machining, and welding different metal parts. If somebody were interested in acquiring an **AGV** of this type, the most practical option could possibly be to provide the project files to a third party metal fabrication company or similar, who already have the necessary infrastructure and skills, at least for the most labor intensive parts, such as the frame and subframe construction.

5.3.3 Metal fabrication

As mentioned in 5.2.3, the base of the unit consists of a welded steel tube chassis. Once compatible tubes were obtained, it was necessary to cut them to length. This was initially done by hand, using a manual metal saw, but this was cumbersome and time-consuming due to many cuts being necessary. Fortunately, thanks to Zimon at the university workshop, it was possible to make the cuts using a machine known as an electric power hacksaw, which made the process much easier.



(A) (B)

FIGURE 5.23: Manual and machine metal cutting

Once all the frame components were cut to the correct length, they had to be welded in the final position using the 3D model as a reference for the measurements. In this case the welding technology that was used was .

FIGURE 5.24: Main frame weld

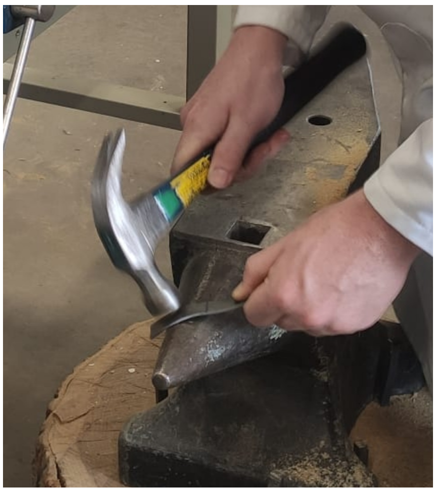
Multiple holes of different sizes have to be made in the frame to attach different components, such as the bearings, the tensioning blocks, the front caster wheels, and the rear tow hook (optional). These holes were mading using a standard electric hand drill, marking the holes with a center punch to make it easier to start the hole and ensure that it is in the correct position. To be able to access the bolts from the top of the frame to tighten the nuts, it is necessary for the holes to be large enough to fit a tool inside, it can be hard to find drill bits large enough to fit, for example, the correct size hexagonal socket inside. To solve this there are multiple options, possibly the fastest could be to make the holes using a plasma cutter, another option would be to use a hole saw with a drill, and the alternative that was used in this project was a step drill bit.

FIGURE 5.25: Step drill tool

Once the frame is finished, what metalwork is left to do is that of the subframe. As seen in 5.2.4, it has a similar structure to the frame, though with smaller section tubes, the addition of a small tube brim to keep objects from sliding off, and sheet metal components. The tubes were cut and welded the same way as the frame tubes, using a power hacksaw and welder. Ideally, for making the curved outer tube, a tube bender would be used. Which can curve hollow tubes without causing them to collapse, maintaining their structural integrity

FIGURE 5.26: Tube bending machine. (*KAKA Industrial TR-60 Tube / Pipe Roll Bender, Versatility Bender, Hig* 2022)

Unfortunately, this tool was not available, as an alternative the curved parts were fabricated using two manually curved sheet metal pieces for each bend, and then they were welded to the tubes. These parts do not have to carry any significant load, only the weight of the sheet metal skirt, so this solution is sufficiently adequate for a prototype.



(A)

(B)

FIGURE 5.27: Curved metal and subframe weld

The top sections of sheet metal have a thickness of 2mm, and were cut using a plasma cutter. It would be difficult to cut parts of this thickness using a shear, adding the fact that some of them have curves, which cannot be cut with a shear. Once the sheet metal parts are cut to the correct shape, they were welded to the subframe.

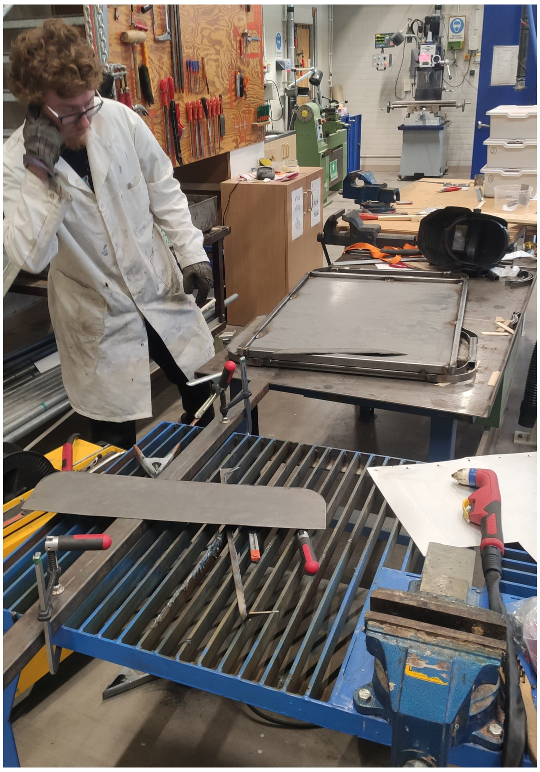


FIGURE 5.28: Plasma cutting and welding

The last step in building the subframe is the sheet metal skirt, which consists of 0.7mm sheets that were cut to the necessary width and height and then attached to the subframe using rivets. There is a small metal section on the front of the unit attached with a hinge, where the battery is housed. This hinged door allows for easy access to remove and insert the battery for recharging. Other metal parts that are required are the C shaped strips for the tensioning system, they are made from perforated metal strip. They were cut using a shear and bent into shape by hand using pliers.

FIGURE 5.29: Subframe and mounted skirt

The last metal part that had to be modified was the powertrain shafts. The material used to build this part was 20mm round rods that were acquired from Nystedts, a local metal supplier. The round rods had to be machined in order to have the thread, which would hold the wheel in place, and the square flat features which would enable the pulley to stay in place.

FIGURE 5.30: Powertrain shaft: Rod machining

5.3.4 Painting

When carrying out a technical project, painting may be overlooked due to it being seemingly a step that only affects aesthetics, but it is just as important as the rest of the process in the long term. Both the frame and subframe are built from untreated, mild steel, and if they where left without any protective coating in a short time they would start to corrode and break down. To avoid this, the metal was first treated with a coat of primer, which seals the surface and creates a uniform base to which the paint can properly adhere to.

(A)

(B)

FIGURE 5.31: Primer coating application

Once the primer was dry, the parts were given a coat of paint, and finally, once this paint was dry, the whole surface was coated with clear lacquer, which gives a transparent layer which protects both the metal and the paint. Painting **AGV** with a high contrast color also serves as a safety measure, making it clearly visible from a distance so that any operators nearby can easily avoid it.

FIGURE 5.32: Paint coating

5.3.5 Assembly

Once the frame, subframe, 3D printed parts, and electronics are ready, it is time to begin assembly. In this context, the 3D model can be considered as the instructions, the final objective is for the real object to look like the model. Nevertheless the assembly will be explained, highlighting parts that can prove to be more tricky or less intuitive.

Motors

First of all, the two "Motor pulleys" (D.1a) must be attached to the hoverboard motors, this is done by removing the M4 screws from the motors, placing the 3D printed pulley, and inserting some slightly longer conical M4 bolts. The motors attach to the frame via the small section of tube that is welded underneath, using the 3D printed parts named "Motor holder top" and "Motor holder bottom"(D.4b) in AppendixD. The fitment for these parts has very tight tolerances, so that the pressure against the walls of the tube fixes the motor in place.

The easiest way to assemble this is to first place "Motor holder bottom" on the motor shaft and place this assembly inside the tube, being careful to pass the cables through the tube without pinching any of them. While this is being held in place, "Motor holder top"(D.4a) can be aligned with "Motor holder bottom", entering from the middle of the frame, checking that they are in the correct orientation, and then "Motor holder top" can be gently tapped with a plastic hammer until it is all the way in the tube.

Axles and bearings

This step is quite intuitive; first, the four 3D printed "Bearing block"(D.2a) parts can be placed on the frame, along with the four "Tension blocks"(D.2b) roughly lined up with the drilled holes. It is important to note that in this project the "Tension blocks" were substituted for some metal L brackets. Then, the eight "Bearing sleeve"(D.5a)

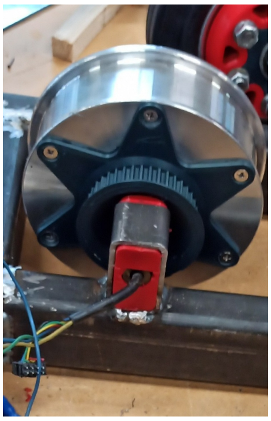


FIGURE 5.33: Motor holder main frame

parts can be placed inside the holes; these parts make it easy to line the bearings, and they also create a better fit for the M10 bolts that are used to fix the bearing assemblies to the frame.

It is best to leave the bolts loose so that the tensioners can tighten the belt before fixing everything in place, this also makes it easier to pass the axles through the bearings, as they may bind due to the tight tolerance. It is important not to forget to pass the shafts through the HTD5M belts when assembling them with the bearings. Once the belts have been tensioned, it is time to tighten the M10 bolts that attach the bearings to the frame.

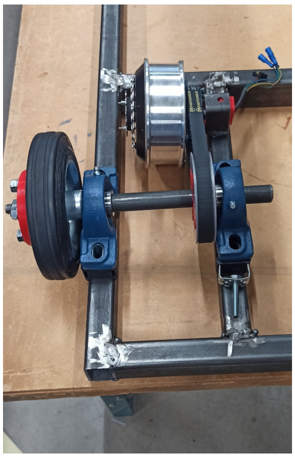


FIGURE 5.34: Tensioner placement

The six "Alignment pins" are press-fit into their respective holes at the top of the frame, the subframe has matching holes so that it can be simply placed on top of the frame without slipping off.The rest of the components are simply placed as they are in the 3D model, some electronics components are zip-tied directly to the frame, being careful not to stick out over the top so that they are not pinched by the subframe when it is placed.

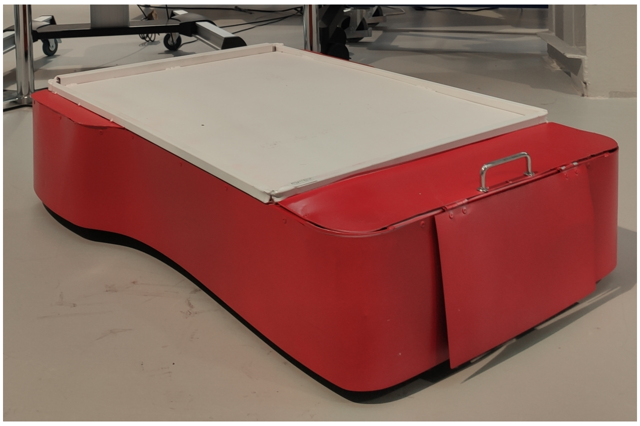


FIGURE 5.35: Full assembly of the **AGV**

5.3.6 Basic programming

In order to perform functional tests with the final **AGV** unit, it was necessary to install both the operating system and the **ROS** environment on the Jetson Nano. This task was carried out with the help of the tutorial described in the final master project of Dzezhyts, 2020, in which the necessary packages were listed and introduced properly in order to control the unit.

The PS4 controller mode was first implemented, as the aim was to control the unit to verify that the electronics systems were working. To do so, it was mandatory to connect and program the Arduino which would be the board in charge of driving the motors with the commanded signals coming from the Jetson Nano. In Appendix C, the initialization steps are described.

Although programming was beyond the scope of the project, additional navigation programs were included and tested along the available **LiDAR** sensor. The next project would consist of developing this part with the aim of automating the unit.

Chapter 6

Results

6.1 Testing and validation

The testing methods were not very rigourous, though sufficient to get a rough idea of the physical limits of the unit. This **AGV** is designed to be used as both an underride and tugger style vehicle, this means that it is necessary to test its capacity to tow objects, and also its capacity to carry weight on the top.

Underride capacity

Multiple tests were carried out to situate the maximum weight under which the unit can still move adequately. First, a box with metal engine parts, estimated to weigh around 30kg. This did not affect the dynamic behaviour at all. Next, it was tested with firstly one, then two, and then three people, weighing in the region of 220kg. When there were three people on top, the motors started to stall when turning, because they are not both turning in the same direction. These tests lead to the conclusion that this vehicle can successfully transport 200kg of load to the top. It is likely that with higher power output motor controllers this number will increase.

Towing capacity

A tow hook was installed in the rear so that it would be possible to attach a rope to things to be towed for testing, in reality it would not be practical to attempt to move materials by towing with a rope, because it leads to unpredictable movement, the trailer or pulled object would have to be attached using a fixed length bar or tube, although for testing purposes a rope was sufficient.

At ASSAR there are two Volvo car engines mounted on engine repair stands with wheels, the first test was to try to tow one of these engines, estimated to weigh around 250kg, including the stand. The vehicle was able to tow this without a problem; no extra weight was necessary for traction. In a later test, both engines where attached to the **AGV** with a rope, there was around 60kg of weight on top of the subframe, and there were two people on top of the engine trolleys. This is a combined weight of around 700kg, and the vehicle was able to move this mass around 5 meters in a straight line. Demonstration video

When measured with a dynamometer tied from the back of the **AGV** to a fixed column, around 250N of towing force was measured, with 80kg of weight on top for traction.

The set of objectives prearranged in the specification of the project are going to be revised and discussed in this section.

FIGURE 6.1: **AGV** towing capacity

FIGURE 6.2: Dynamometer towing capacity measure

6.2 Methodology and results evaluation

The methodology, for the most part, has been quite consistent. The documentation phase didn't strictly follow **DSR**, the problem trying to be solved was the research question, the literature review and frame of reference were not aiming to directly solve the research question, instead their value is to help to provide a path to follow which should lead to some results that can help to evaluate the aim of the project. The methodology was most rigorously applied in the implementation chapter.

The implementation was by far where the most work was carried out and as it consists on building an artifact for a specific purpose, the steps which are defined in the chosen methodology almost arise naturally without applying them explicitly. **DSR** has a very intuitive workflow when applied to this type of project, which indicated that it is a well-fitting methodology.

In reference to the results obtained, due to several relevant limitations, the project was not carried out in the most optimal way, which would have been to strictly follow Gámez and Martínez, 2021 and reproduce their theoretical design in real-life. By doing this it would have been possible to save multiple weeks of work adapting parts, also then the evaluation of the project could more directly answer the research question.

Nevertheless, the fact that it was possible to adapt parts to imitate the proposed parts in the initial design, is itself an indication that **OS** is beneficial for developing new projects, as much inspiration was found in other projects found online, and also the dimensions for different parts due to their plans being publicly available.

This project is one of very few that aims to make progress in the field of **OS** industrial **AGV**s, hopefully this will push others to continue making progress in this field until a definitive solution is reached.

Previous design analysis

This was one of the most time consuming parts of the project. Although the main objective was remaining faithful to the previous design and maintaining the **CAD** as it was given, this was not possible due to the necessity of adapting the requirements to the budget that was available.

In order for the design to be fully parametric, starting the 3D modeling from scratch was necessary. The parametric design allows the user to select the size of the unit as desired, depending on the requirements and use of it. The initial concept was followed taking into account the available parts that had to be measured and introduced into the **CAD** model.

Regarding scalability concerns, the final 3D model file allows fragmentation of the unit into several parts that could be easily changed by the available parts or fabrication methods. It is up to the final consumer to bring the files to a metal professional to have the structure built with the wanted requirements and characteristics, and then arrange the circuitry depending on the use that the unit would have in the working environment.

The most advantageous part will be the possibility given to the **OS** community of developing this project due to its submission to GitHub.

Construction of the AGV vehicle: structure and electronics

Thanks to the help of the university staff and the re-purposing of the available parts of other projects, this task was successfully achieved. Metal labor related activities were time consuming and difficult sometimes but the experience gained in this field is priceless. The opportunity of welding, plasma cutting, drilling, and metal cutting was very pleasant and gave an idea of what it takes to materialize a design.

The worst part was that it was still in a very narrow budget range, but at the same time this pushed the design ideas, moving from using off-the-shelf parts to 3D printed alternatives, which finally was the best choice because it eliminated problems such as delays in deliveries or out of stock and kept the overall **BOM** even lower than expected.

Initial programming

Although this part was out of the scope of this project and was done as an additional feature, with the help found in Dzezhyts, 2020's master's degree thesis, functional testing was achieved. At the same time, it allowed one to set the basis of the program structure and at least include the main **ROS** packages needed to further autonomous application development, the next step to make the unit an **AGV**. y se queda este aqui

Chapter 7

Conclusions

In this chapter there will be discussion about the findings of the project and what impact they have on the research.

7.1 Evaluation of objectives

The following objectives were set in 1.4 to try to answer the research question, here is discussed to what extent the objectives were met.

- **Manufacture a prototype AGV**.

  Firstly, it is important to point out that one big challenge in this project was the actual build of the prototype. The nature of this project somewhat different than other final year projects that are carried out at the University. Due to the lack of precedent of other projects requiring metalworking tools, which can be dangerous if certain safety guidelines are not followed, it can be quite difficult for students to get access to the resources required to carry out a project like this.

  A lot of steps were involved in being able to make each part and supervision by University staff was always necessary. It would have been possible to make a less sturdy prototype, for example by using aluminum extrusion joined by bolts, but this idea was not followed in order to follow the original design as closely as possible. Also at the time it was not known that there would be so many setbacks building the structure. In a different environment, where somebody could have unlimited access to a well equipped workshop, or even a professional metalwork business, a reproduction of this prototype could likely be manufactured in 1-2 days of work.

  In the end, a prototype was built, though at the time of writing this document it does not fulfil the requirements to be considered an **A**utomated **G**uided **V**ehicle. Due to limited time to continue developing the project, only basic software was implemented, so it is lacking the ability to navigate paths in an automated fashion. Nonetheless, the work done is an important step towards the final objective, making it possible for future efforts to develop the necessary software implementing different sensors to make it a fully capable **AGV**.

  Though the design developed in Gámez and Martínez, 2021 was not followed rigorously, due to the inability to purchase the stipulated parts or similar, the prototype uses hardware that is mostly **OS**, and when it was not possible, parts that were easily obtainable were used. For example, the motors are not strictly **OS**, but various accurate cad models developed by the community are easily accessible, along with other technical information. The main exception would be the Maxon motor controllers, which are closed source, but can be substituted by many **OSH** alternative, such as Odrive (*ODrive* 2019). The other main exception is the Jetson Nano, which could also be substituted by any of multiple alternatives that have been mentioned earlier in the project, such as Revolution Pi, or an Arduino-based **PLC**.

- **Evaluate performance of the prototype.**

Since implementing complex software was outside of the scope of this project, the performance evaluation that can be done is that which refers to the physical capabilities of the prototype. The design by Gámez and Martínez, 2021 enables the unit to be used either in a underride configuration, which means that it would carry the load on the top, in the white walled area; or in a tugger configuration, where it would tow some kind of trailer which contains the load.

When used in the underride configuration, the prototype was able to successfully transport in excess of 200kg without negatively affecting the kinematic behaviour. In an overload situation, where the unit was not able to continue moving due to excess weight, the limiting factor was the power of the motors. The power of the unit could be increased by using higher current motor controllers, as the current Maxon controllers are underpowering the motors; or a higher torque motor and motor controller pair could be implemented, which would be relatively simple due to the belt and pulley transmission being used. With more power, the weight capacity in an underride configuration is expected to be in excess of 400kg.

When used in the tugger configuration, the prototype was able to displace in excess of 600kg, though this may not be an accurate representation of the load capacity, because there are multiple factors that can alter this amount, such as friction from the "trailer" wheels. A maximum towing force of 250N was registered using a dynamometer. Depending on the floor surface and the amount of weight being towed it may become necessary to introduce weight on top of the prototype to increase traction.

As mentioned previously, the prototype is lacking software development to have the necessary features to be considered an **AGV**, though its phyisical capabilities are quite promising and it wouldn't take a lot of work to implement what it is missing. It is expected that in the future a very practical tool could result from this project.

- **Evaluate viability of an affordable industrially capable AGV using OSH**

  There is still a long journey before this prototype could perform successful automated logistics operations in an industrial environment.

  The implementation of sensors and extensive software progress would be mandatory for this unit to perform automated tasks. This could define as well a new project branch in the attempt of having a functional autonomous logistic solution in an industrial environment. The foundation has already been included by preparing a functional **ROS** framework where further development could be made to include **SLAM**, path following, map generation or even obstacle avoidance programs.

  The fact that this has been the first manufactured prototype according to the previous existing design, and many changes had to be made because of the low budget, there are plenty of improvements that could be done to optimize the manufacturing process of several units in series in case that a company (e.g. ASSAR) would need a fleet of this logistics solution going around its facilities.

  Future iterations of this project would improve the design to the point that it could compete against commercial available solutions and even force bigger logistics devices manufacturers to adapt to the markets requirements by developing affordable units for the **SME**, allowing them to optimize their processes increase the overall performance and benefits for a relatively low investment.

7.2 Evaluation of the research question

This project was created to answer the question **What are the challenges of developing an Open Source Hardware AGV which could be used in industry by SME?** Going through the process of actually developing an **OSH AGV** has given significant insight to facilitate providing an answer to this question. In the following sections, the main challenges are outlined.

7.2.1 Budget

In the context of this project, one relevant challenge with the development itself was working with components that may not have been the best fit in every case, due to limited resources. This led to the need to create workaround solutions to adapt available parts to perform functions they were not necessarily designed for. Nevertheless, development costs and possible standardization costs aside, the estimated unit price for a future revision **AGV** based on the development carried out in this project would be significantly lower than that of commercially available alternatives, making adding an **OSH AGV** to their logistics system an interesting option to pursue for **SME**.

7.2.2 Design

The mechanical and electrical design in this project is the basis on which everything else is built. Creating a platform capable of movement is not a novel concept, nor is doing so with **OSH**. What is unprecedented is creating a machine of this type with intent for it to be used in an industrial environment, this introduces many design challenges that have to be considered. An **AGV** intended to be used to move small boxes is not designed in the same way as one intended to move pallets. The design used in this project offers a rugged and simple, low-cost base. Thanks to the parametric design it can easily be re-scaled for different applications, for less demanding use cases it can simply be built using more lightweight components, and for heavier loads, higher weight-rating components can be installed. Also, due to the **OS** nature of the project, it would be possible to design modular attachments, such as a lifting mechanism. The unit constructed in this project had a cargo area of 570x770mm and can transport a weight of up to 200kg, which may be interesting characteristics for some **SME**.

7.2.3 Standardization

A major challenge in this project is standardization. This problem is quite complex and extensive enough that it could be covered in another project by itself because of the necessary research and verification steps that it requires. This would increase the development costs due to the requirement of hiring an external company to carry out tests and study the feasibility of this design to comply with industrial norms. Standardization also raises an issue regarding the flexibility of the design, any significant changes may require new testing.
As the main target of this project are Small and Medium-sized Enterprises it is possible that in some cases, **SME** might prefer to use these devices without official industrial standardization at their own risk, so that they could be able to to optimize their processes with a low investment. Safety tests and validation would be recommended to ensure the protection of workers in the facilities where these units would operate.

7.2.4 Software

Software development may be the largest challenge in regards to making an **OSH AGV** actually usable in industry by **SME**. For a machine of this type to be useful it must have a user-friendly software experience. In this project, **ROS** was installed on the Jetson Nano board, it was a complicated and tedious process, with no interface for the user other than the terminal and requiring the installation of many packages. Ideally, a custom installer would be created that would perform the task automatically, taking various parameters as an input such as physical dimensions and the number, type, and location of different sensors. In addition to the challenge of the onboard software **AGV**, there must be control software that creates paths based on the users requirements. One **OS** option for this could be OpenTCS (IML, 2012), which is compatible with **ROS**. However, a user-friendly interface would still necessary to make it simple for non-programmers to control one or multiple AGVs.

7.3 Future work

Future work tasks will be introduced, with the hope of this project to be continued. The main tasks would be standardization and programming, which could be two separate projects without any problem because of the span of these two fields.

Standardization

Although in Miguel's work (Gámez and Martínez, 2021) this field was widely described and discussed, it will be necessary to take action and gather the requisites to use this unit in an industrial environment. After this, it would be necessary to contact an authorised institution to verify the necessary steps and what it really takes to make this possible within the actual EN ISO standards. The most important part of all of this would be the overall price of the unit after performing all the necessary tests and verification processes.

Programming

The basis has already been established for the unit to work with the controller. The next step would be to implement the necessary sensors and actuators to be working autonomously. Thanks to the **ROS** framework, sensors and actuators would be able to talk to the main logical unit very easily. The though part would be implementing proper **SLAM** algorithms, path following and map creating, always ensuring the safety of the materials and workers around the unit.

It would be very interesting to merge this project with others related to augmented/virtual reality. This would expand the set of uses of this device in a real industrial environment.

Bibliography

1175:2020, SS-EN (n.d.). *Safety of industrial trucks - Electrical/electronic requirements*.

3 Best Industrial Raspberry Pi Solutions in 2021 (2021). URL: https://jfrog.com/connect/post/3-best-industrial-raspberry-pi-solutions-in-2020/\#:~:text=The\%20Revolution\%20Pi\%20is\%20an,Secure\%20Digital\%20(SD)\%20storage. (visited on 06/23/2022).

36V high energy 12.0Ah battery $|36VToolBatteries|RyobiSpain$ (2022). URL: https://es.ryobitools.eu/baterias-y-cargadores/baterias-de-36v-y-cargadores-max-power/baterias-de-36v-max-power/ry36b12a/ry36b12a/ (visited on 06/27/2022).

Aetina AN110-XNX-EN70 NVIDIA Jetson Nano. (2020). URL: https://www.steatite-embedded.co.uk/products/industrial-jetson-computers/aetina-an110-xnx-en70/ (visited on 06/23/2022).

AGV Brushless DC Motor Benefits (2020). URL: https://www.orientalmotor.com/brushless-dc-motors-gear-motors/technology/brushless-dc-motors-agv-designs.html (visited on 06/20/2022).

AGV Design and Component Considerations OEM Automatic Ltd (2020). URL: https://www.oem.co.uk/resources/blog/design-considerations-for-agvs (visited on 06/23/2022).

AGVNetwork (2022). *AGV Sensors – The eyes and ears of mobile robots*. URL: https://www.agvnetwork.com/mobile-robot-sensors-agv-amr (visited on 06/22/2022).

Alves, Thiago and Thomas Morris (2018). “OpenPLC: An IEC 61,131–3 compliant open source industrial controller for cyber security research”. In: *Computers Security* 78, pp. 364–379. ISSN: 0167-4048.

Andrey Solovev, OpenSystems (2019). *DC Motor Controllers: Brushed vs Brushless - Embedded Computing Design*. URL: https://embeddedcomputing.com/application/industrial/motor-control/dc-motor-controllers-brushed-vs-brushless (visited on 06/21/2022).

Arduino Mega 2560 Rev3 – Arduino Official Store (2022). URL: https://store.arduino.cc/products/arduino-mega-2560-rev3 (visited on 06/27/2022).

Automation, Monitoring and Control with Industrial PLC Arduino (2022). URL: https://www.industrialshields.com/industrial-plc-based-on-arduino-original-boards-automation-solutions-202205-lp (visited on 06/23/2022).

Barragán, Hernando (2020). *The Untold History of Arduino*. [Online; accessed 2022-06-06].

Behroozpour, Behnam et al. (2017). “Lidar System Architectures and Circuits.” In: *IEEE Communications Magazine, Communications Magazine, IEEE, IEEE Commun. Mag* 55.10, pp. 135 –142. ISSN: 0163-6804.

Brocke, Jan vom et al. (Jan. 2012). “Green Information Systems: Directives for the IS Discipline”. In: *ICIS 2012 Proceedings* 33.

Brushless DC Motor vs. Brushed DC Motor (n.d.). [Online; accessed 2022-06-11].

Brushless Vs Brushed DC Motors: When and Why to Choose One Over the Other | Article | MPS (n.d.). [Online; accessed 2022-06-11].

Budisusila, Eka Nuryanto et al. (2021). "Ultrasonic Multi-Sensor Detection Patterns On Autonomous Vehicles Using Data Stream Method". In: *2021 8th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, pp. 144–150.

Build Instruction: TranspOtter (2022). URL: https://github.com/lucysrausch/hoverboard-firmware-hack/wiki/Build-Instruction:-TranspOtter (visited on 06/24/2022).

Cho, Hyunhak et al. (2017). "Improved positioning method for Magnetic Encoder type AGV using Extended Kalman Filter and Encoder Compensation Method." In: *International Journal of Control, Automation and Systems* 15.4, pp. 1844 –1856. ISSN: 1598-6446.

Commission, European, Directorate-General for Internal Market, and SMEs (2017). *User guide to the SME definition*. Publications Office.

Compact drives by maxon maxon group (2022). URL: https://www.maxongroup.com/maxon/view/content/compact-drives (visited on 06/21/2022).

Cutting A Keyway In A Shaft (2022). URL: https://www.redlinetools.com/resources/key-cutters (visited on 06/24/2022).

Deek, Fadi P. and James A. McHugh (2008). *Open Source : Technology and Policy.* Cambridge University Press. ISBN: 9780521881036.

Dzezhyts, Y (2020). *"Next generation low-cost automated guided vehicle. Master's thesis"*.

Electromechanic Pin Brake for AGVs and Robots Kendrion (2021). URL: https://www.kendrion.com/en/products/solenoids-actuators/locking-solenoids-door-lock-systems/electromechanic-pin-brake (visited on 06/23/2022).

Gámez, Miguel and Miguel Martínez (2021). "OPEN HARDWARE AGV". In.

Gregório, João et al. (2021). "The role of Design Science Research Methodology in developing pharmacy eHealth services." In: *Research in Social and Administrative Pharmacy*. ISSN: 1551-7411.

Guido, Gürtler (2011). *Standardization - Challenge and Opportunity : What Managment and Staff Should Know. Including 11 Practical Examples and 39 Practical Tips.* Quick-Info. Austrian Standards plus Publishing. ISBN: 9783854022312.

Hamilton Caster (2022). URL: https://www.hamiltoncaster.com/Part-Detail/PartID/S-CH-83DB (visited on 06/24/2022).

Hannig, Frank and Jurgen Teich (2021). "Open Source Hardware". In: *Computer* 54.10, pp. 111–115.

Heron, Michael J., Vicki L. Hanson, and Ian Ricketts (2013). "Open Source and Accessibility: Advantages and Limitations". In: *Journal of Interaction Science* 1.1, p. 2.

Hevner, Alan R (2007). ""A Three Cycle View of Design Science Research"". In: *"Scandinavian Journal of Information Systems"* 19.2.

Hevner, Alan R. et al. (2004). "Design Science in Information Systems Research". In: *MIS Quarterly* 28.1, pp. 75–105. ISSN: 02767783.

How Czech startup Prusa Research hit 17,122 | Sifted (n.d.). [Online; accessed 2022-06-07].

IML, Fraunhofer (2012). *openTCS - The open transportation control system*. URL: https://www.opentcs.org/en/index.html (visited on 08/16/2022).

ISO 13849-1:2016, ISS-EN (n.d.). *Safety of machinery - Safety-related parts of control systems - Part 1: General principles for design*.

ISO 3691-4:2020, SS-EN (n.d.). *Industrial trucks – Safety requirements and verification – Part 4: Driverless industrial trucks and their systems*.

Jiang, Zhenghong, Yucheng Xu, and Lei Sun (2021). "A Novel Path Tracking Controller for Magnetic Guided AGVs". In: *2021 33rd Chinese Control and Decision Conference (CCDC)*, pp. 3292–3296.

KAKA Industrial TR-60 Tube / Pipe Roll Bender, Versatility Bender, Hig (2022). URL: https://www.kakaindustrial.com/products/kaka-industrial-tr-60-tube-roll-bender-and-versatility-bender-high-adjustability-portable-tubing-pipe-bender (visited on 06/27/2022).

Keyed Drive Wheel (2022). URL: https://www.casterhq.com/6-Inch-Keyed-Drive-Wheel-18mm-Bore-p/hd-615-ps-kdw-18mm.htm (visited on 06/27/2022).

Koubâa, Anis (2016). *Robot operating system (ROS) : the complete reference.* Studies in computational intelligence: v. 625. Springer. ISBN: 9783319260525.

Magnetic Guide Sensors (2021). URL: https://www.roboteq.com/all-products/magnetic-guide-sensors (visited on 06/22/2022).

Maxon House, Editor (2017). *Motor selection for automated guided vehicles*. URL: https://www.machinebuilding.net/motor-selection-for-automated-guided-vehicles (visited on 06/21/2022).

MPS Open-Source Ventilator (2021). URL: https://www.monolithicpower.com/en/mps-open-source-ventilator (visited on 08/16/2020).

NVIDIA Jetson Nano Developer Kit |NVIDIADeveloper (2022). URL: https://developer.nvidia.com/embedded/jetson-nano-developer-kit (visited on 06/27/2022).

ODrive (2019). URL: https://odriverobotics.com/ (visited on 06/23/2022).

Paramasivam, Naveenkumar (2019). *How to Create a Distance Measuring System using Arduino UNO R3*. URL: https://www.c-sharpcorner.com/article/how-to-create-distance-measuring-system-using-arduino-uno-r3/ (visited on 06/22/2022).

Peffers, Ken et al. (Jan. 2007). “A design science research methodology for information systems research”. In: *Journal of Management Information Systems* 24, pp. 45–77.

Prashar, Gaurav, Hitesh Vasudev, and Dharam Bhuddhi (2022). “Additive manufacturing: expanding 3D printing horizon in industry 4.0.” In: *International Journal on Interactive Design and Manufacturing (IJIDeM)*, pp. 1 –15. ISSN: 1955-2513.

Project HoverArm - Projects - ODrive Community (n.d.). [Online; accessed 2022-05-20].

Prusa i3 - Wikipedia (n.d.). [Online; accessed 2022-06-07].

Rakitin, Ivo and Valentina Ilieva Markova (2022). “Open Source Hardware - Advantages and Applications”. In: *2021 International Conference on Biomedical Innovations and Applications (BIA)*. Vol. 1, pp. 21–24.

Reduction Electric Sprocket Brushed Reductor (2022). URL: https://www.amazon.com/Reduction-Electric-Sprocket-Brushed-Reductor/dp/B0822QCGVZ (visited on 06/24/2022).

RoboteQ (2021). *RoboAMR 2021*. URL: https://www.roboteq.com/roboamr-2021 (visited on 07/12/2022).

Rodrigues, Tiago A. et al. (2019). “Current Status and Perspectives on Wire and Arc Additive Manufacturing (WAAM)”. In: *Materials* 12.7, p. 1121.

ROS Wiki (2022). URL: https://wiki.ros.org/ROS/Introduction (visited on 06/23/2022).

RPLidar A1M8 : MDROBOT (2020). URL: https://www.mdrobot.co.kr/Accessory/?idx=269 (visited on 06/22/2022).

Sabattini, Lorenzo et al. (2018). “The PAN-Robots Project: Advanced Automated Guided Vehicle Systems for Industrial Logistics”. In: *IEEE Robotics Automation Magazine* 25.1, pp. 55–64.

Santos, Santiago Roel (n.d.). *The Power of Open Source. I’ve become more drawn to learning. . . | by Santiago Roel Santos | Medium*. [Online; accessed 2022-05-20].

Savini, A. and G.G. Savini (2015). "A short history of 3D printing, a technological revolution just started". In: *2015 ICOHTEC/IEEE International History of High-Technologies and their Socio-Cultural Contexts Conference (HISTELCON)*, pp. 1–8.

Sensors, Special Issue : LiDAR for Autonomous Vehicles (2021). URL: https://www.mdpi.com/journal/sensors/special_issues/lidar_autonomous (visited on 06/22/2022).

Seybold, Jonathan et al. (2019). "Miniaturized Optical Encoder with Micro Structured Encoder Disc". In: *Applied Sciences* 9.3, p. 452.

Siegwart, Roland and Illah Reza Nourbakhsh (2004). *Introduction to Autonomous Mobile Robots.* Intelligent Robotics and Autonomous Agents. MIT Press. ISBN: 9780262195027.

Simon, Herbert A (1996). *The Sciences of the Artificial. 3rd*.

Sparkfun Electronics HC-SR04 (2022). URL: https://www.sparkfun.com/products/15569 (visited on 07/10/2022).

Stereolithography (SLA) 3D Printing Overview | Xometry Europe (n.d.). [Online; accessed 2022-06-07].

Taalbi, Josef (2017). "Origins and pathways of innovation in the third industrial revolution: Sweden,1950-2013". In: *Industrial and Corporate Change*.

Tao, Yubo, Qing Yin, and Peng Li (Dec. 2020). "An Additive Manufacturing Method Using Large-Scale Wood Inspired by Laminated Object Manufacturing and Plywood Technology". In: *Polymers* 13, p. 144.

Tornincasa, Stefano, Francesco Di Monaco, et al. (2010). "The future and the evolution of CAD". In: *Proceedings of the 14th international research/expert conference: trends in the development of machinery and associated technology*. Vol. 1. 1. Citeseer, pp. 11–18.

Torta, Stephanie and Jonathan Torta (2019). *3D Printing : An Introduction.* Mercury Learning Information. ISBN: 9781683922094.

Ullrich, Günter (2015). *Interdisciplinary Design of Automated Guided Vehicle Systems (AGVS).* Berlin, Heidelberg.

Vieira, Gustavo et al. (2020). "Low-Cost Industrial Controller based on the Raspberry Pi Platform." In: *2020 IEEE International Conference on Industrial Technology (ICIT), Industrial Technology (ICIT), 2020 IEEE International Conference on*, pp. 292 –297. ISSN: 978-1-7281-5754-2.

What Is FDM 3D Printing? [Quick and Simple Explanation] (n.d.). [Online; accessed 2022-06-07].

When Did 3D Modeling Start? A Brief History (n.d.). [Online; accessed 2022-06-03].

Winter, Jeff (n.d.). *What Is Industry 4.0?* [Online; accessed 2022-05-31].

Wiring Was Arduino Before Arduino | Hackaday (n.d.). [Online; accessed 2022-06-06].

Zhang, Lin, Peter Slaets, and Herman Bruyninckx (2012). "An open embedded hardware and software architecture applied to industrial robot control". In: *2012 IEEE International Conference on Mechatronics and Automation*, pp. 1822–1828.

Appendix A

Time plan of the project

In this annex, the time plan and its modification will be included for the reader's information. In the first Gantt diagram, an early estimation was made for the specification on the project, in which all the steps and operations were briefly described. Each activity was given a time span related to the complexity of the task.

TABLE A.1: Specification time plan estimation

Activity \Week	**5**	**7**	**9**	**11**	**13**	**15**	**17**	**19**	**21**
Start									
1. Previous design analysis and AGV State Of Art.									
2. AGV Structure construction. Chassis development.									
3 AGV Assembly									
4 AGV first testing (motors and sensors)									
5 AGV initial programming and testing									
6 AGV programming and overall functionality testing									
7 Documentation & report									
8 Preparation of presentation									

This estimation was very optimistic, and many changes were necessary. These changes were caused mainly because of material availability and problems encountered when trying to repurpose some parts. In the end, the prototype of the project was delivered as expected.

TABLE A.2: Real time-plan estimation

Activity \Week	**5**	**7**	**9**	**11**	**13**	**15**	**17**	**19**	**21**
Start									
1 Previous design analysis and AGV State of Art.									
2 AGV Structure construction. Chassis development									
3 AGV Assembly									
4 AGV first testing (motors and sensors)									
5 AGV initial programming and testing									
6 AGV programming and overall functionality testing									
7 Documentation & report									
8 Preparation of presentation									

Appendix B

Project quotation

The overall budget margin was very narrow for this project. Many parts have been repurposed and some others have been 3D printed which helped to maintain the overall price of the unit **in the range of 20000 SEK**, which was one of the objectives of this research project: delivering an affordable **AGV** solution to **SME**.

The parts which were sourced from the university and ASSAR have been added with actualized prices of authorized sellers such as RS components, Farnell or Maxon motor control. Labor estimation was done taking into consideration the price per hour of an experienced worker.

TABLE B.1: Project quotation table

Part	**Quantity (Pcs)**	**Price per unit (SEK)**	**Total (SEK)**	
20mm x 300mm round steel shaft	2	200	400	
6000mm 30mm x 50mm x 3mm steel tube	3	200	600	
6000mm 25mm x 25mm x 2mm steel tube	3	145	435	
4mm x 6mm aluminum rivets (50Pcs)	1	45.9	45.9	
1000mm x 500mm x 0.7mm sheet metal	2	189	378	
1000mm x 1000mm x 2mm sheet metal	1	300	300	
20mm ID pillow block bearing	4	173.96	695.84	
Hoverboard motor	2	250	500	
ABS 3D printed parts, per kg	1	200	200	
PCCF 3D printed parts, per kg	0.5	700	350	
NVIDIA Jetson Nano	1	3500	3500	
ARDUINO Mega	1	400	400	
M10 x 50mm bolts (5Pcs)	2	84.9	169.8	
M8 x 50mm bolts (10Pcs)	1	79.9	79.9	
M8 nuts (10Pcs)	1	64.9	64.9	
M10 nuts (10Pcs)	1	74.9	74.9	
Paint	1	200	200	
550mm x 15mm HTD5M belt	2	151.76	303.52	
150mm wheel	2	150	300	
Caster wheel	2	79.9	159.8	
MIG welding consumables (gas and wire)	1	170	170	
36V battery	1	4000	4000	
Motor controller DEC 50/5	2	2460	4920	
	Hours		**18247.56**	**TOTAL BOM**
Labor	20	270	5400	
			23647.56	**TOTAL**

Appendix C

Starting the AGV

C.1 Initial Steps

The starting sequence of the **AGV** will be described in this appendix for demonstration purposes.

The first step would be to remove the red and white lid of the **AGV** unit with the help of the chrome handle at the front and the gap in the back of the unit where the back tube of the subframe is revealed. Two people should perform this task.

Before being able to turn on the device, the battery which is situated in the front part of the main frame must be charged, as well as the battery of the PS4 controller, which would be the device that will be used to drive the unit.

At the moment of inserting the main battery on its place, the red switch connected to the battery holder must be set to OFF POSITION.

Now, **before changing the switch to ON POSITION**, the Arduino module that is located in the middle tube of the main frame **MUST BE DISCONNECTED**. After this, the back part of the unit where the powertrain wheels are situated must be elevated for safety concerns until they are not in contact with the ground in case that it starts to move for any reason. This could be done with the help of a wooden brick or similar that is placed between the ground and the back part of the main frame, lifting it **until the back wheels are not touching the ground**.

C.2 Initialisation

Touch screen, bluetooth keyboard and mouse will be needed for this part. In the transparent box of the project, a touch screen can be found. An HDMI and microusb cables will be needed as well, which are stored in the same box.The screen must be connected to the Jetson Nano module, which is situated at the back part of the main frame, in a square black box. The HDMI and USB ports are facing the ground, so the wooden brick that lifts the **AGV** would help perform this task.

When these parts have been found and connected, and the initial steps have been followed, it is time to turn on the device by **changing the switch to ON POSITION**.

The NVIDIA logo will show up in the screen and then the OS will start running. **If the system gets stuck in the NVIDIA logo, the AGV must be restarted by switching off and on the red switch**. If there is no problem, the Linux desktop will be shown on the screen. The system is ready to load the programs, but first, the mouse and keyboard must be connected. This will be done with the help of the touch screen, at first by reaching in the top right part of the screen the bluetooth icon. By pressing it, the connectivity window will be displayed, and then it will be necessary to pair the mouse keyboard and the PS4 controller in case it has been unpaired, which had never happened.

After all this, it is time to run the program.

C.3 PS4 Controller program

First of all, it will be necessary to turn on the PS4 controller by pressing the PS button in the middle of it. The controller light will flash white until it is connected. At that point, the white flashing will stop and remain still.

Then, with the help of the keyboard and mouse, it will be necessary to open the **Scripts** folder, located on the desktop. In the folder, right click inside the folder and open a terminal in that window. In the terminal, write the next command:

- *sudo bash ps4.sh*

A password will be required. The password is: jetson.

After this, **the arduino cable must be connected again**. The motors will start running because of the initialisation of the Arduino board module. When the system has started, they will stop working until one or both of the joysticks of the PS4 controllers is moved.

When the unit has been started and there are no red logs in the terminal, it is time to remove the wooden brick that lifted the **AGV** and disconnect the screen from the Jetson Nano, as it is no longer needed.

The unit is now ready for demonstration.

Appendix D

3D printed parts

Due to the fact that there are so many 3D printed parts in this design, and some of them made from different materials, here there will an explanation regarding how many of each parts are necessary and the possible materials that could be used.

Practically all of the parts are currently PLA, as it was the cheapest and most user friendly material available. Various modifications to different parts in CAD had to be made after observing the real-life tolerances between different parts, so to keep costs down these iterations were made using PLA, instead of more expensive and stronger materials such as PCCF. It has been proven that it is possible to make a working unit using only PLA parts, although it is not recommended to do so long-term due to the properties of this material, over time under constant pressure it suffers from deformation, known as creep.

Some parts would be resistant enough being fabricated from a material that does not suffer from creep, such as ABS, ASA or PETG as they only have to support compression and torque, and will not be subjected to high temperatures. Other parts will have to be able to resist high temperatures, due to the motors generating heat and friction from the belt; and should also have high dimensional accuracy, in the case of the pulleys the tooth profile should perfectly match the belt; due to this it is important that they be printed with a resistant material such as PCCF.

TABLE D.1: 3D printed parts

Part	Number	Weight (g)	Total Weight (g)	TPP* (h)	TT** (h)	Used Material	Recommended Materials
Motor pulley (D.1a)	2	54	108	2.3	4.6	PLA	PCCF
Shaft pulley (D.1b)	2	200	400	7	14	PLA	PCCF
Bearing block (D.2a)	4	105	420	3.75	15	PLA	PETG, ABS, ASA
Bearing sleeve (D.5a)	8	3	24	0.3	2.4	PLA	PETG, ABS, ASA
Tension block (D.2b)	4	20	80	0.9	3.6	PLA	PETG, ABS, ASA
Wheel hub (D.3a)	2	100	200	5	10	PLA	PETG, ABS, ASA
Motor holder top (D.4a)	2	17	34	0.8	1.6	PLA	PETG, ABS, ASA
Motor holder bottom (D.4b)	2	13	26	0.67	1.34	PLA	PETG, ABS, ASA
Ryobi battery holder (D.3b)	1	113	113	5.1	5.1	PLA	PETG, ABS, ASA
Alignment pin (D.5b)	6	1	6	0.08	0.48	PLA	PETG, ABS, ASA

Total printing time	**58.12**	**hours**
Total mass of plastic	**1411**	**g**

**TPP: Time per part.*

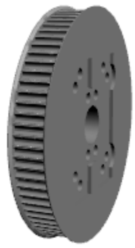
***TT: Total time.*

Most of these parts should be printed with 100% infill due to the forces and weight that they support, there are some exceptions, such as the battery holder and wheel hub, which with around 4 or 5 perimeters and 40% infill would be strong enough.

As seen in table D.1, the mass of filament that is required to print all of the parts is approximately 1.5kg, which is subject to variation depending on the exact printing parameters and material used, due to different densities.

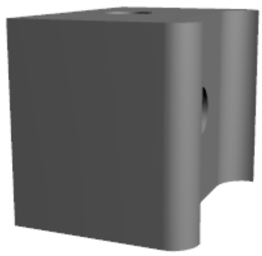
Furthermore, the time to print all the parts is approximately 58h, which can be reduced by around 50% by making multiple parts in a single print, though this increases the risk of failure, so it may not be a good idea when working with expensive filament.

(A) Motor pulley



(B) Shaft pulley

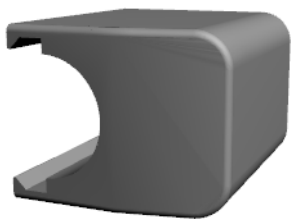
(A) Bearing block



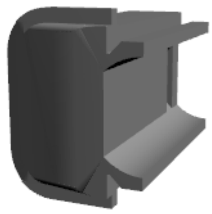
(B) Tension block

(A) Wheel hub

(B) Battery holder



(A) Motor holder top



(B) Motor holder bottom

(A) Bearing sleeve

(B) Alignment pin