Bachelor's thesis
Mini Guardian - Group 4

Daniel Solbrekke[1], Jennifer Marie Macintosh[2], Karine Christensen[3], Marius Heistad[1], Stefan Theodor Zurbuchen Hansen[1], and Veronica Juverud[2]

[1]Electrical Engineering
[2]Software Engineering
[3]Mechanical Engineering

May 22, 2023

Abstract

The bachelors project Mini Guardian aims to develop a scaled down model of the air defense system NASAMS, focusing on key functionalities and components. The project further aims to solve the given task to implement a Sensor for detection and categorization of airborne objects, such that an improved scale model of the Effector can autonomously target the objects deemed 'Hostile' and further be engaged by an operator. LiDAR and camera technology is utilized to have full overview of the surrounding environment. This proof of concept project explores the design, integration and functionality of a miniature air defense system. The objective is to showcase the capabilities of LiDAR and camera technologies, providing valuable insights into the operations and challenges of such complex systems.

This task was given by Integrated Defence Systems - a department in Kongsberg Defence & Aerospace.

Acknowledgments

This bachelors thesis is dedicated to Daniel Solbrekke, a true friend and colleague. Project Mini Guardian would not have been the same without you.

Our greatest gratitude goes to the summer project Coastal Shark, for lending us the essential LiDAR sensor used in this project, as well as contributing with other useful hardware.

We would like to thank Professor Sigmund Gudvangen, who has been our internal supervisor during this project. He has contributed thoroughly and in a sensible (fornuftig) fashion when it comes to reading the thesis as well as providing equipment to measure vibrations.

We are grateful to Project Engineer Vanja Halvorsen, our external supervisor, for placing her trust in us by assigning us this project. We are thankful to have had the opportunity to work on this project, and we sincerely appreciate the chance to contribute to its success.

A big thank you goes to Technical manager Olav Øverby at Kongsberg Terotech for his significant assistance in providing parts and other materials which was used in the making of the Power Supply.

We also want to express our gratitude to Lecturer Ingar Pedersen for his assistance in networking, we truly appreciate his time and support.

We would like to extend our gratitude to Pål Syversen and Ole Foss from Kongsberg Innovationcenter for granting us access to 3D printers, laser cutters, and necessary materials.

Furthermore Toolmaker Anders Hovden at KDA DMA deserves a great thank you for machining parts vital to the project.

We extend our sincere appreciation to Project Engineer Lina Theimann and Project Engineer Ole Christian Walvatne Kopland for their assistance in providing guidance and answering our questions regarding LiDAR, Qt, and the overall functionality of the previous year's project [1].

A big thank you to Associate professor Amir Safari, for helping with the Finite Element Analysis (FEA) and calculations.

Contents

List of Tables

List of Figures

Acronyms

ABS Acrylonitrile Butadiene Styrene. ix, lxvii–lxx, lxxii–lxxiv, 164

AHP Analytic Hierarchy Process. viii, x, 37, 43–45, 48–52

AMRAAM Advanced Medium-Range Air-to-Air Missile. 155, 157

AP Average Precision. 70

API Application Programming Interface. 59, 66, 68

AWG American Wire Gauge. 109

BUE Built-Up Edge. 141

CCW Counterclockwise. 99, 186, 190

CG Center of Gravity. 155, 156, 164

CI Consistency Index. 48

CNC Computer Numerical Control. 152, 153, 155

CP Center of Pressure. 155, 156

CPU Central Processing Unit. 89

CSS Cascading Style Sheets. 72

CW Clockwise. 99, 186, 190

DMA Aerostructures. 140

DNN Deep Neural Network. 71

FDC Fire Distribution Center. 32, 34, 79, 90, 109, 113, 199

FDM Fused Deposition Modeling. ix, lxviii, lxix, 157, 193

FEA Finite Element Analysis. ii, xii, 22, 118, 124, 154, 155, 162–165, 167, 169, 172–174, 177

FoV Field of View. x, xii, 37, 39, 41, 42, 46, 47, 50–53, 64, 158–160

FPS Frames Per Second. 40, 41

GPU Graphics Processing Unit. 69

GUI Graphical User Interface. viii, x, xi, 1, 2, 16–19, 27, 32–35, 55, 58, 65, 72–74, 76–84, 87–89, 99, 107, 108, 186, 189–191

HAZ Heat-Affected Zone. 149, 150

HMI Human-Machine Interface. 18

HTTP Hypertext Transfer Protocol. 79

HUF Hostile, Unknown & Friendly. 18

ID Inner Diameter. 137, 144, 155

IDS Integrated Defence Systems. 1, 6, 51

IEPE Integrated Electronics Piezo-Electric. 119, 120

IP Internet Protocol. viii, 81, 82, 84

KDA Kongsberg Defence & Aerospace. 1, 2, 6, 33, 40, 41, 51, 140

KIC Kongsberg Innovation Center. 106

LiDAR Light Detection And Ranging. i, ii, viii, x, xii, 3, 4, 6, 14, 19, 37–42, 46, 47, 49–53, 55, 58, 64–66, 76, 77, 79, 81, 83–86, 88, 89, 158–162, 199

mAP mean Average Precision. 68–70

MIDI Musical Instrument Digital Interface. 120

MVP Minimum Viable Product. 18, 19

N/A Not Applicable. 16

NASAMS Norwegian/National Advanced Surface-to-Air Missile System. i, 2, 3, 32, 33, 37, 79, 155, 199

NC Normally Closed. 97, 98

NO Normally Open. 97, 98

OD Outer Diameter. 137, 144, 155

OS Operating System. 88, 89

PCB Printed Circuit Board. viii, 101, 103, 106, 111, 112, 198

PE Protective Earth. 107, 115

PLA Polylactic acid. ix, lxvii–lxx, lxxii, lxxiii, lxxv, 128, 129, 140, 155, 157, 164

PoE Power over Ethernet. 113, 115

PS Power Supply. ii, xi, 90, 100, 102, 103, 110, 113, 114, 193, 199

PTZ Pan-Tilt-Zoom. 54

RADAR Radio Detection And Ranging. viii, x, 3, 38–40, 47, 49

RAM Random Access Memory. 89

ROS Robot Operating System. 19, 57, 64–66, 76, 79, 83, 84, 88, 89

RTSP Real Time Streaming Protocol. 79

SLS Selective Laser Sintering. lxviii, 157

SSD Single Shot Multibox Detector. 68

SSD Solid State Drive. 89

TTL Transistor-Transistor Logic. 83

UART (Universal Asynchronous Receiver-Transmitter. 83

UAVs Unmanned Aerial Vehicles. 2

UDP User Datagram Protocol. 76, 82–84

UI User Interface. 59, 73

URL Universal Resource Locator. 79

USB Universal Serial Bus. 34, 120

USN University of South-Eastern Norway. 6, 101, 105, 106, 196

VAT Value Added Tax. 50

VCS Version Control System. 89

VLC VideoLAN Client. 79

VM Virtual Machine. 88

WDR Wide Dynamic Range. 54

WIP Work In Progress. 9, 10

WPS Wi-Fi based Positioning System. 40

YOLO You Only Look Once. 57, 66–69, 72

Glossary

Effector The Systems part that launches the projectiles. i, x–xii, xvii, lxxxv, xc, 2, 3, 12, 13, 16–20, 27, 32–35, 37, 41, 46, 50, 51, 55, 64, 65, 74, 76, 77, 81–86, 90, 92–99, 103, 107, 109–111, 117, 118, 121–124, 131, 155, 158–162, 177, 178, 189, 190, 193, 194, 198, 199

Enclosure The tower stand where the Sensor and Effector are mounted. x, xii, 55, 56, 64, 157, 160–162

Friendly Friendly is defined as a green balloon. 1, 3, 27, 55, 72, 77, 199

Hostile Hostile is defined as a red balloon. i, 1–3, 55, 57, 65, 72, 73, 77, 199

Sensor The Systems sensor, the affiliated code and the operator. i, x, xii, xvii, 1–4, 12, 16–19, 33, 36, 37, 45, 46, 49, 51, 160, 161, 193

System The collection of Effector and Sensor. viii, x, xi, 1, 2, 16, 18, 19, 32, 36–38, 46, 53, 55, 56, 90, 99, 100, 102, 106–110, 113, 124, 197–199

Unknown Unknown is defined as a balloon that is not red or green. 1, 3, 27, 55, 72, 77, 199

Chapter 1

Introduction

1.1 The problem

SH

Mini Guardian is a bachelor thesis project given by Integrated Defence Systems (IDS) - a department in Kongsberg Defence & Aerospace (KDA) - in the spring of 2023. The thesis will build on Project Mini Launcher, a bachelor project from 2022 [1]. This years mission is to develop and implement an advanced Sensor capable of detecting and identifying objects, whilst seamlessly transmitting the information to an intuitive Graphical User Interface (GUI). Through this GUI an operator can make informed decisions and determine further action. The Systems functionality is shown in fig. 1.1. It is to detect and identify objects, here shown as balloons. After an object is identified as Hostile, the operator can engage the target by firing a projectile. A Friendly or Unknown target should not be able to be fired upon and will remain protected.

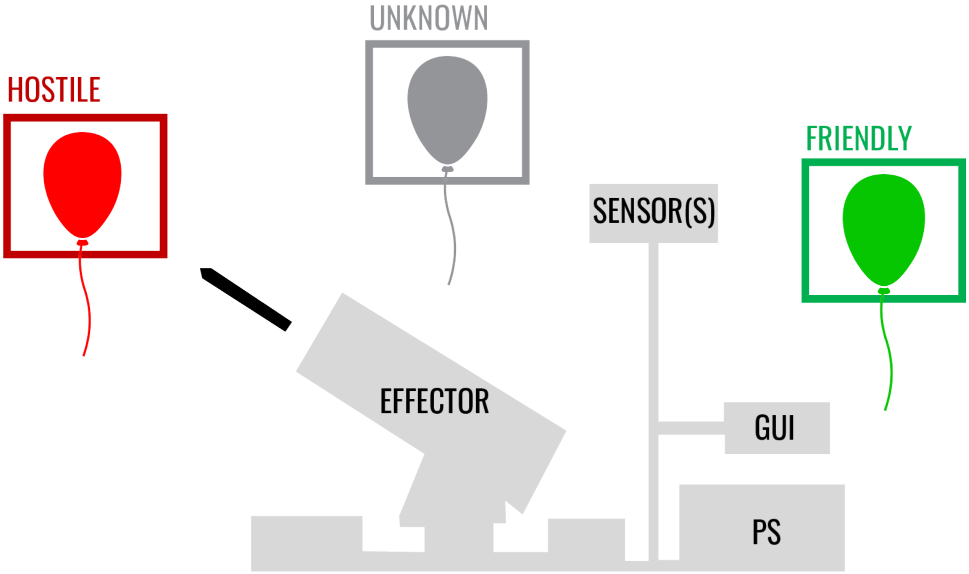


Figure 1.1: Block diagram for the System.

As the project is to be considered as an unclassified project, it must be conducted with commercially available components and 3D-printing. That means that Mini Guardian can not rely on the military products that are developed by KDA.

The project can be summarized into two bullet points as these are the main parts of the project:

- Improve Project Mini Launcher, the previous bachelor thesis project that Mini Guardian builds upon.
- Develop and implement a Sensor for object detection.

The projects problem is developed and agreed upon by both the client KDA and Mini Guardian.

The problem comes with a set of requirements, which have been revised before and through the project in collaboration with the client. The requirements are tabulated in Table 3.3 for functional requirements, and Table 3.4 for the non-functional requirements. To introduce them the next three paragraphs pick out the essence of the requirements.

The Systems Sensor shall detect, identify and categorize objects. Objectives categorized as Hostile shall make the System aim the Effector to the targeted object, such that an operator can decide how to resolve the situation through a GUI. If decided to engage the target, the Effector shall hit the object marked as Hostile.

The GUI shall display various information, such as the Systems operational status, the Effectors direction and orientation, the identified objects, a launch log and should display previous target assessments.

The non-functional requirements of this project is that the System shall be developed with commercially available components, there should not be a risk of health and safety, and the interface shall be documented, in similar fashion there shall be a technical user guide.

1.2 NASAMS

SH

The Norwegian/National Advanced Surface-to-Air Missile System (NASAMS) is an advanced air defense system designed to protect critical assets against airborne threats. Developed jointly by the Norwegian company Kongsberg Defence & Aerospace and the American defense contractor Raytheon, NASAMS is highly regarded for its exceptional capabilities in detecting, tracking and engaging various aerial targets. It is a versatile system that can be deployed in both mobile and stationary configurations, offering a flexible and scalable solution for air defense requirements. With its cutting-edge technology and robust architecture, NASAMS has been widely adopted by numerous countries around the world, providing enhanced protection for military bases, key infrastructure, high-value assets and urban areas. Its sophisticated radar systems, integrated command and control functions and an array of interceptors make NASAMS a formidable defense system against a range of threats, including aircraft, Unmanned Aerial Vehicles (UAVs) and cruise missiles [6].

1.3 Project Mini Launcher

SH

Project Mini Launcher is the previous bachelor project that laid the foundation for Mini Guardian. The project was to develop a 1:12.5 scale model version of the NASAMS. The model had three main functions: rotation, elevation and firing.

The firing mechanism uses compressed air, distributed through electrically controlled valves. The elevation mechanism utilizes a stepper motor and a worm gear drive, and the rotation mechanism employs a stepper motor and a drive belt. The system is controlled by an Arduino Mega 2560, communicating with an Arduino Nano in the top structure for firing control and projectile load information [1].

1.4 Project Mini Guardian

SH

Introducing project Mini Guardian, an endeavor to develop a scaled down model of the successful air defense system NASAMS. This bachelor project aims to further develop the 1:12.5 scaled model that is inspired by NASAMS's key functionalities and components. Notably, the project includes the implementation of a Light Detection And Ranging (LiDAR) sensor and an AXIS camera known in combination as the Sensor as crucial elements of the air defense system.

In comparison to NASAMS, the Mini Guardian model represents a reduced-scale version of the original system. While NASAMS serves as a sophisticated air defense solution deployed by various armed forces globally, Mini Guardian replicates its core functionalities and features in a more compact form. The Mini Guardian project team has chosen a LiDAR sensor and an AXIS camera to mimic the sensing capabilities of the NASAMS.

When comparing the sensors used in both systems, NASAMS employs advanced Radio Detection And Ranging (RADAR) technology for target detection, tracking, and identification. On the other hand Mini Guardian utilizes a LiDAR sensor, which employs laser-based technology to detect objects in its surroundings and provides measuring information to range a target. Additionally, Mini Guardian incorporates an AXIS camera to provide visual information for target identification, categorizing them as either Hostile, Unknown or Friendly based on HUF symbology from MIL-STD-1477C [7]. NASAMS too has an incorporated camera which can be controlled by an operator [8]. It is among others used to confirm target termination.

While the Mini Guardian project aims to replicate the essential sensing aspects of the NASAMS through its LiDAR sensor and AXIS camera, it is important to note that the scaled-down model won't achieve anywhere near the capabilities of the NASAMS. One of the biggest challenges with this project is the clear distinguishing factor of the projectiles. Whilst NASAMS uses sophisticated missiles which use guidance in its path to the target, our projectiles are dumb and will only follow the projectiles ballistic path fired out of the Effectors canisters. However, the project presents an exciting opportunity to explore and develop a functional air defense system on a smaller scale, offering valuable insights into the operations and challenges of such complex technologies.

Through project Mini Guardian, the team seeks to showcase the scaled-down air defense system and gain practical experience in the design, implementation, and integration of key components. By using the Sensor system employed in this project, Mini Guardian offers valuable insights into the capabilities of LiDARs and camera technologies to develop a proof of concept that can be the foundation of a functional miniature air defense system.

Chapter 2

Project management

2.1 Group members

Mini Guardian is a multidisciplinary bachelor group consisting of six students, where two are studying software, three electronics and finally one mechanical student. The members are currently students at University of South-Eastern Norway, campus Kongsberg. The members are introduced below.

Daniel Solbrekke - *Document Manager & Chief of Control Systems* - has experience from the process industry after a fulfilled apprenticeship as an automation technician with Norske Skog Saugbrugs, a paper mill in Halden. He served 15 months in the Royal Norwegian Air Force as a weapons system assistant on F-16 at Bodø main air station. Additional eight years as a freelance audio engineer. He is now an electronics & cybernetics engineering student at University of South-Eastern Norway.

Jennifer Marie Macintosh - *Finance and Website Manager & Chief of Object Identification* - is studying Cyber Physical Systems at University of South-Eastern Norway. Born in Gothenburg where she later worked with physical therapy as a self-employed. Moved to Norway in 2018, and worked as a sales advisor before starting her studies. She has a keen interest in technology and is very eager to learn how things work.

Karine Christensen - *Requirement Manager & Chief of Mechanical Design* - is from a small village on the west-coast of Norway called Fitjar. She ended up in Kongsberg by coincidence in 2016. In 2020 the studies to become a mechanical engineer started, and this is something she really enjoys. The best part of the Product development studies is how interesting and challenging the topics are, and how they teach you to see the world from a different perspective. Karine has earlier experience from graphic design and retail working, alongside horseback riding. In the most recent years, she has gained experience from a part-time job as a design assistant at Kongsberg Ferrotech and a summer internship at KDA's summer project "Coastal Shark".

Marius Heistad - *Test and Risk Assessment Manager & Chief of Energy Flow* - was born and raised on Kongsberg, in 2015 he started at K-tech as an automation technician apprentice. Got the certificate of apprenticeship as an automation technician in 2016 and begun to work at Kongsberg Terotech in 2017 as a service technician. He has served 12 months of service in the Royal Norwegian Navy as a marine engineer assistant on the His Norwegian Majesty's Ship (HNoMS) Otra. Studies to become an Electrical Engineer at the University of South-Eastern Norway campus Kongsberg.

Stefan Theodor Zurbuchen Hansen - *Project leader & Chief of Sensors* - has previously worked as an electrician at Bravida AS after a fulfilled apprenticeship. He has experience from the Royal Norwegian Air Force, 15 months service at Bodø Main Air Station as a weapons systems assistant on F-16's. The summer of 2022 he had an internship at KDA - department IDS - where he worked on the project Coastal Shark and got into contact with the client of this project. Some relevant knowledge about LiDAR and other sensors was also acquired here. Currently, he is an electronics engineering student specializing in cybernetics at the University of South-Eastern Norway.

Veronica Juverud - *Communications Manager & Chief of Object Detection* - is currently studying Cyber Physical Systems at University of South-Eastern Norway, campus Kongsberg. She has worked as a substitute during the summer at Sparebanken Øst's IT department in Drammen, and participated in projects while working towards employees with support and other technical assistance. She is working part time as a sales advisor alongside the studies.

2.2 Project models

KC

The group decided at an early stage to adopt an agile project model to facilitate efficient development and testing for our project. The flexibility and ability to navigate between iterations quickly and cost-effectively were crucial factors in this decision. New ideas throughout the process should always be welcomed and we need to be able to explore every possibility and idea, even later in the process. Embracing an agile approach allows for such exploration and new ideas throughout the project, also at later stages. Consequently, we explored various agile models, such as Scrum and Kanban, as well as other alternatives to determine the most suitable approach for our project.

2.2.1 Different project models

KC | MH

Unified process is a project management model that combines the best commonly accepted practices into a risk-driven and iterative project model. The development in the project is organized into what is essentially short mini projects called iterations that has a given time frame to be completed within. The product of each iteration is a well-documented and tested system, which can be totally or partially integrated into an existing system. Each iteration cycle has its own analysis of requirements, a design phase, an implementation phase, and testing/validation phase. Unified Process relies on successive enlargement and refinement of a system which is achieved trough multiple iterations during the projects life-cycle. Furthermore, it relies on cyclic feedback after each iteration as well as adaptation [9]. This model creates a good system over its iterations but its heavy on the documentation, as a result of this the group did not see Unified Process as a good project model to implement into our project.

The Waterfall model is a model based on specific steps throughout the project: “Start-up”, “Planning”, “Requirements”, “Design”, “Build”, “Test” and “Implement”. It is a quite rigid model, meaning that until you finish one stage, the next one cannot start. This also means that implementing changes will be difficult and expensive, since the “Waterfall” is not supposed to “flow upwards”. For this model to be successful, it is essential to gather as much information about the project as possible. This includes the costumer needs, how to perform tests and maintenance and who the stakeholders are and what they need [10]. Since our project is dependent on rapid changes and iterations, this is not a well-suited project model and will not be looked further into.

The Spiral model is a risk-driven project model designed as a spiral with an undetermined number of spirals. Each spiral consists of four main activities: “Determine objectives, alternatives and constraints”, “Evaluate alternatives, identify and resolve risks”, “Develop and verify” and “Plan next phase”. These activities, or spirals, are repeated as often as needed. This makes the model suited for projects that are dependent on change. Prototyping is essential for defining stakeholder requirements, since early produced prototypes gives the stakeholder a taste of how the product will turn out [11]. Even though this model is open for as many iterations as needed, it is also heavier on the documentation than other models we investigated. This is because every spiral has to be documented properly with for example new risk evaluations. Because of this, we don't see this model as the best suit for our project.

Scrum is a lightweight model designed to help solve complex problems or tasks in the early 1990s. This model depends on five very important values (the Scrum values): commitment, focus, openness, respect and courage. Every team member commits to these values for achieving the common goal and being a good team-player [12].

The Scrum team is small (containing a maximum of 10 persons) and consists of a Scrum master, a product owner and developers. The Scrum master should be the person in the group that has the best knowledge about the project model, since this person is supposed to help the other team members understand and work with the principles of Scrum. The product owner is one person who is accountable for the product result and product backlog management, amongst other things. The developers consist of the rest of the Scrum team. They are responsible for planning the sprints and adapting their plan towards the common sprint goal [13].

During the project lifespan, several Scrum events will occur. In the time of the project, its products will go through several sprints, the list is inspired by the Scrum guide [12]:

- First there is the sprint planning. This is where the following sprint is planned, through discussion of three topics: Why is this sprint valuable? What can be done this sprint? How will the chosen work get done? These sessions cannot take longer than eight hours for a sprint that is lasting for one month. Usually, the session is shorter.
- Second comes the sprint itself. This is where the ideas turn in to products and goals are reached. The sprint cannot last for more than one month, and is usually shorter to keep the team on their toes. No changes are made during sprints, especially not if they can endanger the goal of the sprint. New ideas and changes usually wait for the next iteration/sprint.
- After the sprint is completed, there will be held a sprint review meeting lasting a maximum of four hours for the longest sprints. In this meeting, the discussion will be about the results from the previous sprint and how to take the results further. The team also presents their results to the key stakeholders.
- When the sprint review is complete, the sprint retrospective starts. This is a process of maximum three hours where the purpose is to plan ways to increase effectiveness and quality. The discussion revolves around what went well, problems that occurred and how or if the team solved them.
- In addition to the listed events, there will be a daily 15 minute Scrum meeting, where the developers discusses plans and activities that lay ahead. This is designed to improve communication, quick decision-making and to early spot problems that occurs during the sprint. The daily Scrum will also minimize the need for other meetings.

After getting to know this project model using the Scrum guide [12] written by the brains behind the framework, we have some thoughts about how this model will fit with our project and group:

- The daily Scrum-meetings is a good idea, since our group consists of multiple disciplines. This will keep every member up to date on what the other members are working on. These meetings will help the group understand all members contributions and help each other with difficult tasks, surprises or problems.
- The part where Scrum is dividing the group into a Scrum master, project owner and developers is not something we think is a good fit for us. This is because we want every member to be equal with the same responsibilities.
- The sprint lasting a maximum of one month is something we really see the value of. This will help dividing the project into smaller projects and making the goal more reachable. Although, we would like to be able to make changes during the sprint, if we encounter something that needs changing.

Kanban is a Japanese project model developed at Toyota in the 1940s. This model is, according to Kanban trainer Grady Brumbaugh, something all teams and projects can use. It is supposed to be document-light and simple to learn and understand. In the Kanban framework, there are no defined roles like in Scrum, but there are six core principles: visualize your work, limit the work in progress, manage flow, make policies explicit, implement feedback loops, improve collaboratively and evolve experimentally [14].

There are several essential features to the Kanban project model, some of whom are shown in this list inspired by the complete Kanban project guide [14]:

- Visualizing your work is probably what Kanban is most known for. Most people recognize the famous “Kanban-board”, where tasks are sorted into “to do”, “doing” and “done” and categorized based on urgency. This board is the absolute centerpiece of this framework. There can also be added for example “plan” and “test” to the board if needed. The meaning of the Kanban-board is to give every team member an impression of how the workflow is going: everyone can see when a new task is moved from “to do” to “doing”. This board is unlike the similar Scrum-board, very agile and you can move or change tasks as much as needed.
- Limiting Work In Progress (WIP) is done by implementing an allowed number of tasks for each member to work on simultaneously, forcing prioritizing of the different tasks. This is often hard to do, but according to Kanban it will increase the efficiency of the work.

There are also a few methods to help find and get rid of inefficiencies. The most common are:

- Cycle time: tracking how long it takes for a task to travel through the Kanban-board.
- Due date performance: track how the task is evolving according to schedule.

As we can see, there are a lot of positive sides about the Kanban method. The visualization and agility of the Kanban-board is one to notice. This seems to give every team member the opportunity to stay updated on the workflow without having to ask or arrange meetings. Although, we also think this project-model lacks a bit of structure.

Summary: Given the iterative nature of our project, which involves frequent changes and extensive testing, we have decided not to pursue the Waterfall model any further. Additionally, we have collectively agreed to avoid models that involve excessive documentation, such as the Unified Process or the Spiral model. This decision was made by statements from the experiences of the previous bachelor group - Project Mini Launcher.

2.3 Scrumban

KC | MH

After learning all this, the decision is still tough. There are different aspects of both Scrum and Kanban that will suit our project well, but there are also parts that are less suited in both frameworks. Because of this, we have decided to go with the best of both worlds and chose Scrumban. This is a hybrid project model that combines the structure from Scrum with the visualization and agility from Kanban. By doing this, we can work with what we were most attracted to within both models.

We will use the daily Scrum meetings and integrate a weekly update-meeting where subjects from the sprint planning, sprint review and the sprint retrospective will be discussed. This means that the delivery cycles will be continuous, like in Kanban, but we will meet up, like Scrum, at the end of a sprint. Since the sprints will be less defined than in Scrum, it will be easier for the group to implement changes and new ideas, like in Kanban. There will be no Scrum master or project owner. For visualization, the Kanban board will be a key part during this project. The Kanban-board will consist of several post-it notes representing tasks that needs to be completed. Our board is divided into the categories “Chores”, “Planning”, “Process”, “Testing” and “Completed” and we have chosen the WIP to be a maximum of two tasks per person simultaneously. The group has identified iterations in two different ways, regarding the form of the task: if the task is purely administrative documentation, one iteration will be one revision. This means that there will be different revisions of documents like requirements, risk analysis and so forth. Tasks including research and more specific technical work has its iterations within the Kanban-board. This means that the tasks iteration will be the life of the post-it note on the board. The note can be moved back and forth between the different board-categories, iterating again every time it is moved. Also, we decided to replace Scrums "Push" with Kanbans "Pull". This means that instead of assigning tasks to each member, each member chooses their own tasks from the board.

A digital version of the Scrumban board was also created using a Miro board [15], the intent was to update the board digitally to showcase the progress trough the project. An example is shown in fig. 2.1. Some of the digital versions can be found in Appendix G, and Appendix F.

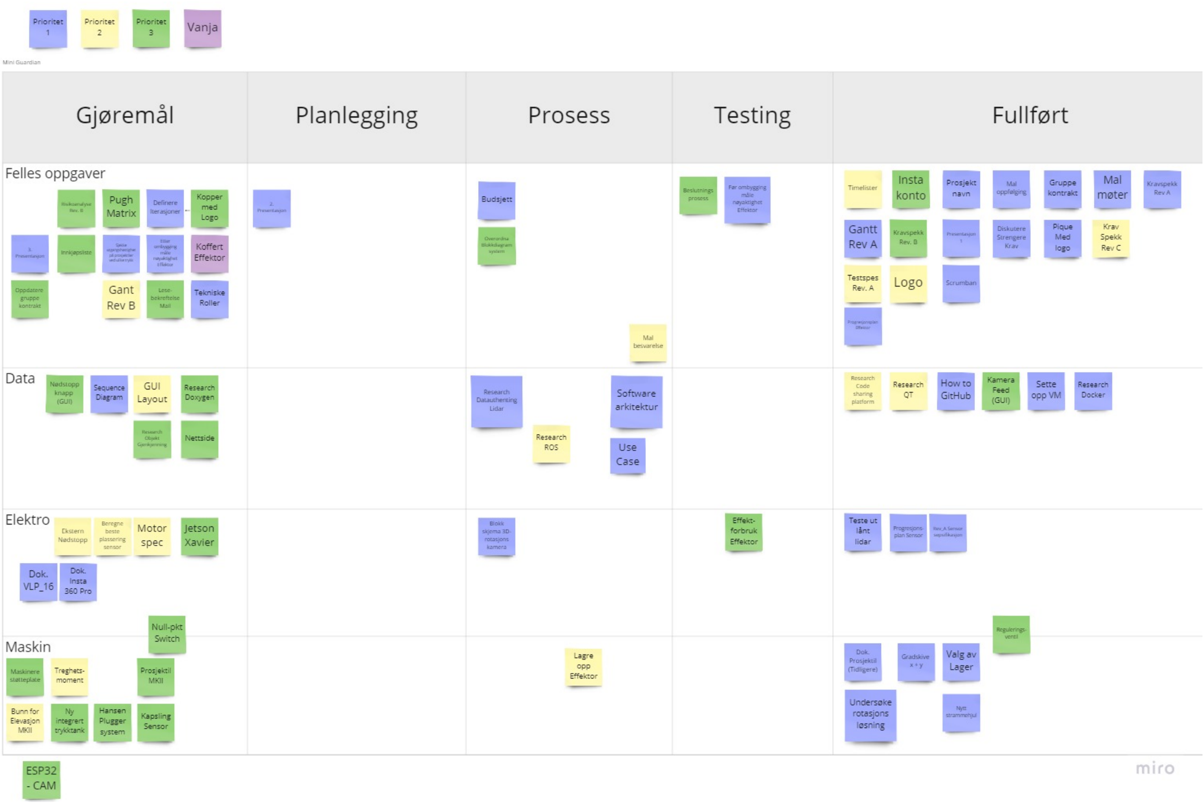


Figure 2.1: Scrumban board week 10

The physical Scrumban board per 05/21/2023 is displayed in fig. 2.2. Blue post-it notes contains the tasks of the highest priority, yellow contains the second highest priority while green contains the least prioritized tasks.

Figure 2.2: Scrumban board per 05/21/2023.

2.4 Timeline

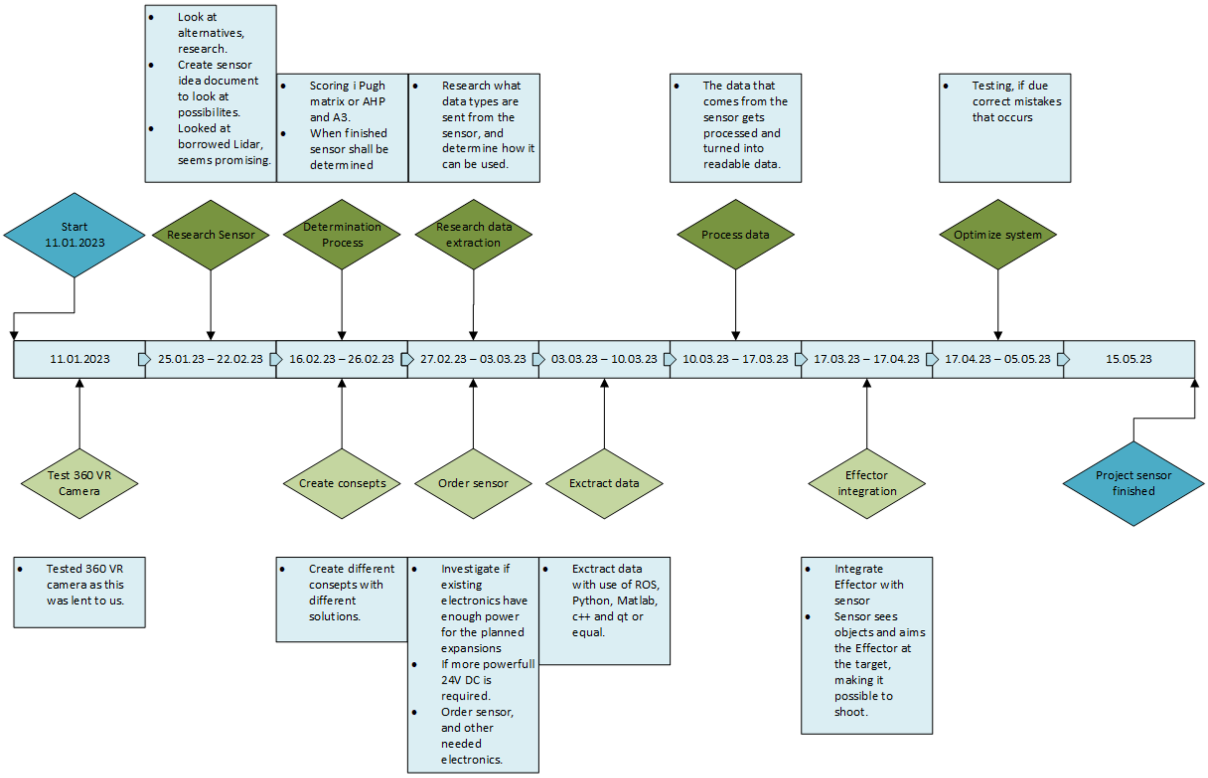
SH | MG

The timeline for project Mini Guardian was originally set up in a Gantt chart. The chart can be seen in fig. 2.5. A previous iteration of the chart is shown in Appendix H. Gantt charts is a tool for visualizing and managing project schedules. A Gantt chart provides a graphical representation of project tasks, durations, dependencies, and milestones over time. It allows the project to plan, track, and coordinate various activities involved in the project effectively.

One of the benefits of using Gantt charts is the ability to provide a clear overview of the project timeline. By breaking down the project into specific tasks and assigning them durations, it can easily identify the sequence of activities and their dependencies. This helps to understand how different tasks relate to each other and visualize the overall project progression.

There were also made two separate progression plans. One for the Sensor and one for the rotational base on the Effector. These can respectively be seen in fig. 2.3 and fig. 2.4. This was to further visualize expected progression and also to provide an open platform for communication and collaboration within the project. By sharing the plan with team members and stakeholders, everyone gains a common understanding of the project timeline and objectives. This promotes coordination, enables better task prioritization and effective teamwork.

It also provides a basis for tracking progress and monitoring project performance. As the progression plan consists of start and end dates, the team can compare them against the planned schedule, identifying deviations or delays. This allows for future decision-making, enabling the team members to take corrective actions to bring the project back on track.



- Look at alternatives, research.
- Create sensor idea document to look at possibilites.
- Looked at borrowed Lidar, seems promising.

- Scoring i Pugh matrix or AHP and A3.
- When finished sensor shall be determined

- Research what data types are sent from the sensor, and determine how it can be used.

- The data that comes from the sensor gets processed and turned into readable data.

- Testing, if due correct mistakes that occurs

- Tested 360 VR camera as this was lent to us.

- Create different consepts with different solutions.

- Investigate if existing electronics have enough power for the planned expansions
- If more powerfull 24V DC is required.
- Order sensor, and other needed electronics.

- Exctract data with use of ROS, Python, Matlab, c++ and qt or equal.

- Integrate Effector with sensor
- Sensor sees objects and aims the Effector at the target, making it possible to shoot.

Figure 2.3: Progression plan for the Sensor.

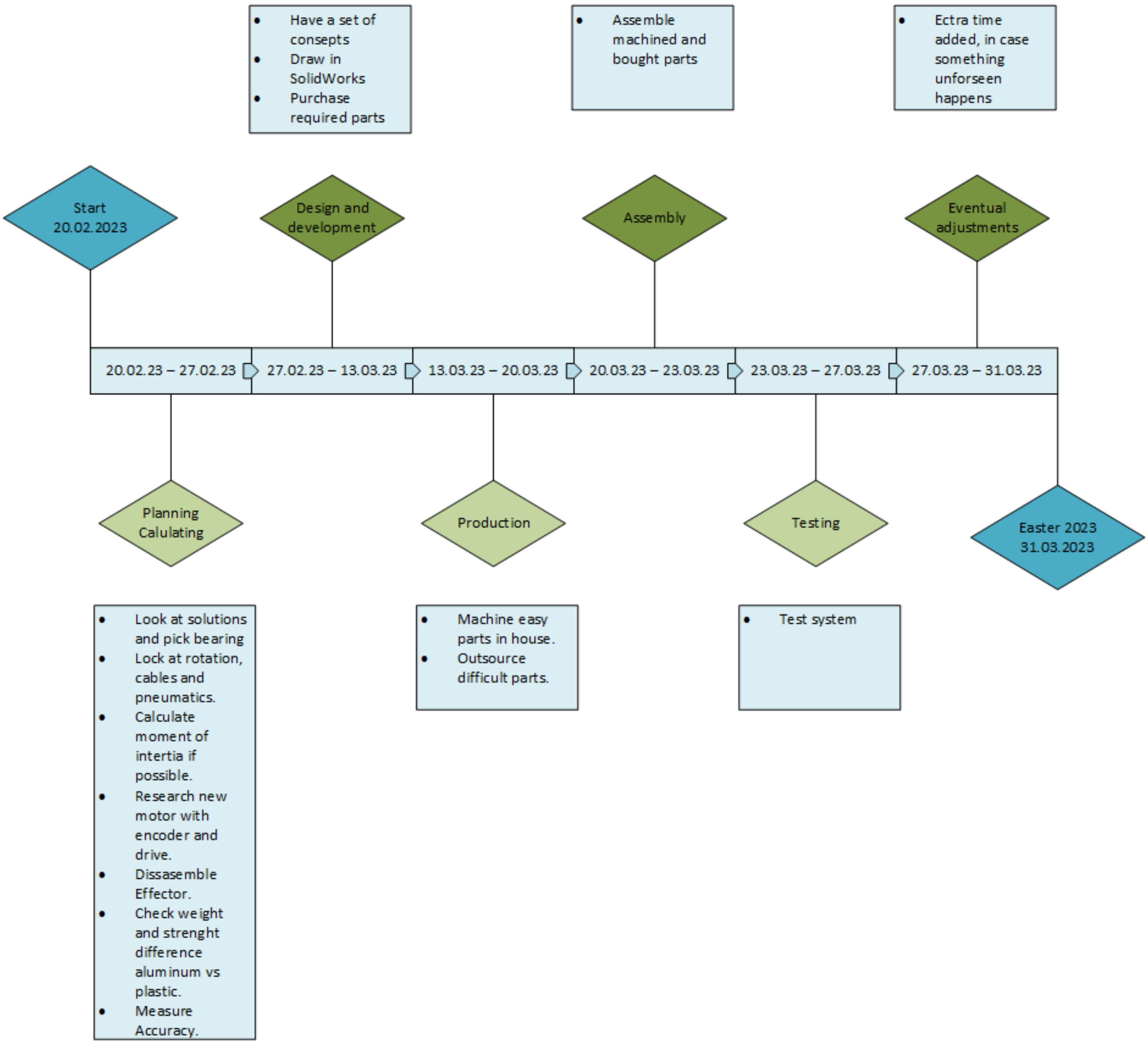
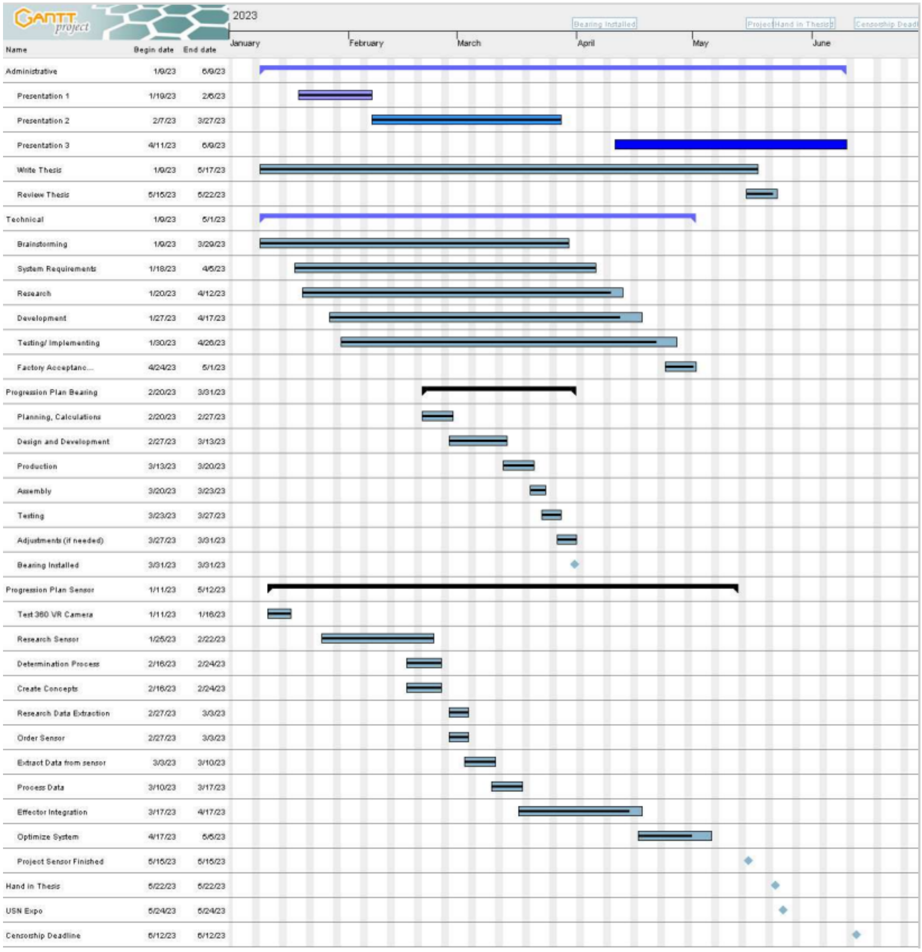

Have a set of consepts
Draw in SolidWorks
Purchase required parts
Assemble machined and bought parts
Ectra time added, in case something unforseen happens
Start 20.02.2023
Design and development
Assembly
Eventual adjustments
20.02.23 – 27.02.23
27.02.23 – 13.03.23
13.03.23 – 20.03.23
20.03.23 – 23.03.23
23.03.23 – 27.03.23
27.03.23 – 31.03.23
Planning Calulating
Production
Testing
Easter 2023 31.03.2023
Look at solutions and pick bearing
Lock at rotation, cables and pneumatics.
Calculate moment of intertia if possible.
Research new motor with encoder and drive.
Dissasemble Effector.
Check weight and strenght difference aluminum vs plastic.
Measure Accuracy.
Machine easy parts in house.
Outsource difficult parts.
Test system

Figure 2.4: Progression plan for the Effector.

In summary, timelines are an invaluable tool for project Mini Guardian. They offer a visual representation of project tasks, dependencies, and timelines, helping to plan, coordinate and monitor the project effectively. By utilizing Gantt charts and other timelines, one can enhance project management, improve communication, identify critical paths, and track progress towards successful completion of the project.



Name	Begin date	End date
Administrative	1/9/23	6/9/23
Presentation 1	1/19/23	2/6/23
Presentation 2	2/7/23	3/27/23
Presentation 3	4/11/23	6/9/23
Write Thesis	1/9/23	5/17/23
Review Thesis	5/16/23	5/22/23
Technical	1/9/23	5/1/23
Brainstorming	1/9/23	3/29/23
System Requirements	1/18/23	4/5/23
Research	1/20/23	4/12/23
Development	1/27/23	4/17/23
Testing/ Implementing	1/30/23	4/26/23
Factory Acceptanc...	4/24/23	5/1/23
Progression Plan Bearing	2/20/23	3/31/23
Planning, Calculations	2/20/23	2/27/23
Design and Development	2/27/23	3/13/23
Production	3/13/23	3/20/23
Assembly	3/20/23	3/23/23
Testing	3/23/23	3/27/23
Adjustments (if needed)	3/27/23	3/31/23
Bearing Installed	3/31/23	3/31/23
Progression Plan Sensor	1/11/23	5/12/23
Test 360 VR Camera	1/11/23	1/16/23
Research Sensor	1/25/23	2/22/23
Determination Process	2/16/23	2/24/23
Create Concepts	2/16/23	2/24/23
Research Data Extraction	2/27/23	3/3/23
Order Sensor	2/27/23	3/3/23
Extract Data from sensor	3/3/23	3/10/23
Process Data	3/10/23	3/17/23
Effector Integration	3/17/23	4/17/23
Optimize System	4/17/23	5/5/23
Project Sensor Finished	5/15/23	5/15/23
Hand in Thesis	5/22/23	5/22/23
USN Expo	5/24/23	5/24/23
Censorship Deadline	6/12/23	6/12/23

Figure 2.5: Gantt chart for project Mini Guardian.

The original timeline for the project proved to be overly optimistic, resulting in a significant shift of some tasks of over a month. Several factors contributed to this delay, highlighting the importance of realistic planning and thorough consideration of potential challenges. One major setback was encountered during the process of potentially purchasing a LiDAR, which turned out to be more complex and time-consuming than initially anticipated. This unexpected difficulty in acquiring the necessary hardware significantly impacted the projects progress. The unforeseen circumstances emphasizes the importance of allocating sufficient time for critical project components. By learning from these experiences, future project planning can be enhanced, ensuring more realistic timelines that account for potential obstacles and allow for better overall project management. The client has proposed to buy a LiDAR for next years project and have it ready from the project start which is sensible.

Taking into account the unforeseen delays encountered during the initial timeline, we implemented a more realistic and proactive timeline for the remaining stages of the project, prioritizing and focusing on the tasks most important for the project. Addressing the potential technical issues and challenges that were left, they were incorporated into a new adjusted timeline to make sure the project was completed within the defined timeframe.

Chapter 3

Requirements

MH

A common way of placing requirements are in what is known as the problem domain, and the solution domain. Requirements that are formulated in the problem domain are commonly called stakeholder requirements, whereas the solution domain requirements are normally called system requirements. Furthermore these requirements can be split into functional and non-functional requirements. Functional requirements are requirements given to describe how the system will perform. The non-functional requirements are requirements that is used to define qualities the system is supposed to have [16].

3.1 Customers requirements

SH | MG

The requirements given by the customer are split into two parts, namely functional requirements and non-functional requirements. The first draft of requirements was given without a priority, except for the traditional 'shall' and 'should' wording. The team has since then added a priority, based on descending from 'A', 'B' and 'C', the priority and their descriptions are tabulated below in Table 3.1. As the requirements are being processed they are also attached a status description, that will provide a short overview of their completeness, these are tabulated in Table 3.2. To view the approved requirements, see section 5.4.

Table 3.1: Priority legend.

Legend	**Description**
A	Important! Required for project
B	High priority. Essential for project
C	Neutral. Optional for project
N/A	Not applicable

Table 3.2: Status legend.

Legend	Description
Approved	Passed test requirement
Awaiting	Implemented and awaiting test
Ongoing	Work in progress
Started	Implementation has started
Not started	Not started
Failed	Failed test requirement

3.1.1 Functional Requirements

Table 3.3: Customers functional requirements.

Req. ID	Description	Priority
FR-01	The System's Sensor shall detect objects	A
FR-02	The Systems Sensor shall identify and categorize the detected objects	A
FR-03	The System shall be controlled through a Graphical User Interface (GUI)	A
FR-04	The System's Effector shall hit marked targets	A
FR-05	The System's Effector shall only trigger against targets that are categorized as hostile	B
FR-06	The GUI shall display information of the System's operational status	B
FR-07	The GUI shall display information of the Effector's direction and orientation	B
FR-08	The GUI shall display information of the Sensor's identified objects	B
FR-09	The GUI shall display information of previous projectile firings (log)	C
FR-10	The GUI should display information of previous target assessment	C

3.1.2 Non-functional Requirements

Table 3.4: Customers non-functional requirements.

Req. ID	Description	Priority
NFR-01	The System shall be developed with commercially available components	N/A [1]
NFR-02	The System shall not be a risk for health and safety	A
NFR-03	The System's interfaces shall be documented	B
NFR-04	The System shall have a technical user guide	B

Early on, customer requirements were thoroughly discussed and established in collaboration with the client, outlining the primary tasks to be completed. The system requirements were then developed to a greater extent, translating the customer requirements into measurable and testable criteria. These system requirements have been subject to multiple iterations throughout the project's duration. The final iteration of the system requirements reached their current state when Mini Guardian transitioned into a proof of concept.

[1] NFR-01 was given Not Applicable (N/A) due to the projects latent effect of being a bachelors project, as we do not have access to our customers military products or any other military products through our customer.

3.2 System requirements

SH | MG

The customer requirements were expanded to system requirements, that are meant for internal use to further specify what needs to be done to complete and achieve the customer requirements. The system requirements are iterated throughout the project, as preliminary research will surface new restrictions of the project. The system requirements can be viewed in full in Table 3.5, the non-functional requirements are not expanded on, as these are defined well enough by the customer, except for NFR-02: "The system shall not be a risk for health and safety", as it needed to be measurable for testing. The non-functional system requirements are tabulated in Table 3.7.

Table 3.5: System requirements.

Req. ID.	**Sys. req. ID.**	**Description**	**Priority**
FR-01	**SR-01-01**	The Sensor shall measure distance	A
	SR-01-02	The Sensor shall detect objects within 2.5 ± 0.5m	A
	SR-01-03	The Sensor should detect objects from all azimuths	B
	SR-01-04	The Sensor should detect objects within the Effectors vertical range (0 - 40°)	C
	SR-01-05	The Sensor should detect newly appeared or relocated objects within 5 s	B
FR-02	**SR-02-01**	The Sensor should detect identifiable properties	B
	SR-02-02	The Sensor shall categorize targets as Hostile, Unknown and Friendly (HUF)	A
FR-03	**SR-03-01**	The Graphical User Interface (GUI) shall work on the computer belonging to the project	C
FR-04	**SR-04-01**	The Effector shall hit marked targets with a diameter of 30±5cm at a range of 2.5±0.5m	B
	SR-04-02	The Effector shall have azimuth accuracy within ±0.5 ° (degrees)	A
	SR-04-03	The Effector shall have elevation accuracy within ±0.5 ° (degrees)	A
FR-05	**SR-05-01**	The GUI's fire button shall only be enabled when target is categorized hostile	B
	SR-05-02	The GUI shall display if target marked friendly is tried fired upon	C

Table 3.6: System requirements.

Req. ID.	Sys. req. ID.	Description	Priority
FR-06	SR-06-01	The GUI shall display if Effector is safe or armed	B
	SR-06-02	The GUI shall display if Effector is disconnected or connected to Human-Machine Interface (HMI)	B
	SR-06-03	The GUI shall display if Effector is unpowered or powered	B
	SR-06-04	The GUI shall display whether the Effector projectile chambers are empty or loaded	B
FR-07	SR-07-01	The System shall know where Effector and Sensor are located according to a reference point	B
	SR-07-02	The GUI shall display Effectors azimuth and elevation	B
	SR-07-03	The Effector shall have a fixed reference/zero point	A
FR-08	SR-08-01	The GUI shall display target information information according to MIL-STD-1477	B
FR-09	SR-09-01	The GUI shall display which targets were fired upon	C
	SR-09-02	The GUI shall display when the projectile was fired	C
	SR-09-03	The GUI shall display what projectile canister was fired	C
	SR-09-04	The GUI should display whether the projectile hit the target	C
FR-10	SR-10-01	The GUI should have memory of previous objects and their HUF assessment	C
None	SR-11	The Sensor should have an enclosure	C

Table 3.7: Non-functional system requirements.

Req. ID.	NSR - ID.	Description	Priority
NFR-02	NSR-02-01	The system should have a physical emergency stop button	C
	NSR-02-02	The system shall have two mechanical safety barriers for firing projectiles	A

3.3 Minimum Viable Product

SH

A Minimum Viable Product (MVP) is the most basic version of a product that include essential features and functionalities to meet the core needs of the client [17]. Based on the set of requirements from the client, the MVP can be defined as a System where each component functions independently. That means that the LiDAR will detect objects and mark them in RViz within the Robot Operating System (ROS) environment. The camera component captures visual data and identifies targets as Hostile, Unknown, or Friendly. The target information can be read manually after the identifiable data is written to the serial port. The Effector will be controlled through the GUI to accurately hit targets at a distance of 2.5 meters, a distance restricted by the Sensor proof of concept. The Power Supply will deliver power to all components from a single module.

By focusing on these core functionalities, the MVP provides a functional system that demonstrates the fundamental capabilities of Mini Guardian. It allows for testing and validation of the individual components while fulfilling the basic requirements of object detection, target identification, accurate hitting and a streamlined power management. As the project progresses, additional features and enhancements can be incorporated to further refine and expand the Systems capabilities.

Chapter 4

Risk

SH | MG

In a project involving multiple disciplines such as software, electronics, and mechanical design, it is crucial to consider risks that could occur. Each discipline brings its own unique challenges and potential risks. By proactively identifying and assessing risks specific to software, electronics, and mechanical design, the project members can anticipate potential issues that may arise during the project's execution. This allows for early mitigation measures to be put in place, reducing the likelihood of delays or failures.

Integrating different disciplines seamlessly is often a complex task. Risks associated with compatibility issues, interface complexities or conflicting design requirements can emerge during the integration process. By considering and addressing these risks in advance, the project members can develop strategies to ensure smooth integration and reduce the likelihood of technical hurdles. As there is no fully functional simulation - a digital twin - of the Effector, the software team needs access to it for testing. This means that the Effector needs to be operational, which in turn means that for example electronics can't perform hardware upgrades or changes to the system. This is one of the reasons that good planning for each discipline in essential is interdisciplinary projects. By identifying potential risks and implementing contingency plans, the team can proactively manage project timelines, align tasks effectively and minimize schedule disruptions.

The project may require specialized resources, such as specific software tools, hardware components or expertise in each discipline. Risks related to resource availability, purchasing delays or budget constraints can impact the progress of the project. By identifying and managing these risks, the team can ensure that the necessary resources are allocated appropriately, minimizing disruptions and optimizing project outcomes.

Effective communication and collaboration among team members from different disciplines are essential for project success. Risks associated with miscommunication, differing interpretations of requirements, or conflicting priorities can hinder progress and lead to misunderstandings. By recognizing these risks, the team members can establish clear communication channels, promote collaboration, and facilitate efficient information sharing, ultimately enhancing overall project performance. Having a good project model in place makes communication and collaboration an easier task to perform, and Scrumban has proven to be an efficient tool.

Considering risks in an interdisciplinary project is vital to minimize uncertainties, enhance coordination among disciplines, optimize resource allocation, manage time effectively, maintain quality and ultimately increase the chances of successful project completion. It allows the project team to proactively address potential challenges, ensure smooth integration of different disciplines and deliver a well-executed and impactful thesis [18].

The 5x5 risk matrix is a popular tool to visualize risk evaluation and prioritization [19]. Our matrix is shown in Table 4.1.

Table 4.1: 5x5 Risk matrix.

	Insignificant	**Minor**	**Significant**	**Major**	**Severe**
	1	**2**	**3**	**4**	**5**
1 Rare	Very low 1	Very low 2	Low 3	Medium 4	Medium 5
2 Unlikely	Very low 2	Medium 4	Medium 6	Medium 8	High 10
3 Moderate	Low 3	Medium 6	Medium 9	High 12	Very high 15
4 Likely	Medium 4	Medium 8	High 12	Very high 16	Extreme 20
5 Almost Certain	Medium 5	High 10	Very high 15	Extreme 20	Extreme 25

The risks identified for project Mini Guardian can be seen in Table 4.2 and Table 4.3. The risks are split into the categories interpersonal risks and project risks respectively. The interpersonal risks are potential challenges or conflicts that may arise between team members or stakeholders. The project risks are linked to the risks that directly impact the project. The interpersonal risks can be seen as a subset of the project risks that specifically relate to human interactions within the project team. The risk analysis evaluates the potential impact of risks before implementing any actions. Once the necessary actions are taken, the impact of the identified risks is reduced, reaching a level that is considered acceptable for the project to proceed.

Table 4.2: Interpersonal Risks

Interpersonal Risks	Probability	Impact	Total risk	Measures/action
Illness	5	2	10	Work from home if possible
Group conflict	3	5	15	Be nice, take the problem by root
Bad communication	3	4	12	Scrumban , morning meetings
Injuries and/or longer periods of illness	1	5	5	Adapt accordingly
Reduced well-being	3	5	15	Daily coffee and other welfare measures
Conflict with external groups	5	1	5	A little bit of secrecy and documentation of cooperation
Group member or supervisor dropouts	1	5	5	Facilitate arrangements
International conflict	1	5	5	Nato, project will be put on hold

Table 4.3: Project Risks

Project Risks	Probability	Impact	Total risk	Measures/action
Overdue parts delivery, or parts not in stock	3	5	15	Order early, and make sure items are in stock at vendor
Damaged in house produced parts/components	4	3	12	Design to be replaced easily and easy to manufacture
Damaged ordered parts/components	2	5	10	Order replacements parts, be particularly careful with these parts
Projectiles hitting unintended targets	1	4	4	Two or more safety barriers
Long delivery	4	5	20	Order early, plan ahead, check other vendors
Wrong choice of sensor	2	4	8	AHP
Parts delivered defect/damaged under transport	2	3	6	Order new, get compensated
Trouble with OneDrive	4	4	16	Local backup at least every week. Towards end of project every day
Break the budget limits	4	3	12	Pitch new ideas to increase the budget. Acquire hardware other ways
Unable to machine parts	1	1	1	3D-print instead, outsource problems
Short circuits	4	4	16	Be careful, spare parts
Deletion of data	3	5	15	Backup
High demand on 3D-printers	4	4	16	Make parts early, or use own or other printers available
Tools break	3	2	6	Be careful, buy new tools at Tools
Too difficult task?	2	5	10	Reduce scope
AI fails	2	5	10	Spend time training the AI, look for alternatives if failed
Sensor is shot by effector	1	5	5	Create restricted zone around sensor, prevent engaging target
Dangerous air pressure event	2	5	10	Use relevant protective gear
Crushing hazard	1	3	3	Keep clear of moving parts
Electrical shock	2	4	8	Work on unpowered system and always keep two safety barriers
Time frame not kept	4	4	16	Gantt and daily/weekly meetings. Deliver what is finished
Interdisciplinary project challenges	3	2	6	Effective communication and collaboration between disciplines
Cutting hazard by tools	5	2	10	Keep track of where fingers and extremeties are. Band-aid
Rotating tool hazards	5	2	10	Keep track of where fingers and extremeties are. Band-aid
No digital twin	5	3	15	Keep timetables for testing on System
Effector firing at friendly target	2	5	10	An operator operating the firing button. Unable to fire against friendly
Discovering unknown problems from previous group	4	4	16	Talk with previous group, pick System apart
Problem integrating components together	4	4	16	Good planning for integration
Problem with enclosure for Sensor	1	3	3	Design enclosure to fit the project needs
Wrong material choices	2	5	10	Research different materials, FEA
Computer not powerful enough	4	5	20	Optimize code, acquire computer dimensioned for project

Chapter 5

Testing

MH | MG

To verify that the given functional and non-functional requirements have been met a testing phase will be completed in the timeline of the project. This is because the stated requirements are wanted by the customer, which in turn is to represent a given functionality that is wanted from the system. To make sure that the customer requirements have been met, they need to be verified. To achieve this, objective evidence has to be gained to prove that the requirements have been fulfilled. The objective evidence can be collected in different ways throughout the projects lifespan, and the scope of collection may also vary. Testing can be performed with verification or validation in mind, where verification is the process that seeks to confirm that the stated requirements have been met, and validation is the process that seeks to approve that the correct system has been created with the given need or opportunity in mind. It may seem that these two processes are redundant, but they are in fact complementary activities, both are equally needed in some projects. Furthermore in some projects it may be difficult to gain enough objective evidence to prove that all the given functional requirements have been fulfilled. To fully investigate and assess a system demands time and resources, and as a result, time and money are constraints when testing. This in turn makes it so that a full verification and validation phase of a system may not always be achievable. The end goal is therefore to have a flexible approach when testing to get the phase as comprehensive as possible with the given system, time and budget constraints [16].

With these constraints in mind for project Mini Guardian, a verification process of the given system requirements will take place. The project will not go trough a validation process due to the fact that we are creating a system on the basis of the customer requirements and do not have information about the given need or opportunity. Another reason is that the time frame is short and the project will have to utilize the time elsewhere. This makes it so that testing will be conducted throughout the project where appropriate. We will also have a testing phase at the end of the project, this is to gain objective evidence that the specified systems requirements have been reached.

5.1 Verification

MH | **JM**

Verification is the name of the process where the objective is to gather enough information in order to provide evidence that the system or system elements satisfies their respective system requirements. To gather information the verification process uses different techniques, rules or standards to identify errors, faults or defects. These gathering methods are used on the implemented system or system elements and are described in the list below [20].

1. **Inspection** - The inspection method of testing involves the use of visual or dimensional examination of an object, the method is also generally nondestructive. This method relies on human senses such as sight, hearing, smell, taste and touch, furthermore the method can also use simple methods of measurement. It can also include other means such as simple physical manipulation, or electrical and mechanical measuring instrument such as gauges. When performing an inspection test it is most common to check properties that are best confirmed by the use of observation, such as paint color, surface finish, physical condition and construction features [21].

2. **Demonstration** - The demonstration method of testing is used to check if the system is functioning correctly. A demonstration utilizes little to no physical data collection, and instead it relies on a general set of selected actions. These actions are meant to show that the system responds in a way that operators can perform their tasks when using the system. This may include serviceability, transport-ability or accessibility. The observations made are compared with a set of expected or predetermined responses [21].

3. **Test** - A test is a method that involves a series of controlled inputs to a system, in order to verify its performance. The inputs can either be real or simulated, furthermore the test method may involve the use of specific instruments or equipment to collect quantitative data. The data is then analyzed to give an understanding of how the system performs. A typical way of testing is to do a function test of the system both before and after an environmental test to thoroughly verify the given systems performance [21].

4. **Analysis** - The analysis method is used when testing in real conditions is problematic. This method utilizes the usage of mathematical models to generate information of a systems performance, and in turn decide if a system requirement is fulfilled. The mathematical models can include the use of systems engineering analysis, computer modeling, computer and hardware simulations and statistical and qualitative analysis. The analysis method is often applied in the design phase, where the systems performance can only be estimated or predicted. It is however not limited to this phase. The use of the analysis method can also be appropriate for verifying reliability system requirements [16].

In the requirements phase of the project a great effort is to identify the appropriate verification method that will be used when confirming a requirement. The most appropriate verification methods are dictated by financial, operational and technical considerations. A given requirement can have various appropriate verification methods in the projects life-cycle, this is to reinforce the evidence gained with the end goal of showing that the given requirement has been met. Usually the analysis verification method may be most appropriate in the early design stages, whereas a demonstration method may be most applicable when a working prototype of the system is achieved. When selecting a method or methods to verify a requirement it is also important to specify a detailed scope, and how the test will be performed. To give an example a requirement may state that maintenance tasks should be performed within a given time. This alone will not be enough, it should also state how many times the given test is supposed to be carried out. Furthermore, an environment the test will be performed in needs to be given and by whom. Finally a statistical confidence will conclude to what extend the given requirement has been met or not [16].

5.2 Test specifications

MG | MH

After all the system requirements were identified, a verification method was needed to gain objective evidence to show that the given requirements were fulfilled. Throughout the project the test method has been changed if it has been noted that a different verification method is more appropriate. When choosing a verification method the list in section 5.1 has been used. The resulting methods can be seen in Table 5.1 where the left column shows the ID of functional requirements given by the customer, the middle left column shows the ID for the system requirements, and the middle right column shows the test ID, and finally the chosen test method is seen in the right most column. Table 5.2 shows the given acceptance criteria for all the tests to be performed to verify the systems requirements within the project.

Table 5.1: Test specifications

Req. ID.	System req. ID.	Test ID.	Test Method
FR-01	SR-01-01	TI-01-01	Test
	SR-01-02	TI-01-02	Test
	SR-01-03	TI-01-03	Test
	SR-01-04	TI-01-04	Analysis
	SR-01-05	TI-01-05	Test
FR-02	SR-02-01	TI-02-01	Demonstration
	SR-02-02	TI-02-02	Demonstration
FR-03	SR-03-01	TI-03-01	Demonstration
FR-04	SR-04-01	TI-04-01	Demonstration
	SR-04-02	TI-04-02	Test
	SR-04-03	TI-04-03	Test
FR-05	SR-05-01	TI-05-01	Test
	SR-05-02	TI-05-02	Test
FR-06	SR-06-01	TI-06-01	Demonstration
	SR-06-02	TI-06-02	Demonstration
	SR-06-03	TI-06-03	Demonstration
	SR-06-04	TI-06-04	Demonstration
FR-07	SR-07-01	TI-07-01	Demonstration
	SR-07-02	TI-07-02	Test
	SR-07-03	TI-07-03	Test
FR-08	SR-08-01	TI-08-01	Demonstration
FR-09	SR-09-01	TI-09-01	Demonstration
	SR-09-02	TI-09-02	Demonstration
	SR-09-03	TI-09-03	Demonstration
	SR-09-04	TI-09-04	Demonstration
FR-10	SR-10-01	TI-10-01	Demonstration
None	SR-11-01	TI-11-01	Inspection
NFR-02	NSR-02-01	TI-22-01	Test
	NSR-02-02	TI-22-02	Demonstration

Table 5.2: Test ID. with given accept criteria

Req. ID.	Sys. req. ID.	Test ID.	Acceptance criteria
FR-01	SR-01-01	TI-01-01	The Sensor will aim at an object with a given distance. The measured distance and the given distance will be compared. The test is deemed successful if the measured distance is ± 3 cm of given distance.
	SR-01-02	TI-01-02	Targets will be placed within a range of 3m, and the LiDAR feed shall display a detected object. The test is deemed successful if the LiDAR displays a detected object 4 out of 5 times.
	SR-01-03	TI-01-03	Targets will be placed in different azimuth locations, and the target will be displayed in the LiDAR feed. The test i deemed succesfull if target is detected 4 out of 5 times.
	SR-01-04	TI-01-04	Sensors range and angle will be analyzed using appropriate tool, deemed succesfull if angle is within range of Effector.
	SR-01-05	TI-01-05	New object will be placed in sensors line of sight, time will be taken using stopwatch and compared to when object appears in LiDAR feed. Deemed succesfull if object appears within 5 s, 5 out of 5 times
FR-02	SR-02-01	TI-02-01	When detecting object, identifiable (color) properties shall display in the GUI 4 out of 5 times.
	SR-02-02	TI-02-02	The Sensor will aim at HUF objects, the test is successful if the GUI displays the correct categorized object.
FR-03	SR-03-01	TI-03-01	An attempt to run the GUI on the project computer will be performed.
FR-04	SR-04-01	TI-04-01	Targets with a diameter of 30 cm ± 5 cm will be placed within a range of 3 m and fired upon 5 times by each individual canister. If the target is hit 4 out of 5 times from a canister, the test is deemed successful.
	SR-04-02	TI-04-02	The Effector's azimuth will be compared to a reference measurement instrument. The test is deemed successful when repeatability is within 1 degrees 10 out of 10 times.
	SR-04-03	TI-04-03	The Effector's elevation will be compared to a reference measurement instrument. The test is deemed successful when repeatability is within 0.5 degrees 10 out of 10 times.

Req. ID.	Sys. req. ID.	Test ID.	Acceptance criteria
FR-05	SR-05-01	TI-05-01	In a safe environment, we will attempt to trigger the Effector when the target is categorized as Friendly or Unknown. The test is deemed successful if action is prohibited 5 out of 5 times.
	SR-05-02	TI-05-02	Attempts to fire on non-hostile targets will appear in the GUI, deemed successful if the attempts appear in GUI 10 out of 10 times.
FR-06	SR-06-01	TI-06-01	Demonstrate that status displayed on the GUI corresponds with the actual status of the Effector.
	SR-06-02	TI-06-02	Demonstrate that status displayed on the GUI corresponds with the actual status of the Effector.
	SR-06-03	TI-06-03	Demonstrate that status displayed on the GUI corresponds with the actual status of the Effector.
	SR-06-04	TI-06-04	Demonstrate that status displayed on the GUI corresponds with the actual status of the Effector.
FR-07	SR-07-01	TI-07-01	If information about locations appear in the GUI the test is deemed successful.
	SR-07-02	TI-07-02	The GUI shall display elevation and azimuth values. If value appears, the test is deemed successful.
	SR-07-03	TI-07-03	When the Effector hits reference points, elevation and azimuth values goes to zero in the GUI. If values goes to zero at reference points, the test is deemed successful.
FR-08	SR-08-01	TI-08-01	If correct HUF information appears on identified targets in the GUI according to MIL-STD-1477, the test is deemed successful.
FR-09	SR-09-01	TI-09-01	If correct target information appears in firing log = ok
	SR-09-02	TI-09-02	If correct date information appears in firing log = ok
	SR-09-03	TI-09-03	If correct firing data, i.e. what canister used appears in firing log = ok.
	SR-09-04	TI-09-04	The Effector will fire upon a hostile target, and if the object dissapears the GUI will display this information. Deemed succesful i correct information is displayed 4 out of 5 times.

Req. ID.	**Sys. req. ID.**	**Test ID.**	**Acceptance criteria**
FR-10	SR-10-01	TI-10-01	Information about previous target assessments are shown in the GUI. If information is shown, the test is deemed successful.
None	SR-11-01	TI-11-01	By inspection, if the Sensor has an enclosure, the test is deemed successful.
NFR-02	NSR-02-01	TI-22-01	When Effector moves and the emergency stop button is pressed the system will stop. Deemed successful if correct 10 out 10 times.
	NSR-02-02	TI-22-02	Demonstrate the two barriers. If sufficient, deemed successful.

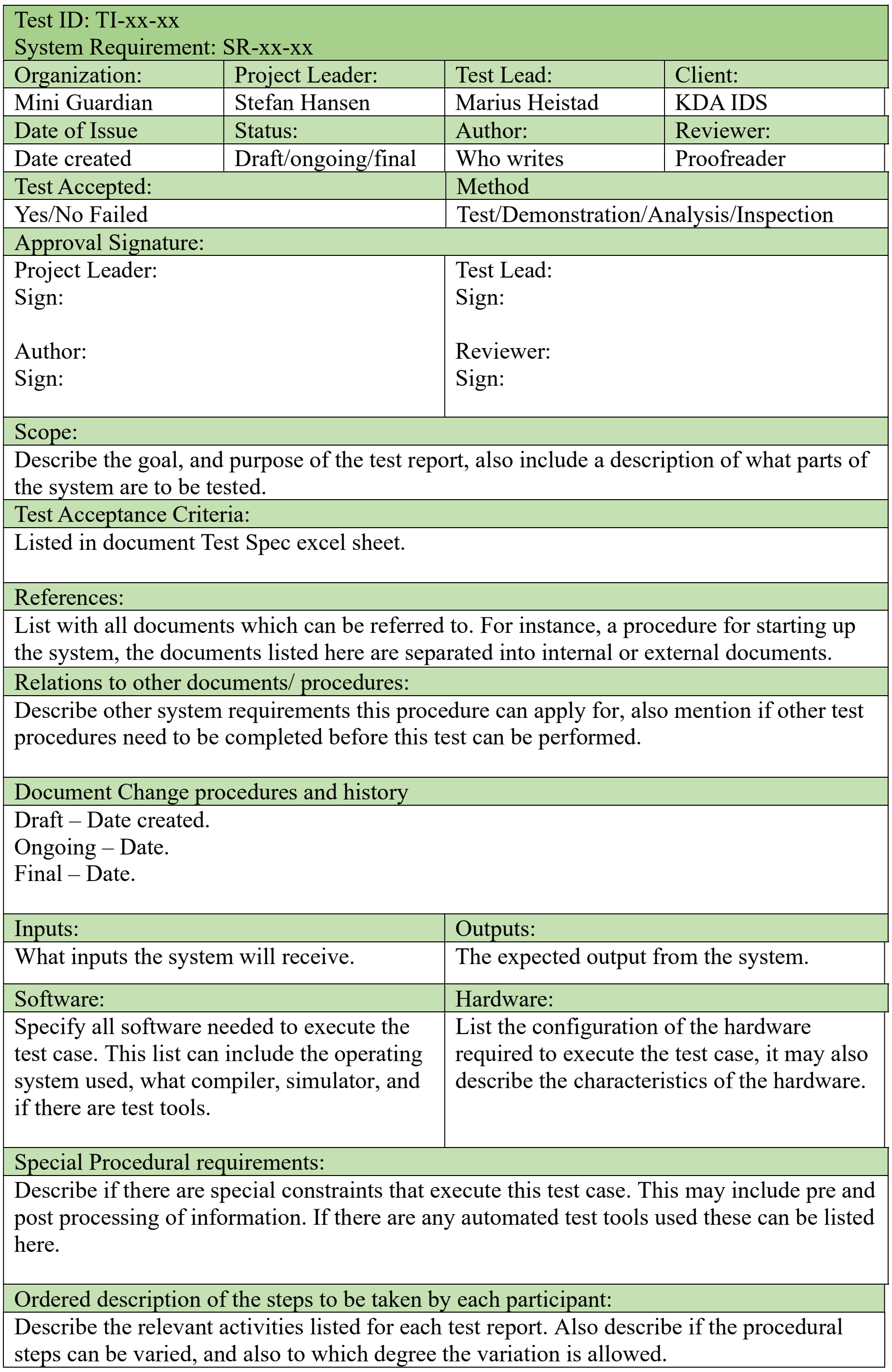
5.3 Test documentation template

MH | MG

To make sure that when testing different solutions in the project, a standard format is used when creating the test report template. The format is based on the IEEE - Std 829-2008 [22] and the ISO/IEC/IEE 29119 [23] standard. These standards are comprehensive and complicated to understand, which resulted in a lot of time being spent researching. Because the bachelor project is a relatively small and short project, the group has decided to partially use the standard. As a result the test report template shown in Table 5.3 and Table 5.4 is a compressed document template, which includes the level test design, level test case, level test procedure and level test log from [22].

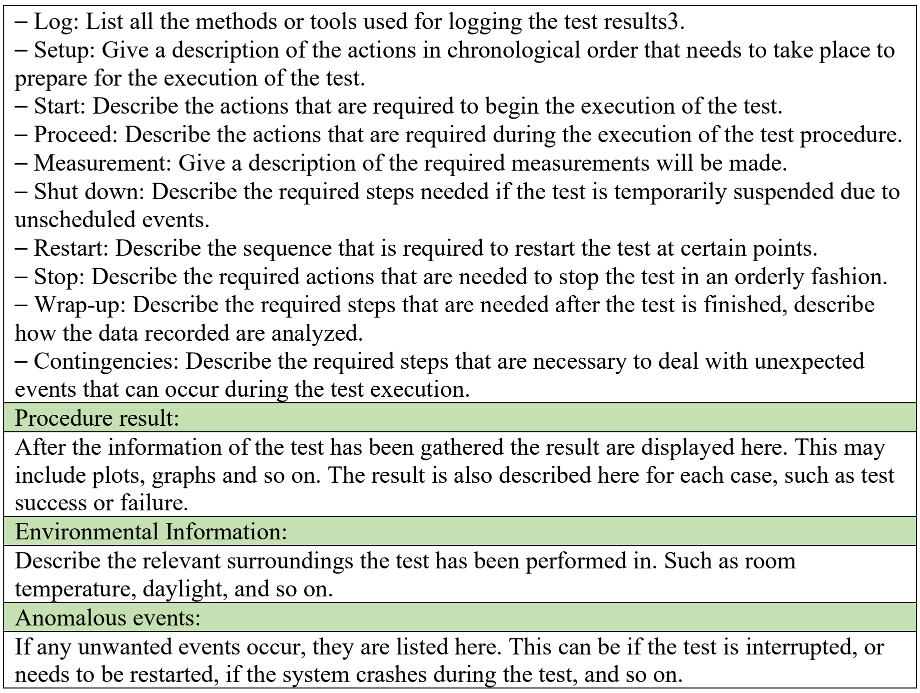
The test template was created to have a report that documents testing of system requirements, this was in order to make sure that information was not lost during testing, and also to have a template that is easy to maneuver. In the template each section is provided with a little informational text that says what information is to be included in the given section. The reason for this was to make it so that in theory everyone can open the template and fill in the sections without needing to get an introduction on how the documentation needs to be.

Table 5.3: First page test report template



Test ID: TI-xx-xx System Requirement: SR-xx-xx			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
Date created	Draft/ongoing/final	Who writes	Proofreader
Test Accepted:		Method	
Yes/No Failed		Test/Demonstration/Analysis/Inspection	
Approval Signature:			
Project Leader: Sign: Author: Sign:		Test Lead: Sign: Reviewer: Sign:	
Scope:			
Describe the goal, and purpose of the test report, also include a description of what parts of the system are to be tested.			
Test Acceptance Criteria:			
Listed in document Test Spec excel sheet.			
References:			
List with all documents which can be referred to. For instance, a procedure for starting up the system, the documents listed here are separated into internal or external documents.			
Relations to other documents/ procedures:			
Describe other system requirements this procedure can apply for, also mention if other test procedures need to be completed before this test can be performed.			
Document Change procedures and history			
Draft – Date created. Ongoing – Date. Final – Date.			
Inputs:		Outputs:	
What inputs the system will receive.		The expected output from the system.	
Software:		Hardware:	
Specify all software needed to execute the test case. This list can include the operating system used, what compiler, simulator, and if there are test tools.		List the configuration of the hardware required to execute the test case, it may also describe the characteristics of the hardware.	
Special Procedural requirements:			
Describe if there are special constraints that execute this test case. This may include pre and post processing of information. If there are any automated test tools used these can be listed here.			
Ordered description of the steps to be taken by each participant:			
Describe the relevant activities listed for each test report. Also describe if the procedural steps can be varied, and also to which degree the variation is allowed.			

Table 5.4: Second page test report template



– Log: List all the methods or tools used for logging the test results3. – Setup: Give a description of the actions in chronological order that needs to take place to prepare for the execution of the test. – Start: Describe the actions that are required to begin the execution of the test. – Proceed: Describe the actions that are required during the execution of the test procedure. – Measurement: Give a description of the required measurements will be made. – Shut down: Describe the required steps needed if the test is temporarily suspended due to unscheduled events. – Restart: Describe the sequence that is required to restart the test at certain points. – Stop: Describe the required actions that are needed to stop the test in an orderly fashion. – Wrap-up: Describe the required steps that are needed after the test is finished, describe how the data recorded are analyzed. – Contingencies: Describe the required steps that are necessary to deal with unexpected events that can occur during the test execution.
Procedure result:
After the information of the test has been gathered the result are displayed here. This may include plots, graphs and so on. The result is also described here for each case, such as test success or failure.
Environmental Information:
Describe the relevant surroundings the test has been performed in. Such as room temperature, daylight, and so on.
Anomalous events:
If any unwanted events occur, they are listed here. This can be if the test is interrupted, or needs to be restarted, if the system crashes during the test, and so on.

5.4 Performed tests

MG

Table 5.5 shows the performed test, and their status. Furthermore, it also shows who has performed the test, and by whom the test has been approved. The test document column lists the test report in the Appendix.

Table 5.5: Performed Tests

Test ID	Test Method	Done by	Status	Approved by	Test document
TI-01-01	Test	VJ	Accepted	JM	A
TI-01-02	Test	JM	Accepted	VJ	B
TI-01-03	Test	JM	Accepted	VJ	C
TI-01-04	Analysis	KC	Accepted	MH	D
TI-01-05	Test	VJ	Accepted	JM	E
TI-02-01	Demonstration	JM	Accepted	VJ	F
TI-02-02	Demonstration	JM	Accepted	VJ	F
TI-03-01	Demonstration	VJ	Accepted	JM	G
TI-04-01	Demonstration	JM	Accepted	VJ	H
TI-04-02	Test	MH	Accepted	KC	I
TI-04-03	Test	MH	Accepted	KC	J
TI-05-01	Test	VJ	Accepted	JM	K
TI-05-02	Test	VJ	Failed	JM	L
TI-06-01	Demonstration	JM	Accepted	VJ	M
TI-06-02	Demonstration	JM	Accepted	VJ	M
TI-06-03	Demonstration	JM	Accepted	VJ	M
TI-06-04	Demonstration	JM	Accepted	VJ	M
TI-07-01	Demonstration	JM	Accepted	KC	N
TI-07-02	Test	JM	Accepted	VJ	O
TI-07-03	Test	JM	Accepted	KC	P
TI-08-01	Demonstration	VJ	Accepted	JM	Q
TI-09-01	Demonstration	JM	Accepted	KC	R
TI-09-02	Demonstration	JM	Accepted	VJ	S
TI-09-03	Demonstration	JM	Accepted	VJ	S
TI-09-04	Demonstration	JM	Accepted	VJ	T
TI-10-01	Demonstration	JM	Accepted	VJ	U
TI-11-01	Inspection	KC	Accepted	MH	V
TI-22-01	Test	SH	Accepted	MH	W
TI-22-02	Demonstration	KC	Accepted	MH	X

Chapter 6

Project Mini Launcher overview

6.1 Inherited Elements

SH

Several elements and components were inherited from Project Mini Launcher, providing a starting point for the further development of the scaled-down version of the NASAMS. These inherited aspects include the launcher itself known in this project as the Effector, the Fire Distribution Center (FDC) model, and the Graphical User Interface (GUI).

The Effector seen in fig. 6.1, which serves as the firing mechanism of Mini Guardian, was originally designed and developed as a downscaled replica of the NASAMS launcher. This inheritance ensures continuity and consistency with the original system, keeping the similarity of shape for allowing for a transition and integration of the scaled-down version into the overall project framework.

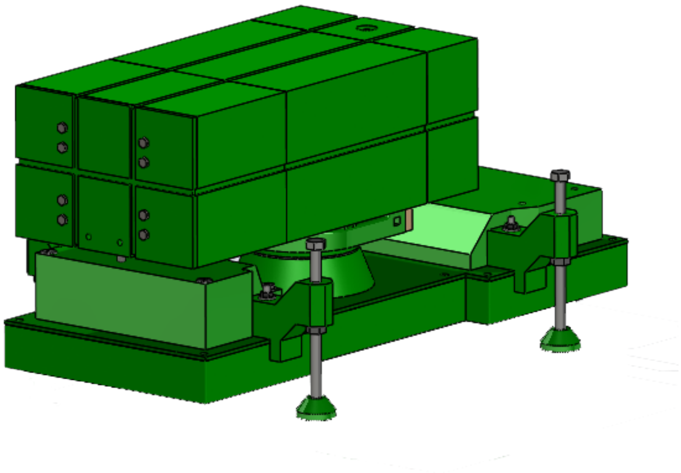
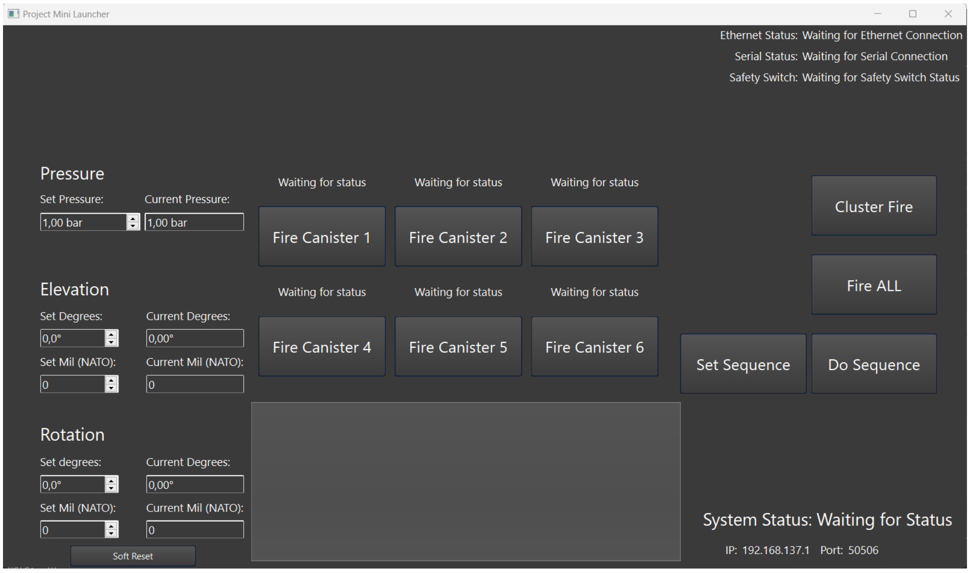


Figure 6.1: Inherited Effector

The FDC inherited from Project Mini Launcher acted like the power supply for Project Mini Launcher. They also had a pressure tank inside it with a pressure regulator.

Additionally, the GUI inherited from Project Mini Launcher serves as the primary interface for operators to interact with the System. The GUI featured setpoints for azimuth and elevation control, allowing to set a wanted degree and move the Effector to aim in that direction. Furthermore, it included dedicated firing buttons corresponding to each of the six canisters present in the launcher. Fig. 6.2 shows the state of the GUI when taking over the project.



Project Mini Launcher

Ethernet Status: Waiting for Ethernet Connection
Serial Status: Waiting for Serial Connection
Safety Switch: Waiting for Safety Switch Status

Pressure
Set Pressure: 1,00 bar
Current Pressure: 1,00 bar

Elevation
Set Degrees: 0,0°
Current Degrees: 0,00°
Set Mil (NATO): 0
Current Mil (NATO): 0

Rotation
Set degrees: 0,0°
Current Degrees: 0,00°
Set Mil (NATO): 0
Current Mil (NATO): 0

Waiting for status

System Status: Waiting for Status
IP: 192.168.137.1 Port: 50506

Figure 6.2: The GUI of Project Mini Guardian.

By inheriting these elements, they act as the foundation of project Mini Guardian. That inheritance comes with the well-established starting point of a functional system, but also comes with the foundation laid for some solutions and how they were executed, including the problems and issues that come with that.

6.2 Project State at Takeover

SH

Mini Guardian is a continuation of an earlier project, the so-called Project Mini Launcher [1]. As mentioned earlier, their objective was to recreate a functional scale model of the KDA developed system NASAMS in a 1:12.5 scale.

Since Mini Guardian's goal is for the Effector to communicate with the Sensor to detect, identify and hit objects, the requirements are clearly different from those of Project Mini Launcher. For Mini Guardian to be successful, there has been demanded tougher requirements regarding precision. The stakeholder requirement FR-04 with the belonging system requirements SR-04-01, SR-04-02 and SR-04-03 are therefore implemented, see Table 3.5.

The mentioned stakeholder – and system requirements concerns the Effectors accuracy both in azimuth and elevation, as well as the ability to hit specified sized targets.

With this in mind, there has been conducted several exercises to inspect and get to know the physical system. There has been testing of the firing-mechanism, the elevation and azimuth accuracy, the Effector has been dismantled and meetings with representants from Project Mini Launcher have been conducted. During these exercises, several challenges have been uncovered.

6.2.1 Software

JM

Upon taking over the project, we encountered some known faults in the software. However, as the project progressed, several additional issues appeared. The software architecture provided from the previous year turned out to be different from the one actually implemented, which initially posed difficulties in understanding the system's functionality. After conducted testing and engaging in meetings with individuals involved in the project last year, it became clear that a significant portion of the code was either unused, sub-optimal, or unnecessary for our requirements.

Most of the known faults within the project related to the Graphical User Interface (GUI), and there was a significant amount of hard-coded functionality that required improvement. There was a lot of code that was made to make the system look good, but in reality had little or no function to it.

6.2.2 Electronics

SH

During the course of the project, several areas were identified in the electronics aspect that could benefit from improvements and refinements. These findings shed light on potential enhancements that can be made to ensure better functionality and reliability.

When we were handed the project, the only thing we were foreshadowed was that the Effectors azimuth was inaccurate, likely due to the encoder being placed in a corner of the belt being tensioned way above its limits of 4.90 N of radial shaft loading [24].

Inside the FDC there was an air pressure tank belonging to an old paintball gun. We were told that it was 16 years old and had never been tested by Project Mini Launcher. It seemed to be a dangerous piece of equipment for continued use. We had been given a Cocraft HC 16 air compressor at the start of our project so we wouldn't have to rely on the paintball gun pressure tank either way.

Other challenges were identified, such as the power supply to the Arduino Mega being insufficient. It had to be connected to the USB-B port to get enough power. This workaround indicates the need for a more robust and adequate power supply system to ensure stable and reliable operation of the Arduino board without relying on external sources.

The soldering in general was in a bad state. The soldering quality on the Arduino Mega and Nano indicate the importance of ensuring proper soldering techniques to prevent loose connections or potential failures. Later after an inspection of the homemade firing circuit, it was revealed that it too needed a revamp.

The tensioning of the belt was also identified as a challenge, as achieving proper tension proved difficult. After some testing we observed the stick slip phenomenon which further emphasized the need for a more robust and reliable tensioning mechanism to ensure stable and consistent belt operation. Finding a suitable solution to address this issue would enhance the Effectors reliability and overall performance.

Lastly, the complexity and quality of the previous wiring diagrams were found to be inadequate. A well-structured and comprehensible wiring diagram is essential for troubleshooting, maintenance and future development. Restructuring and organizing of the wiring diagram would facilitate easier comprehension, reduce errors and improve the overall usability and maintainability of the system. Especially since it was foreshadowed that the client might want to produce several of these Effectors in future bachelor projects.

6.2.3 Mechanics

KC

After receiving the Effector, meetings with former Project Mini Launcher members and several tests was initiated. During these meetings, tests and inspections, a few significant inaccuracies surfaced.

As already mentioned, the Effectors azimuth was not accurate enough to hit the detected targets. When inspected, it was clear that the Effector did not move to the desired degree and the inaccuracy was not consistent. The movement was not running smoothly and the Effector seemed to struggle. This issue can have many reasons, both mechanical and electrical. One of these reasons may be static friction between the rotation base elements. A lack of structural strength may also contribute to this.

The Effectors elevation movement was a lot more smooth and consistent than the azimuth, but a major issue here was the fact that the Effector would not elevate at all times. This happened when the axle the Effector rotates around was not kept static in its place. Due to material choice, this was difficult to achieve.

Other mechanical aspects in need of improvements or redesign is the projectiles and the air pressure system. The projectiles fracture easily because of the polymer material used, as well as the print orientation. The air hose system leaks air, emptying the compressor tank and reducing the desired pressure when firing.

6.2.4 Summary

In short, these were the elements in need of an upgrade, in no particular order:

- Effectors azimuth accuracy
- Effectors elevation mechanism
- The belt drive
- The projectiles
- The air pressure system.
- Encoder problems
- Power supply makeover
- Soldering
- Wiring diagram
- Safety features
- GUI

Chapter 7

Project Mini Guardian

SH

This chapter presents the process of selecting a sensor. It begins with an exploration of important characteristics and preliminary research. Testing of available sensors is then discussed, leading into the selection process. The chosen Sensor, along with why project Mini Guardian was transformed into a proof of concept is described. Fig. 7.1 provides an overview of the System.

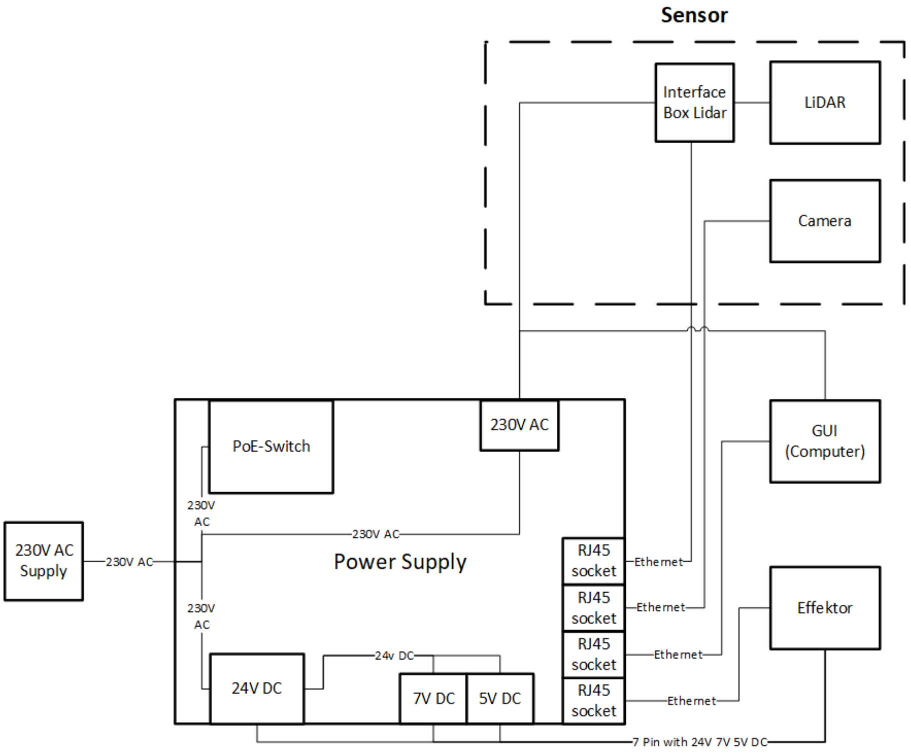


Figure 7.1: Main overview of the System.

7.1 Sensor Selection

SH

As mentioned in the introduction of Project Mini Guardian, see section 1.4, the main objective for this years mission was highlighted as the implementation of the crucial

Sensor within the scaled-down air defense system. Inspired by the air defense system NASAMS, the goal is to develop a functional model that replicates key functionalities and components on a reduced scale. It is therefore essential to choose the appropriate sensor solution for this application. The Sensor, consisting of a LiDAR sensor and an AXIS camera, plays the role of detecting and identifying targets. This chapter aims to provide an insight into various sensor options, relevant sensor characteristics and a reasoned and objective choice of Sensor solution based on the Analytic Hierarchy Process (AHP).

7.1.1 Characteristics

SH

There are a number of factors that need to be considered in the selection of a sensor [3]. To define the required characteristics for the sensors application, sensor characteristics have been derived from the system requirements from Table 3.5. These are used as the underlying reasoning for the sensor selection process, which is tabulated in Table 7.1 below. This table also shows what specification was rated more attractively in the AHP. These are the criteria that the sensor(s) decision making will be generated out of, see the section Selection process.

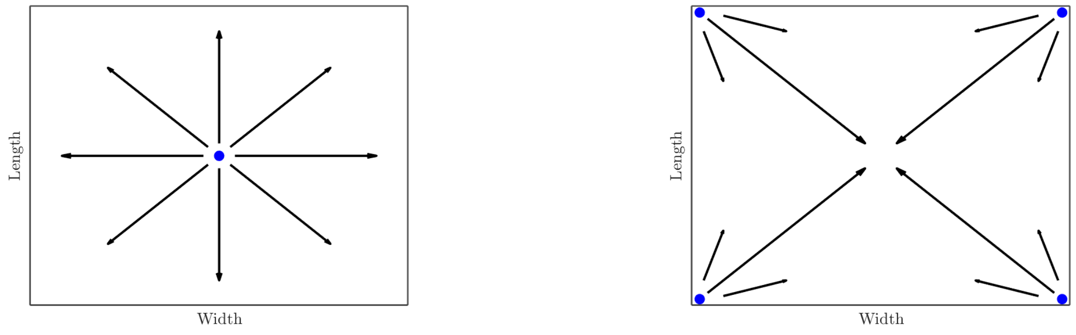
Table 7.1: Sensor characteristics of interest.

Characteristic	**Specification**	**Requirement**
Range	Longer is better	SR-01-02
Identify	More ways of identifying objects is better	FR-02
Ranging information	How and can it retrieve a coordinate point?	SR-01-01
Accuracy	Smaller error range is better	SR-04-01
Sample time	Faster is better	SR-01-05
Vertical FoV	From 0 ° to 90 ° is optimal	SR-01-04
Horizontal FoV	Closer to 360 ° is better	SR-01-03
Safety	Safer is better	NFR-02
Price	Lower is better	Budget
Enclosure	Yes/No	SR-11

Arrangement

DS | SH

The arrangement of the System does affect area compatibility which is correlated to the sensor selection process. There is mainly two ways to arrange the sensor or sensors. The inside-out arrangement with a central sensor cluster that looks outwards as illustrated in fig. 7.2a, or the outside-in arrangement - a decentralized arrangement of sensors scattered around the area of interest that looks inwards as shown in fig. 7.2b. The sensor placement(s) are shown in blue, the arrows show direction of sensing. These arrangements do not consider the placement of the Effector.



(a) Inside-out.

(b) Outside-in.

Figure 7.2: Sensor arrangement.

To not restrict the area the System can work in, it is preferred to have a sensor option which is inside-out, such that it is the sensors range that is the limiting factor and not the room it is placed in.

Active and passive

DS

The sensor solution can either be passive or active. Active sensors send and receive information, whereas passive sensors only receive.

7.1.2 Preliminary Research

SH

Conducting preliminary research is vital when selecting a sensor for a project like Mini Guardian. The choice of sensor plays a crucial role in the Systems overall functionality and performance. By conducting research, the project team can explore various sensor options, compare their specifications, capabilities and compatibility with the project requirements. This research enables informed and objective decision-making, ensuring that the selected sensor aligns with the projects goals and objectives. Additionally, preliminary research helps identify any potential limitations or challenges associated with the sensor, allowing the team to plan ahead and develop appropriate solutions. Thorough preliminary research provides a solid foundation for selecting the most suitable sensor, setting the project on a path to success.

Listed below are several existing sensor options that were identified for consideration, in no particular order. Those sensors that had the potential to fulfill the requirements of the project and contribute to the overall functionality are further described below.

1. Camera
2. RADAR
3. LiDAR
4. Active radio/WI-FI based positioning
5. Microphone array

6. Ultrasonic sensor
7. Infrared sensor
8. Photoelectric sensor

Camera

DS | SH

A camera is an optical device that captures and records images or videos. It consists of a lens, which focuses light onto a photosensitive surface such as a digital sensor or film. When an object is in front of the camera, light from the object enters through the lens and forms an image on the sensor or film.

Cameras are passive sensors, which rely on ambient light such that the photo detectors can detect the brightness. In digital cameras, the sensor converts the incoming light into electrical signals that are then processed and transformed into a digital image. The cameras processor adjusts various settings such as exposure, focus and color balance to optimize the captured image quality.

Other sensors and filters (or the lack of filters), such that the camera can detect the infrared spectrum also exist. This can be used as an active system, with an infrared transmitter and reflectors on the objects.

In contrast to a normal camera, a 360° camera is dependent on special lenses and/or multiple camera modules. To achieve the desired Field of View (FoV), there are fish eye lenses which are commonly used in combination with two or more camera modules.

Cameras also include other components like a shutter mechanism, which controls the duration of exposure to light, and an aperture, which regulates the amount of light entering the camera through the lens.

Distance estimation can also be accomplished using either active sensors or passive sensors like cameras. One common approach is to use a stereo vision system, which typically consists of two cameras to simulate human binocular vision. Utilizing a stereo vision system could potentially retrieve measuring data [25].

RADAR and LiDAR

SH

Radio Detection And Ranging, commonly known as RADAR is an active sensor that transmits and receives signals in the electromagnetic spectrum. As the name implies, this type of sensor is capable of *detecting* objects and acquire *ranging* information. The transmitter and receiver can use the same antenna, or consist of two separate antennas [26].

Light Detection And Ranging or LiDAR works in much the same way as the aforementioned RADAR, it however deploys infrared light with a shorter wavelength, for example 905 nm, resulting in a frequency seen in (7.1),

$$f = \frac{c}{\lambda} = \frac{3 \cdot 10^8 \text{ m/s}}{905 \text{ nm}} = 331.5 \text{ THz}, \tag{7.1}$$

where λ is the wavelength, f is the frequency and c is the speed of light.

In comparison, RADAR uses radio frequencies between 5 MHz and 300 GHz [26]. LiDAR utilizes eye-safe laser beams to create a 3D representation of the environment it is located in, providing accurate data for machines and computers. The sensor emits pulsed light waves that bounce off objects and return to the sensor. By measuring the time it takes for each pulse to return, the sensor calculates the distance traveled. This process is repeated rapidly to create a real-time 3D map, known as a point cloud. The point cloud can be further processed by a computer for various applications [27].

The ranging information can be determined by the time delay of the signals return and is described in (7.2),

$$R = \frac{c}{2}\tau_R, \tag{7.2}$$

where R is the range from the RADAR/LiDAR to the object [m], c is the speed of light [m/s] and τ_R is the time delay from the signal is sent, until it is returned [s] [26]. Depending on how the signals are transmitted, the ranging can be ambiguous if the return signal is detected after a new signal is transmitted by the sensor [26].

Wi-Fi based positioning system

DS

Wi-Fi based Positioning System (WPS) was explored as a possible solution by the last bachelors project, Project Mini Launcher [1], page 287.

Sound sensors

SH

A sound sensor can detect sound waves or specific frequencies, allowing for acoustic target detection or identification. It can be beneficial for detecting audio cues or specific acoustic signatures. For use with this type of sensor with project Mini Guardian, we would need an object that makes sound, for example a flying drone. This however would lock the project to that exact object, which could be a hazard for testing and showcasing it. Using a drone would not be possible as there are many restrictions of where and how it could be used.

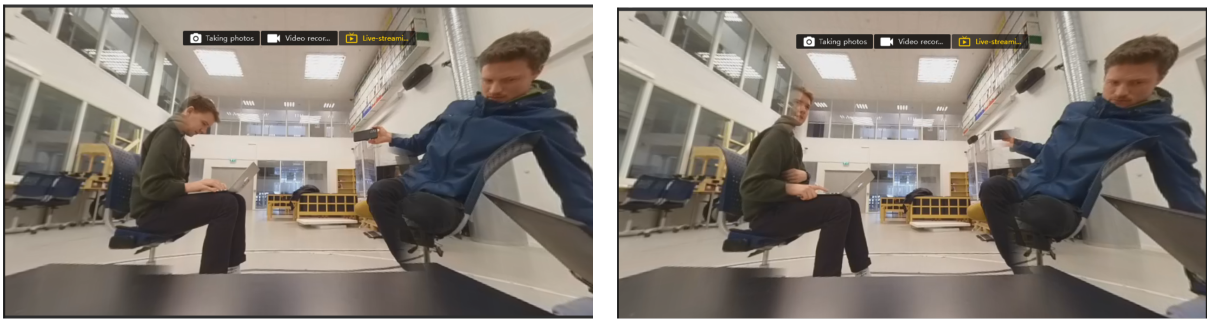
7.1.3 Test of sensors

There is a plethora of sensors which falls into the aforementioned sensing technologies, and even though the specifications can be compared theoretically, there is no substitute to a first hand experiment/test. It was therefore necessary to acquire a selection of sensors to test, two sensors are tested below.

MH

Insta360$^{\text{TM}}$ Pro camera

We were fortunate enough to borrow a 360° camera by our client; KDA. As the camera has surround vision this could be used for object detection. The camera is developed and produced by Insta360$^{\text{TM}}$ [28], and it has features such as six 200° fish-eye lenses with an aperture of f/2.4. This allows the camera to capture a 360° panoramic video with real time stitching in 4k resolution at 30 Frames Per Second (FPS). The camera can be connected in various ways, namely Insta360$^{\text{TM}}$ Farsight, Ethernet connection and Wi-Fi connection.



(a) Phone. (b) Missing phone.

Figure 7.3: Stitching.

While testing the Insta360$^{\text{TM}}$ Pro camera, where it connected via Ethernet, and controlled by the proprietary software Insta360$^{\text{TM}}$ Pro, this showed that the live feed had a high latency of 0.5-3 seconds. One of the consequences of the high latency was that it takes a second for movement in real life to appear on the computer screen depending on the camera mode. The camera documentation [28] also states that the FPS is maximum 30, but when using the camera 30 FPS was rarely achieved, it stabilized itself around $10 - 15$ FPS. As the camera consists of six 200° fish-eye lenses with an aperture of f/2.4, the stitching in the software makes it so that information is lost in the gaps between the cameras. This can be seen in fig. 7.3, in the fig. 7.3a the entire phone is displayed, whereas some parts of the phone are missing if it is moved in the picture as seen in fig. 7.3b. This is for close objects, further away less information is lost. The camera does not measure distance, which makes it difficult to provide information when aiming the Effector. Furthermore this makes it so that achieving the requirement SR-01-01 provided in Table 3.5 will become difficult.

Velodyne Puck$^{\text{TM}}$, the VLP-16 LiDAR

SH

Similarly to the Insta360$^{\text{TM}}$ Pro camera, we were fortunate enough to borrow a 360° surround LiDAR from Coastal Shark; a KDA summer project. The Velodyne LiDAR has software called VeloView, and when installed it displayed the capabilities of the LiDAR well. With the LiDAR scanning the room with 300'000 data points per second, it had a low latency meaning that movement in the physical world appeared within 100 ms. The VeloView software writes the information captured into a spreadsheet, it also has its own Python console integrated in the software. This means that the information written, a so-called point cloud, can be further processed to retrieve a single coordinate if wanted.

The Puck$^{\text{TM}}$ has a range of 100 m and a compact form factor. It scans across a 360° horizontal Field of View (FoV) and a 30° vertical FoV. The vertical FoV is ±15° above and below the horizon, so if placed on the ground it would have an effective FoV of 15°. It features 16 channels across the vertical FoV, which means that it has 16 lasers being sent out of the sensor equally spaced apart. The sensor is Class 1 eye-safe using 905 nm technology, which makes it a capability to be used around humans or other animals [29].

Fig. 7.4 shows a picture taken of the LiDAR feed in their included visual program VeloView [30]. It was placed inside our office space at Innovasjonsloftet i-114. It is a top-down view, and to the left of the image is the corridor. The corridor is separated with a glass wall which the LiDAR sees straight through. The top of the image has a wall-mounted TV, whilst the right side of the image has three people sitting at the table performing the test.

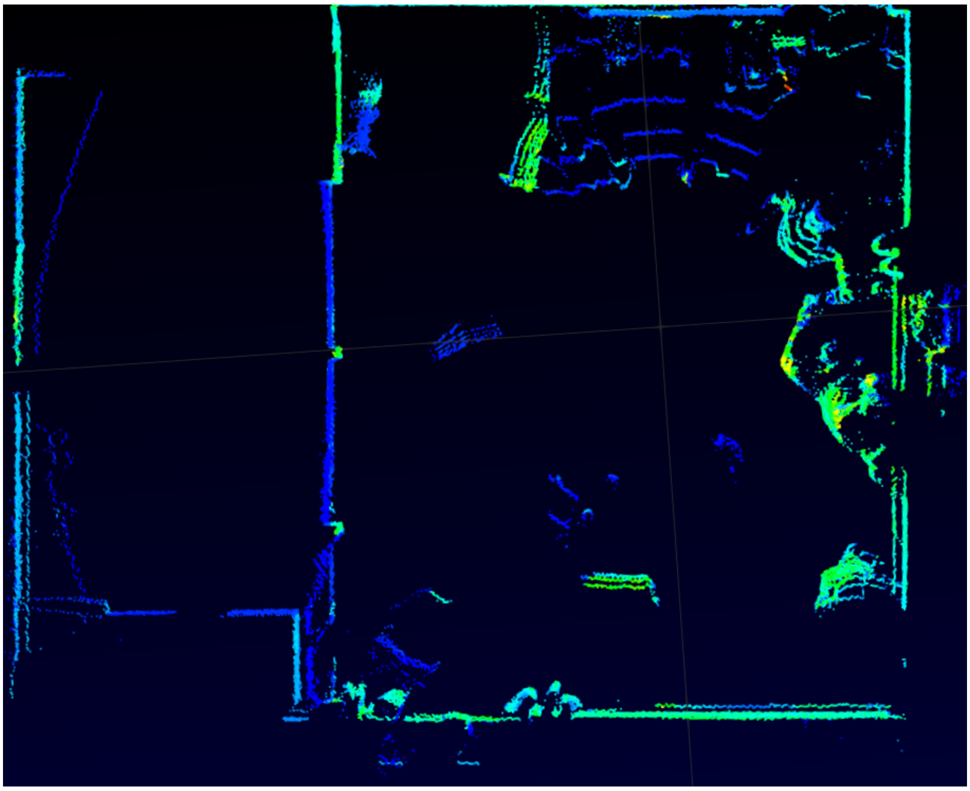


Figure 7.4: LiDAR capture of room, through Velodynes VeloView.

In Table 7.2, the relevant datasheet specifications are listed. As this sensor had potential for the project these were the characteristics considered in the selection process. We also had access to this sensor through the entire bachelors project.

Table 7.2: Sensor characteristics of the Velodyne Puck^{TM}.

Characteristic	**Specification**	**Requirement**
Range	100 m	SR-01-02
Identify	Can identify a silhouette	FR-02
Ranging information	Can retrieve exact coordinates	SR-01-01
Accuracy	±3 cm	SR-04-01
Sample time	Real-time updating	SR-01-05
Vertical FoV	From $-15°$ to $+15°$	SR-01-04
Horizontal FoV	360 °	SR-01-03
Safety	Class 1 eye-safe	NFR-02
Price	Medium	Budget
Enclosure	Can partially be enclosed	SR-11

7.1.4 Selection process

MH | SH

The selection of the systems sensor is crucial for this project as stated in The problem. Therefore, a considerable time was spent researching a decision making process that could ensure a decision as objectively as possible, while considering the different criteria derived from the requirements, see Table 7.1.

The decision making process is typically subjective, influenced by feelings based on experience and knowledge. To minimize the affection for a pre-chosen sensor from our subjective thoughts the decision making process Analytic Hierarchy Process (AHP) was the preferred model over other decision making processes such as Pugh Matrix. A major difference in the two models is that the AHP gives an inconsistency factor which gives a good indication of how objective the decisions were.

Analytic Hierarchy Process

A basic approach to decision making can be the Analytic Hierarchy Process also known as AHP. This form of decision making is designed to cope with both the intuitive and the rationale when choosing the most suited solution based on several criteria. During the selection process, the criteria being used are compared pairwise in a scoring process by the decision maker. This scoring is then used to show the overall rating of the criteria which are being used during the selection process. The AHP allows for inconsistency when judging, and also provides a way to improve consistency [31].

When structuring a decision selection, the simplest form is by setting up a hierarchy which consists of three levels. The top block in the hierarchy is the goal for the decision process. The second level in the hierarchy contains the criteria, which are used when selecting a solution, and are listed. And lastly in the third level the different alternatives also known as concepts are listed. This way of structuring the decision process makes it so that it is easier to cope with diversity. In short this way of organizing the decision in gradual steps form the general goal in the upper level, to the particular criteria and alternatives in the lower levels. The reason for doing so is to make it possible to decide the importance of the different elements in the hierarchy with respect to other elements or levels in the hierarchy. Once this structuring of the decision process is completed the rest of the AHP can begin [31].

Selection

AHP is a comprehensive and complicated model to set up from scratch. Therefore a pre-made program was used called SuperDecisions [32].

Because the AHP is complex, there are software packages developed for this purpose, an example is the SuperDecisions software which we used. This is a software that is developed as a cost free product by the Creative Decisions Foundation, and it has made the AHP decision making process a much easier method to use. The steps needed to reach a decision using the Super Decisions software follows the AHP way, and are summarized as follows [33]:

1. Create a hierarchical model for the decision: Set the decision taken into a hierarchy consisting of the goal, the criteria used and the alternatives.

2. Derive priorities for the criteria: Every criteria used to make the decision is compared pairwise with the desired goal in mind. This is to give each of the different criteria a weight. When doing so the consistency of the judgments is checked, in other words a continual review of the criteria weight is performed during the comparing. This is to ensure that there is a reasonable level of consistency in terms of proportionality and transitivity.

3. Compare the alternatives with the criteria used: This step follows the same procedure as the last, in the sense that every alternative is compared pairwise with respect to each criteria. The consistency is checked and adjusted if needed.

4. Synthesize the model: The software sums up all the alternatives as a weighted sum, making it so that the weight of each criterion is taken into account. This in turn established the overall priorities of the alternatives in the decision process. The alternative with the highest priority is seen as the best choice for the decision.

5. Perform analysis of the sensitivity: This step looks into how changing the weight of a criterion affects the result, making the rationale behind the result understandable.

6. Make a final decision: By using the synthesized result from the software and sensitivity analysis a decision can be made.

When using the SuperDecisions software the first step is to build the hierarchical model being used for making the decision. In the software each level of the hierarchy is called a 'Cluster', the standard procedure is to create three clusters and call them 'Goal', 'Criteria' and 'Alternatives'. Moving on the next step is to create the different elements within the Clusters, the software calls these 'Nodes'. When all the Clusters are made with their corresponding nodes it is time to connect the nodes in the hierarchy. This means that the node in the highest level of the hierarchy is connected with the relevant node(s) in the level below, and so on [33]. An example of a hierarchy of a decision is shown in fig. 7.5, the top level, known as the goal, has the Choose sensor as its Node. Furthermore the level below has the different criterion's used for the decision listed, and finally at the bottom level the different alternatives are listed. This is a short example of the way we structured our hierarchy for the decision.

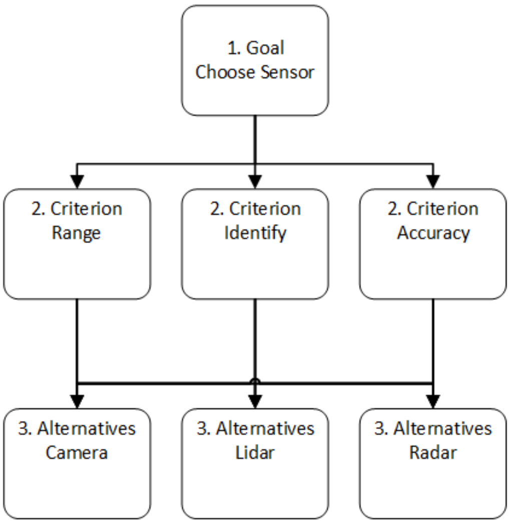


Figure 7.5: Hierarchy example used for AHP

After completing the hierarchy the next step is to derive the priorities of the criteria. This is because the different criteria used in the decision will have different importance. Doing this step ensures that the criteria will get its relative priorities when they are compared with each other. Here is an example of a car just to show, when buying a car we can have three criteria, such as cost, safety and comfort. For a student the most important criteria can be cost, compared to a parent where the most important criteria might be comfort or safety [33]. To compare the criteria with each other a numerical scale has been developed by the author of the method. The scale used when doing the pairwise comparison can be seen in Table 7.3. The scale has been validated for effectiveness, by using theoretical justification of what scale one must use when comparing homogeneous elements and also by the many applications used by a number of people. There can be many situations where elements are equally important or nearly equal when measuring, this means that the pairwise comparison determines how many times and of what fraction a given criteria is larger than another criteria [31].

Table 7.3: Numerical scale for AHP criteria scoring

Importance	Definition	Explanation
1	Equal importance	Two activities contribute equally to the objective
2	Weak	
3	Moderate importance	Experience and judgment slightly favor one activity over another
4	Moderate plus	
5	Strong importance	Experience and judgment strongly favor one activity over another
6	Strong plus	
7	Very strong or demonstrated importance	An activity is favored very strongly over another; its dominance demonstrated in practice
8	Very, very strong	
9	Extreme importance	The evidence favoring one activity over another is of the highest possible order of affirmation

7.1.5 Selection of sensor for project Mini Guardian

SH | DS | MH

The selection process for the Sensor of project Mini Guardian began with sorting out the different requirements that were associated with the Sensor. The relevant requirements and what Sensor criteria it led to are listed in Table 7.4.

Table 7.4: Criteria for Sensor from requirements.

Req. ID.	**Description**	**Sensor criteria**
SR-01-02	The Sensor shall detect objects within a distance 2.5±0.5 m	Range
FR-02	The system's Sensor shall identify and categorize the detected objects	Identify
SR-01-01	The Sensor shall measure distance	Ranging information
SR-04-01	The Effector shall hit marked targets with a diameter of 30±5 cm at a range of 2.5±0.5 m	Accuracy
SR-01-05	The Sensor should detect newly appeared or relocated objects within 5 s	Sample time
SR-01-04	The Sensor should detect objects within the Effectors vertical range (0 - 40°)	Vertical FoV
SR-01-03	The Sensor should detect objects from all azimuths	Horizontal FoV
NFR-02	The System shall not be a risk for health and safety	Safety
Budget	Budget limit of 20000kr, if more a pitch is needed	Price
SR-11	The Sensor should have an enclosure	Enclosure

The criteria were derived from the requirements into some measurable information that could be retrieved through e.g. datasheets for some different sensors that could be viable for the project. The main categories of sensors that are applicable and could fulfill the set criteria were radar sensors, LiDAR sensors and optical cameras. This was established in the Preliminary Research, see section 7.1.2. With the categories of viable sensors in place, the search for sensors began. Knowing how LiDARs from Velodyne worked through testing, see section 7.1.3, it was natural to pick a few different LiDAR options from them. Three of them had potential. Another LiDAR from Leishen was also added to the list. The Insta360$^{\text{TM}}$ Pro camera had also been tested and got added to the list as well as a high quality camera from Raspberry Pi. Radar is a potential health hazard, but thinking of the automotive and robotics industry some short range radars also had potential as they are not powerful enough to do bodily harm. One from Bosch and one from Arbe Robotics was added to the list. The different sensor datasheets were compared with each other according to the acquired criteria. See Table 7.5 and 7.6 for the data. More data was acquired from the datasheets but are not listed here as it was not deemed insightful.

Table 7.5: Comparing the different LiDAR relevant information through datasheets.

Requirement	Velodyne Puck LiDAR [29]	Velodyne Ultra Puck LiDAR [34]	Velodyne HDL-32E LiDAR [35]	Leishen C32W LiDAR [36]
Range	100 m	200 m	100 m	100 m
Identify	By silhouette	By silhouette	By silhouette	By silhouette
Ranging information	Yes	Yes	Yes	Yes
Accuracy	±3 cm	±3 cm	±2 cm	±2 cm
Sample time	Real-time updating	Real-time updating	Real-time updating	Real-time updating
Vertical FoV	$-15°$ to $+15°$	$-25°$ to $+15°$	$-30.67°$ to $+10.67°$	$-55°$ to $+14.9°$
Horizontal FoV	360°	360°	360°	360°
Safety	Class 1 Eye-safe	Class 1 Eye-safe	Class 1 Eye-safe	Class 1 Eye-safe
Price	Medium (on loan)	High	Constrained	$4995
Enclosure	Can partially be enclosed	Can partially be enclosed	Can partially be enclosed	Can partially be enclosed

Table 7.6: Comparing the different cameras and RADARs relevant information through datasheets.

Requirement	Insta360$^{\text{TM}}$ Pro camera [28]	Raspberry Pi HQ Camera [37]	Bosch Front Radar [38]	Arbe Robotics Radar [39]
Range	≈ 15 m	≈ 15 m, 25 m with zoom lens	210 m	36 m
Identify	By silhouette, color & sound	By silhouette & color	Potentially by silhouette	Potentially by silhouette
Ranging information	Potentially	Potentially	Yes	Yes
Accuracy	Unknown	Unknown	10 cm	9.5 cm
Sample time	0.5-3 s delay	Real-time updating	Real-time updating	Real-time updating
Vertical FoV	$-90°$ to $+90°$	45°	$-15°$ to $+15°$	$-15°$ to $+15°$
Horizontal FoV	360°	45°	$-60°$ to $+60°$	$-50°$ to $+50°$
Safety	No hazard	No hazard	See certifications	See certifications
Price	$3499 (on loan)	$242	Contacted for price	Contacted for price
Enclosure	Can partially be enclosed	Can partially be enclosed	Can be enclosed	Can be enclosed

After the data was acquired, we could start filling in the scoring for each individual specification from Table 7.5 and 7.6 up against the criteria listed in Table 7.4 in SuperDecisions. They were linked up as shown in fig. 7.6. The AHP includes a Consistency Index (CI). This index gives a pointer of how subjective the process was. An inconsistency factor of less than 10% is deemed an acceptable level of consistency and implies that the adjustment is small compared to the actual values of the eigenvector entries [31]. The ranking of each concept up against every criteria gave a total inconsistency factor of 0.09377 which corresponds to 9.377%. This indicates that the matrix has an acceptable level of consistency.

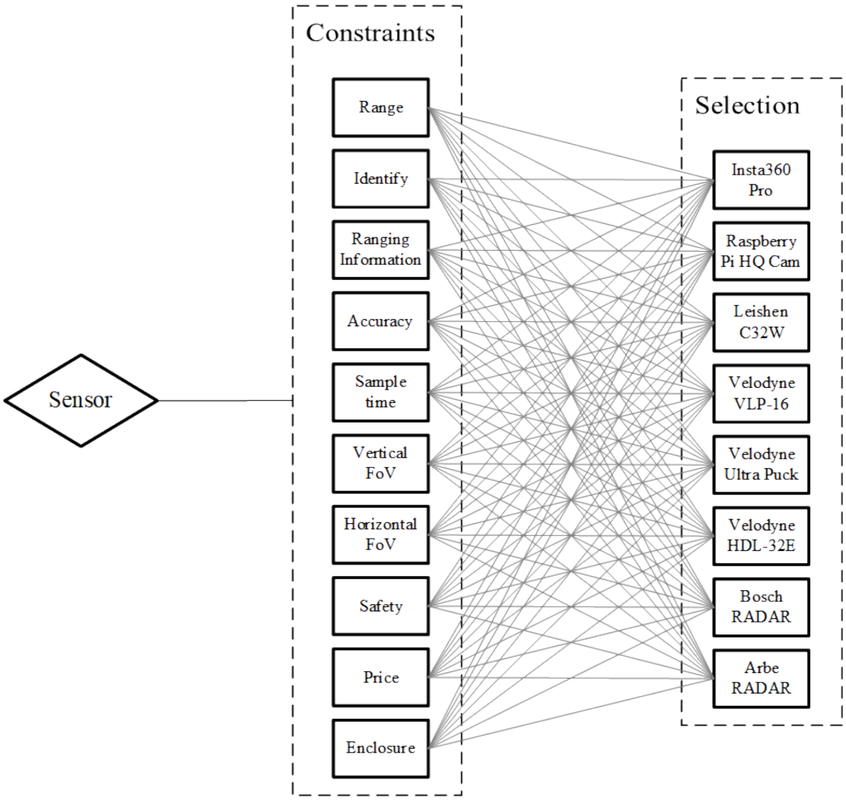
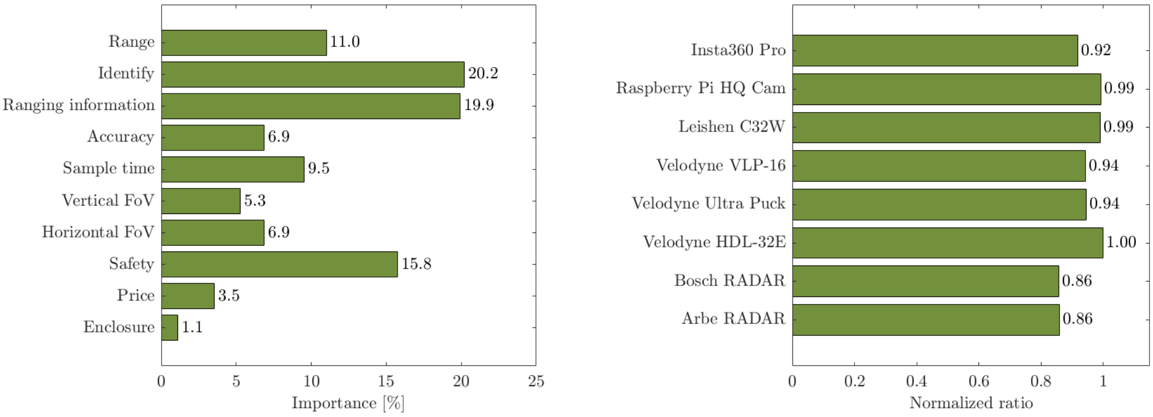


Figure 7.6: AHP setup for sensor selection.



(a) Constraints.

(b) Sensor selection.

Figure 7.7: AHP outputs.

Fig. 7.7a shows the requirements and how valued they are up against each other. The most important requirements and almost equally valued for the Sensor, is that it should be able to identify objects, and be able to give out ranging information, in the form of a location in coordinates. Safety is also a valued requirement.

As fig. 7.7b shows, there was no distinctive winner. The LiDAR from Velodyne HDL-32E won, but on a close second and third place we got the Raspberry Pi High Quality Camera concept and the LiDAR from Leishen C32W. All other concepts also ranked high, the RADAR concepts a bit lower than the rest. The similar ranking is due to the concepts being good at some things, but not all things. The camera concepts vastly outperform the LiDAR and RADAR concepts when it comes to identifying objects, which is a majorly valued criteria as shown in fig. 7.7a. On the other hand, a camera can't retrieve ranging information - it can't measure distance on its own - which is also a majorly valued criteria. This, the LiDAR can handle. Therefore the different concepts, especially the optical camera and LiDAR concepts score almost equally in the end. What does not emerge from the AHP, is that none of these concepts alone completely fulfill all the systems requirements.

Realizing this, we added a combination of the best singular concepts in combination with each other to the AHP. The best LiDAR and best optical camera concept were added. The RADAR concepts could have been excluded due to those concepts scoring lower than the LiDAR options as mentioned earlier, but are added for comparison. The results can be seen in fig. 7.8. This time there is a clear winner. The LiDAR and optical camera combination is distinguishably better.

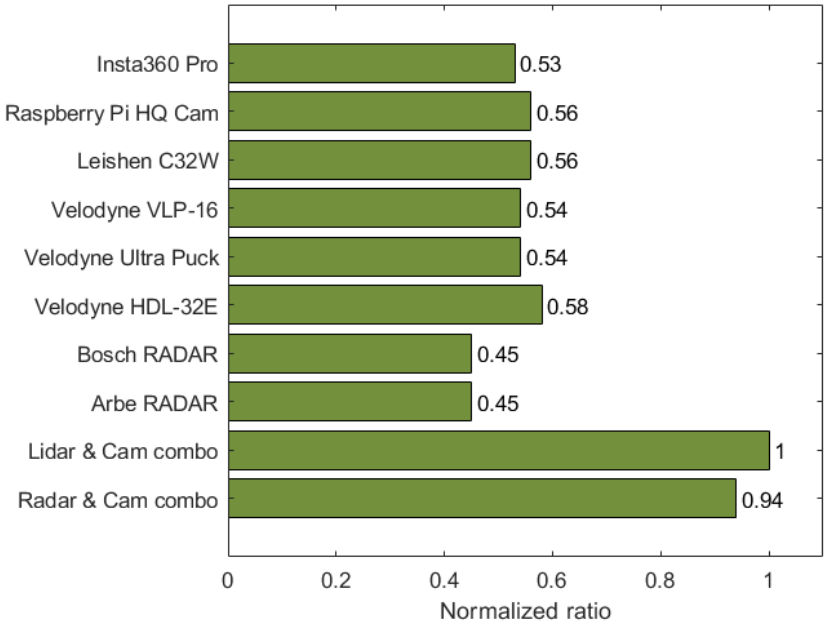


Figure 7.8: Sensor selection with combination of LiDAR & Camera and RADAR & Camera.

The LiDAR and optical camera combination is distinguishably better. The RADAR and camera combination could also solve The problem well, but it still scores lower which was expected due to earlier results. Therefore the Sensor option became a combination of the LiDAR and camera.

7.1.6 Ordering sensors

SH

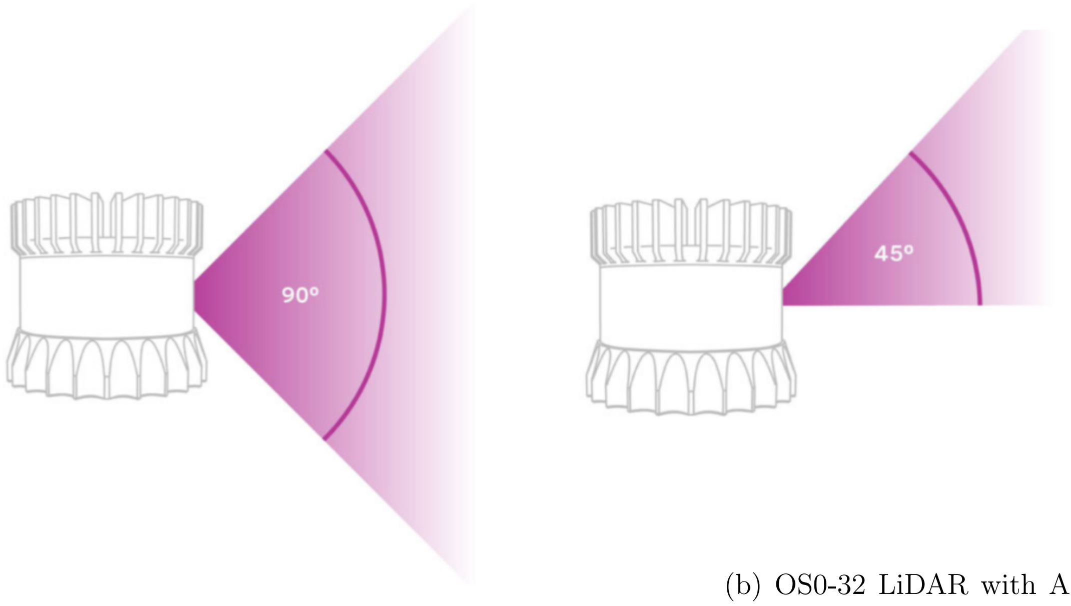
This section is included to provide an insight of why a lot of time was spent on potentially ordering a LiDAR, and the transition into a proof of concept.

Getting a price point of any of the LiDARs and radars turned out to be inherently difficult. The information online was almost non-existent. We found some price ranges for LiDARs up to $300'000, but that was years ago. LiDARs have advanced technologically, both in functionality and assembly. They should be much cheaper now, but it was difficult to know what price ranges different companies operated with at this time.

Therefore, during the selection process, we reached out to Velodyne (hoping to save time using the same brand of LiDAR as the one we are borrowing, so we could start some early developing before receiving our own LiDAR) through their contact form on their website the 18th of February. After a couple of days of not hearing anything from them we grew impatient and sent an e-mail directly to one of their associates the 20th of February. Heard nothing from them either and found out that Velodyne had merged with a company called Ouster on the 13th of February. Velodyne's website updated a few days after we tried to contact them. Therefore we reached out to Ouster instead the 27th of February. They answered within a day and we set up an e-meeting the 3rd of March. Spoke about what project Mini Guardian was and they spoke about selling us an equivalent LiDAR to Velodyne's LiDAR because they are discontinuing their products. Ouster LiDARs were all around better than Velodyne's LiDARs, they are digital instead of analog and had a better range of products, therefore it was unnecessary to update the AHP with new LiDAR specifications as the Ouster ones were superior to the ones compared either way. We tried to ask about the pricing. We did not get a price from them, but they promised a sales manager would send us a price listing. It took another six days before growing impatient yet again, asking for a follow-up on the pricing. We were then let know that the sales manager was out of-office until the 10th of March. The following day we received the price listing. The LiDAR with the best specifications for our project - OS0-32 - was $4000 without Value Added Tax (VAT). We asked some follow-up questions, some about technical specifications and some about the pricing and the sales manager asked for a phone meeting. This was held the 14th of March, where we got the price down to $3000 instead, but if we wanted the Above Horizon functionality, see fig. 7.9, it would cost $1000 more. The advantage of having an Above Horizon LiDAR is that all the channels are concentrated within the smaller FoV, which means that the angle between the channels is tighter which gives more accuracy. If we were to place the LiDAR on the same plane as the Effector the LiDAR would cover the entire vertical FoV of the Effector, and the channels below the horizon would be worthless to the project. Therefore Above Horizon functionality was preferred.

As we would blow up a budget (or two) purchasing OS0-32 the concept had to be pitched to our external supervisor, where she would take it further up the chain of command to get approval.

As this was in the middle of other final exams, time was limited to pitch it in a proper way. Originally it was an opportunity to pitch it directly to the client ourselves, but at this point the final exams were in focus. The facts were laid on the table, and the 23rd of March the project was let know that we would not receive the budget to purchase a new LiDAR. It would instead be an investment for next years bachelors project.



(a) OS0-32 LiDAR with Uniform FoV.

(b) OS0-32 LiDAR with Above Horizon FoV.

Figure 7.9: OS0-32 LiDAR with different FoV settings.

Ouster was updated by the project, and as a last effort to persuade the client we were offered the LiDAR with above horizon functioning for a total of $3000. This was also rejected by the client.

Thankfully we have access to another LiDAR borrowed from a summer project called Coastal Shark - also from the IDS department inside KDA. Therefore we have access to a Velodyne VLP-16, also known as the Puck™. With that in place we can make the project a proof of concept. It will function as intended, but not in an optimal way as originally wanted, so the system requirements were revised and updated to adapt over to a proof of concept.

7.2 The Sensor

SH

The selected sensors chosen was the Ouster OS0-32 and the AXIS M5074 PTZ Camera. This sensor solution is known in this project as the Sensor. The AXIS M5074 PTZ Camera was chosen over the camera option winner, the Raspberry Pi HQ camera, due to limited time left as a consequence of the sensor purchasing described in section 7.1.6. The Raspberry Pi HQ Camera would need a self-built rotating system to be able to record the 360° surrounding of the Effector. This section will describe the two sensors, and a caveat which led to the project using another LiDAR than what the AHP showed, due to budgetary limits. This also led to the project transitioning into a proof of concept.

7.2.1 OS0-32 LiDAR

SH

As Velodyne merged with Ouster, project Mini Guardian was recommended the OS0-32 LiDAR. This LiDAR is digital instead of analog, which Velodyne's sensors are. Digital

LiDARs are superior to the analog ones and Ouster are therefore discontinuing all of Velodyne's LiDARs. The OS0-32 has as its name implies 32 channels. Having the uniform channel beam spacing means that between each channel there is 90°/32 channels = 2.81° between each channel. Quite a bit between each beam as one goes further away from the sensor as will be shown in the next section; Velodyne VLP-16 LiDAR. With the Above Horizon functionality described in Ordering sensors, the angular resolution will be 45°/32 channels = 1.41° between each channel which would result in a potential of object detection at a further distance. There are also OS0-64 and OS0-128 options which respectively have 64 and 128 channels. At its best that means 0.35° between the channels with Above Horizon configuration, but of course more expensive, $12000 for the latter mentioned.

One of the digital advantages are more datapoints per second. The Puck$^{\text{TM}}$ has 300'000 datapoints per second with a possiblity to get 600'000 datapoints per second, whilst the OS0-32 has 1'310'720 points per second with a possibility up to 2,621,440 points per second.

In Table 7.7 it is specified to have a range up to 15 m. This is the range when the object has a 10 % Lambertian reflectivity with a 90 % detection probability. The maximum representable range is approximately 50 m. Other specifications are listed in the same table. Comparing these specifications to the other LiDARs, the OS0-32 is superior to all of them. This means that it would come victorious through the AHP either way, as another LiDAR with worse specifications already won the selection. Therefore there is no need to revise the AHP with an updated sensor specification [2].

Table 7.7: Sensor characteristics of the OS0-32 [2].

Characteristic	**Specification**	**Requirement**
Range	15 m	SR-01-02
Identify	Can identify a silhouette	FR-02
Ranging information	Can retrieve exact coordinates	SR-01-01
Accuracy	±1 cm	SR-04-01
Sample time	Real-time updating	SR-01-05
Vertical FoV	From −45° to +45°	SR-01-04
Horizontal FoV	360 °	SR-01-03
Safety	Class 1 eye-safe	NFR-02
Price	Medium	Budget
Enclosure	Can partially be enclosed	SR-11

7.2.2 Velodyne VLP-16 LiDAR

SH

As this section already has been described in Velodyne Puck$^{\text{TM}}$, the VLP-16 LiDAR, it will be recapped here. The specifications of this sensor is in Table 7.2. Among others it has a:

- Vertical FoV of ±15°
- Accuracy of ±3 cm
- Range of 100 m

Most importantly the project is borrowing it from Coastal Shark. That means we have unrestricted access to it throughout the duration of the bachelors project, which has shown itself to be very valuable. Ultimately, the OS0-32 was not purchased. We do therefore not have it available for project Mini Guardian, which leads into the next section, the Proof of concept.

Proof of concept

SH

This section describes why project Mini Guardian formed into a proof of concept. Most notably from the section Velodyne VLP-16 LiDAR, the VLP-16 has a vertical FoV of 30°, 15° above and below the horizon uniformly distributed. With 16 channels, this means $30°/16 = 1.875°$ between each channel. This is shown in fig. 7.10.

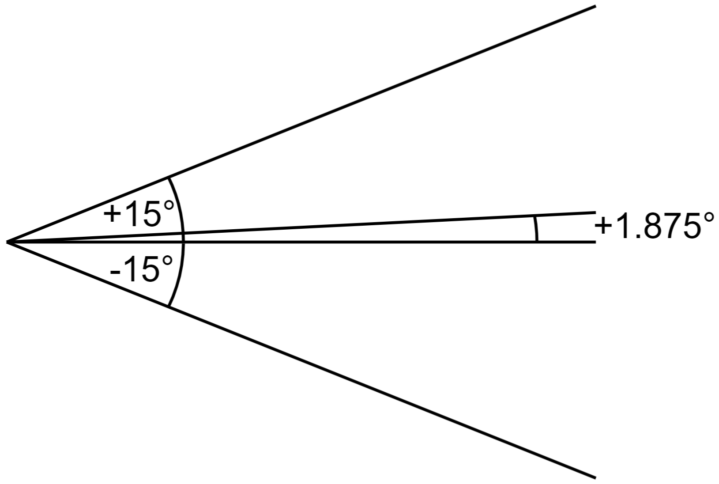


Figure 7.10: VLP-16 LiDAR FoV.

Although the sensor has a capability of viewing a range of up to 100 m, detecting the objects for this project the range is drastically reduced. The vertical distance is calculated with (7.3),

$$y = x \tan \alpha, \tag{7.3}$$

where y is the distance between each channel at a distance of x with the angle α. This leads to a a vertical distance y at $x = 5$ m to be 16.4 cm between each channel. With balloons as objects being detected, this will lead to one channel going across the balloon, perhaps two channels at best. Getting any noteworthy data out of this is difficult. At a distance of $x = 3$ m, the vertical distance $y = 9.8$ cm between the channels. At this distance theoretically two to three channels will go across a balloon. In a practical test this was confirmed, as can be seen in fig. 8.9. This led to a heavy reduction of the original system requirements, making project Mini Guardian into a proof of concept, showcasing the potential of the System in such a way that the client will put in an investment for further development at a later stage with a renewed budget.

7.2.3 AXIS M5074 PTZ Camera

JM

The decision to change the camera option was made after careful consideration of the project timeline and the need to deliver the system on time without compromising on quality. There has been invested a significant amount of time in researching and selecting a sensor that would meet the client's requirements and the process was started accordingly. However, as the deadline for the project approached, it became clear that we needed to make a change to the camera option in order to deliver the system on time.

The AXIS M5074 Pan-Tilt-Zoom (PTZ) camera is a small, discreet camera that delivers high-quality video with its 5x optical zoom and 12x digital zoom. With its PTZ functionality, it provides overviews and allows for the zooming in to get the details. It has built-in cybersecurity features that help prevent unauthorized access and safeguard our system. The camera also includes autofocus and WDR features.

- High-resolution imaging: The camera is capable of capturing high-quality images with a resolution of up to 1080p, which can be helpful for detecting balloons in the sky [40].
- Pan-tilt-zoom functionality: This camera has PTZ function, which allows it to move in various directions and zoom in on specific areas of interest. This can be beneficial for tracking the movement of balloons and capturing close-up images for identification purposes [40].
- Dome Design: The camera's dome design allows it to capture a wide field of view, making it ideal for monitoring large areas. This feature is particularly useful where a wide field of view is essential [40].
- Weather-resistant design: The camera has a weather-resistant design, which can be beneficial for detecting balloons in outdoor environments, where weather conditions may be unpredictable [40].
- Audio: The camera also includes audio capabilities, such as one-way audio streaming and built-in microphone. It supports several audio encoding formats, with configurable bit rates, which can be useful for further development [40].

While the decision to change the product option was not taken lightly, it was necessary to ensure that we delivered a high-quality system to the client on time. We believe that the alternative option we chose will still meet the client's requirements and deliver value while addressing the production constraints.

7.3 Scenario

SH

In this scenario, the primary objective of project Mini Guardian is to achieve a functional air defense system capable of detecting and engaging targets. The system consists of several components strategically placed in the Enclosure.

Placed at the bottom of the Enclosure is the Effector, which serves as the firing mechanism for the projectiles. Straight above the Effector, on the same z-axis of the Effector, the AXIS camera is positioned to provide visual identification of objects. It can be turned 360 ° to cover every position around itself within a few seconds. Above the AXIS camera, also on the same z-axis, follows the LiDAR sensor, which detects objects in its surroundings in real-time, up to 360 ° around the Enclosure. The setup is shown in fig. 7.11.

To demonstrate the Systems capabilities, a balloon is placed in the surroundings of the Enclosure. The LiDAR immediately detects the balloon and marks it with a square in the Systems interface. At the same time the LiDAR measures the center coordinate of the detected balloon. The AXIS camera can be pointed towards the detected balloon and can identify it as either Hostile, Friendly or Unknown using a color detection algorithm. This camera feed is also sent to the GUI for the operator to visually confirm the target.

Using the GUI, the operator can interact with the System. The GUI allows the operator to control and coordinate the Systems functions. By pressing "Start automatic aim" in the GUI, the Effector will be pointed towards the detected target using the coordinate sent by the LiDAR.

If the balloon is green, it will be identified as Friendly. If the balloon is any other color than green or red, the balloon will be classified as Unknown. In any of these cases, the Effector will not be able to fire as the firing buttons are disabled in the GUI.

If the balloon is red however, it will be identified as Hostile. After hitting "Start automatic aim" and waiting for the Effector to aim at the target, the firing buttons are activated and the operator can proceed to take action. Any of the GUI buttons to fire each respective canister can be pressed. If pressed, the Effector will fire the projectile using air pressure to eliminate the target.

This scenario illustrates the objective of the Mini Guardian project, showcasing the Systems ability to detect, identify, and engage targets using a combination of the LiDAR, AXIS camera and the Effector controlled through the GUI.

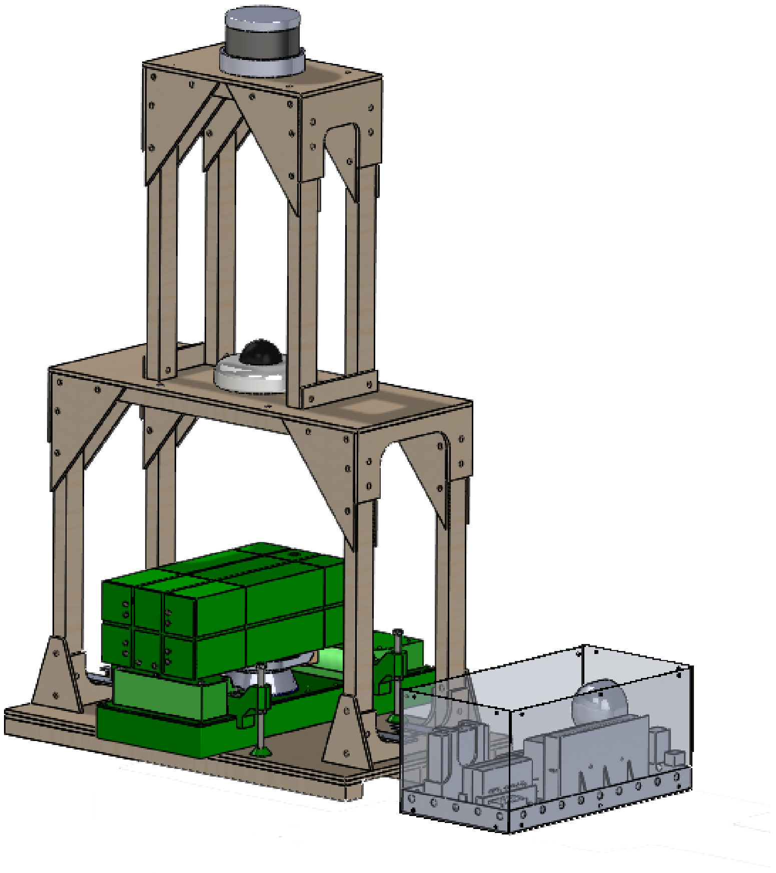


Figure 7.11: The total System in its Enclosure.

Chapter 8

Software

JM | VJ

This chapter focuses on software development and specifically discusses object detection and identification. It provides an overview of various software tools and algorithms used for object detection, including Robot Operating System (ROS), TensorFlow, OpenCV, PyTorch, and YOLOv5. The chapter also explains their functionalities, applications, and advantages.

As this project is a further development on previous years project [1], this chapter also concerns the software improvements and additions to the existing software.

Overall, this chapter provides an exploration of object detection software, training processes, and device communication, showcasing the tools, algorithms, challenges, and ongoing work related to the project.

8.1 Software Development

8.1.1 Use Case

VJ | JM

A use case is a description of interactions or functionality of a system seen from a users perspective. It describes the steps involved, the actors involved, and the expected results to achieve a particular objective. Use cases help define the behavior of the system and guide the development and testing process [41].

As the software architecture provided from previous year [1] was not optimal, the provided use case had similar issues. The sensor system were added as well as adapting the use case for the project. The updated use case can be seen in fig. 8.1.

- **Fire Projectiles:** Firing projectiles is an "extend" to the system control as it shall be a controlled event only triggered when a Hostile target is identified.
- **Motor Control:** The system controls the motors related to rotation and elevation.
- **Detection:** While the system is active, the sensor shall scan its environment continuously.
- **Categorization:** While the detection is continuous, the categorization shall only occur when an object being detected, making it an extend to the system control.

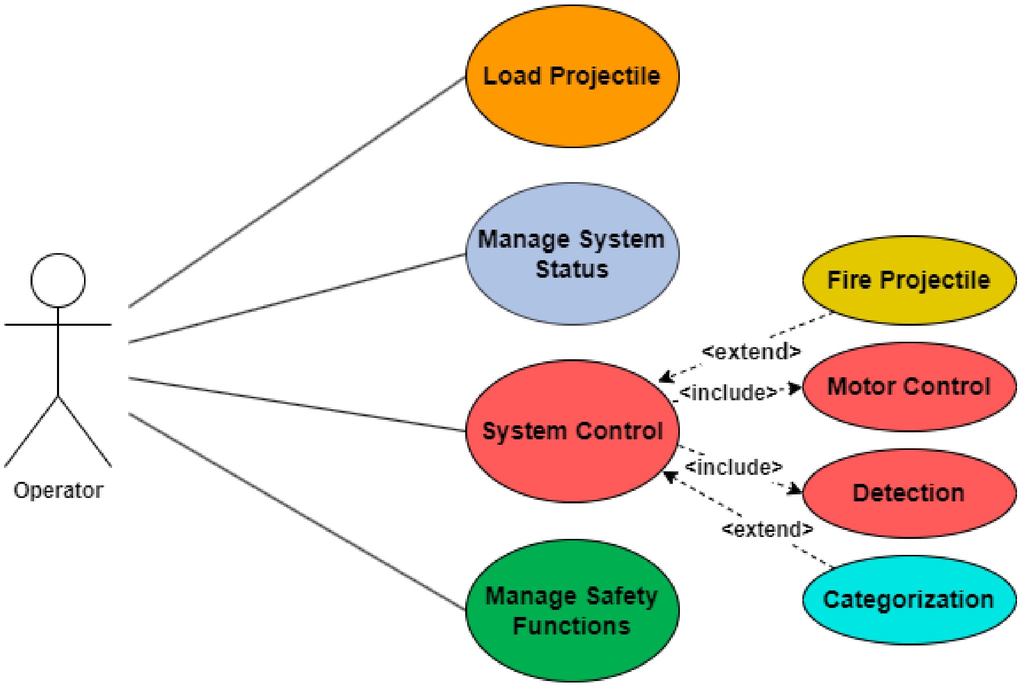


Figure 8.1: Use Case diagram

8.1.2 Software Architecture

VJ | JM

The software architecture provided from the previous year [1] had some flaws and did not correspond to the architecture for the final system. Making a new software architecture meant understanding the previous system as well as implementing the new concept, which was incorporating a LiDAR sensor and an AXIS camera. The architecture is included to bring structure as well as an overall understanding of the system and its behavior.

- **Interface:** There is both a physical and a digital interface as the system has physical buttons and switches, while there is also a GUI. The system offers both physical and digital interfaces, including physical buttons and switches as well as a GUI.
- **Manage System Status:** The systems status which includes Ethernet and serial status is updated with inputs from the system interface.
- **Manage Safety Functions:** Both a physical safety switch and safety functions in the GUI manages safety functions with inputs from the system interface.
- **Load Projectile:** A part of the physical interface as loading projectiles is a manual task and is not automated.
- **System Control:** The control of the system, including firing projectiles, detection, categorization and controlling the motors for rotation and elevation.

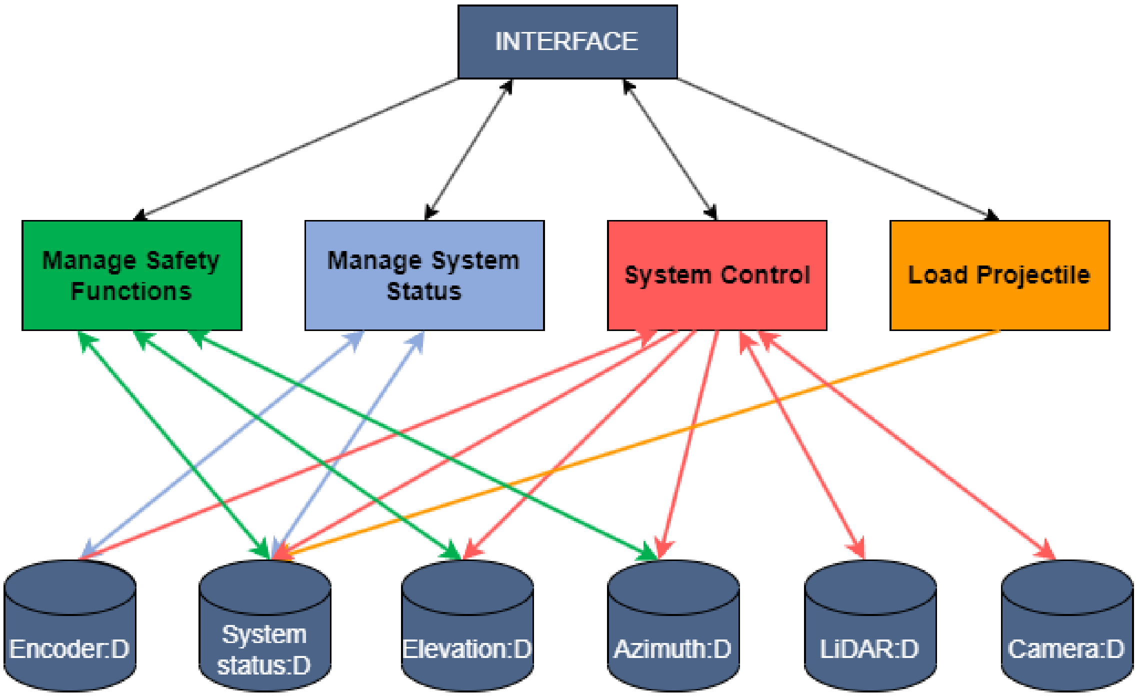


Figure 8.2: Software Architecture

8.1.3 Sequence Diagrams

JM | VJ

In this subsection the different sequence diagrams are shown in fig. 8.3 through 8.8. A sequence diagram is a visual representation in software development that shows the interactions and order of events between different components or actors in a system. It helps to understand how objects or actors communicate by displaying the sequence of messages exchanged over time. It's a valuable tool for designing and documenting system behavior in a concise and understandable way [41].

- User Interface (UI) represents the part of the system that interacts with the operator, it allows them to input information, view the outputs, and interact with the application. In a sequence diagram, the UI actor typically initiates the interaction by sending messages to other components or actors in the system [41].

- C represents the Controller responsible for coordinating the flow of information between the UI and components in the system. The Controller processes user inputs, calls the appropriate action, and manages the overall behavior of the system [41].

- D stands for both Dependency and Data. It represent the external dependency and data source that the system interacts with. This includes external services and APIs that the system needs to communicate with to fulfill its functionality [41].

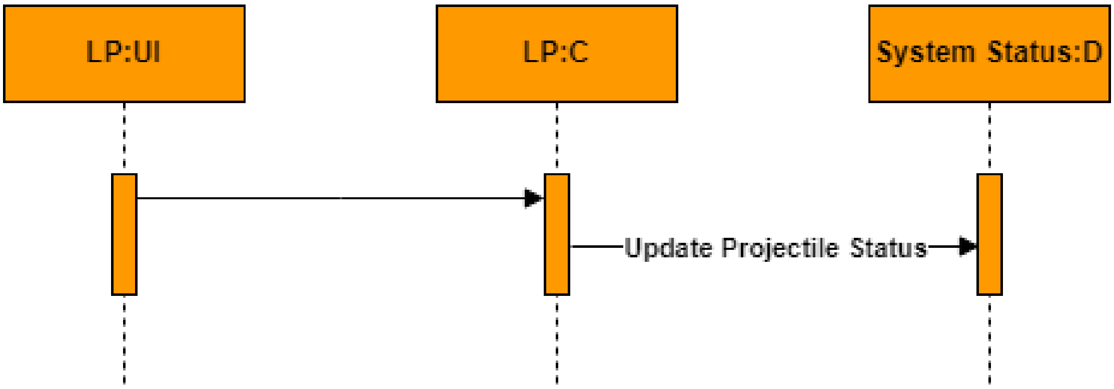


Figure 8.3: Load Projectiles

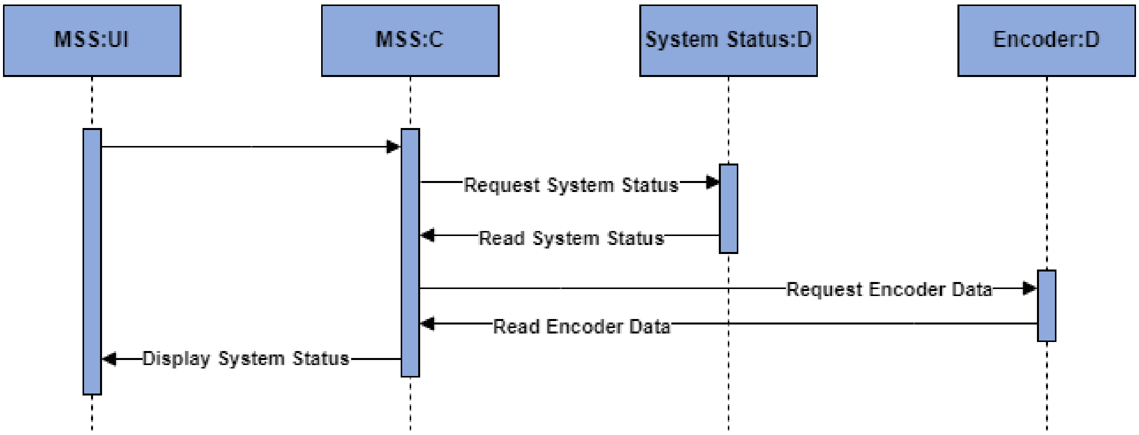


Figure 8.4: Manage System Status

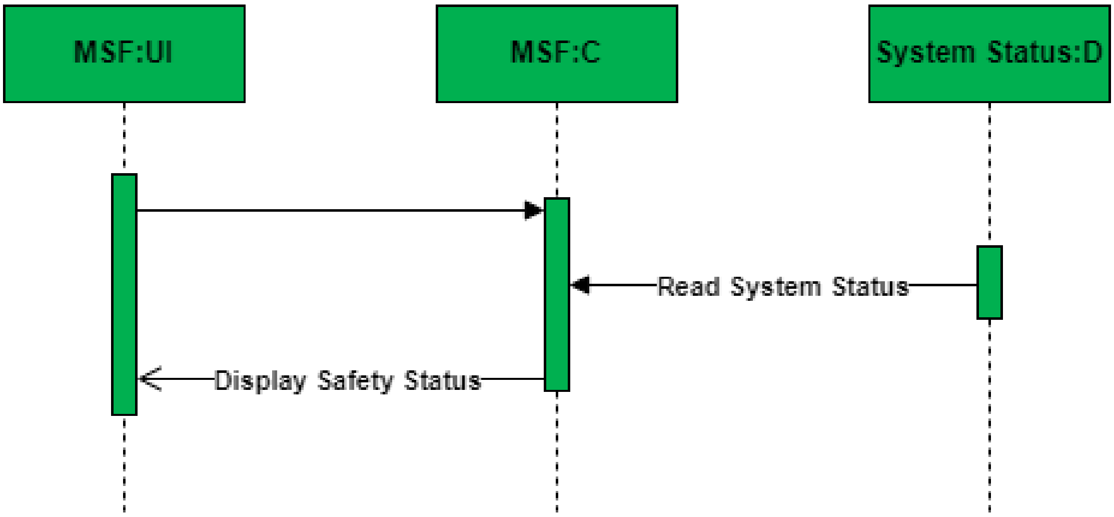


Figure 8.5: Manage Safety Functions

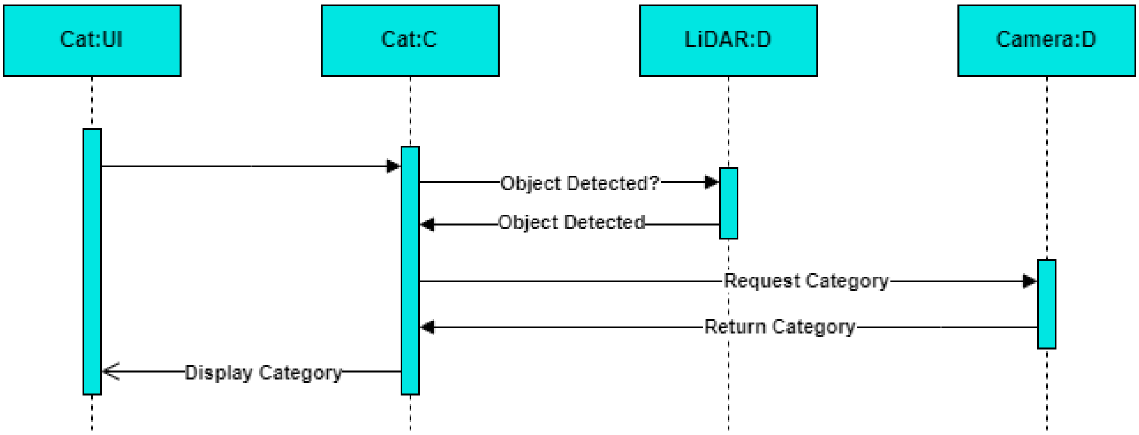


Figure 8.6: Categorization

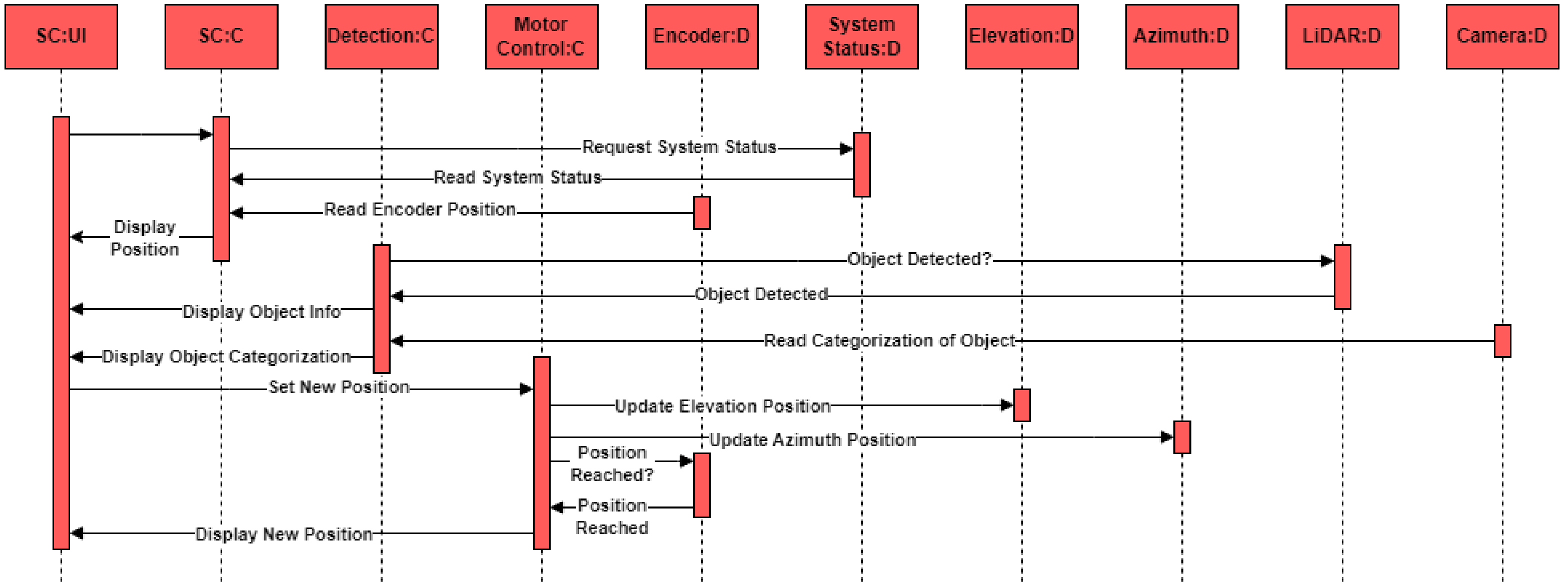


Figure 8.7: System Control, Detection and Motor Control

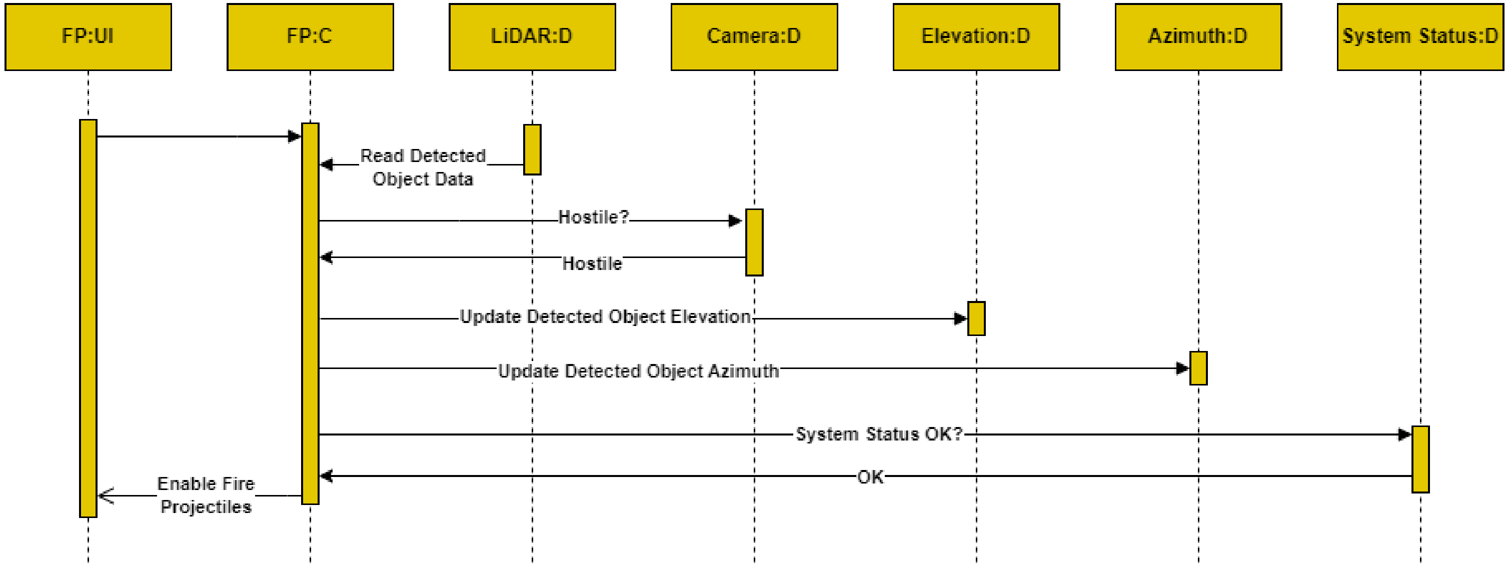


Figure 8.8: Fire Projectiles

8.2 Object Detection and Identification

VJ

Incorporating the sensor system into the previous system included an implementation of object detection and identification in order to fulfill several requirements. This section provides an understanding of the algorithms, codes and methods used for the detection and identification of objects in this project. To detect and identify objects a LiDAR sensor and an AXIS camera has been used. Considering the large scale and scope of the object detection software, this chapter only touch on the surface of the subject.

8.2.1 LiDAR for Detection

VJ

Requirement FR-01 states that *"The systems Sensor shall detect objects"*. A Velodyne VLP-16 LiDAR is employed to achieve the requirement, providing a 360-degree view of the surrounding area. The sensor alone is however unable to detect objects without extensive programming. Accordingly, the LiDAR has been programmed making it possible to detect balloons within a range of 3 m.

Being able to read and process the data output by the LiDAR required extensive research in order to find an appropriate solution, given the unfamiliarity of working with a LiDAR. A visualization application called VeloView was explored as a suitable solution for visualizing and processing the data, which can be seen in Appendix D. However, the unavailability of real-time data access and utilization were one of the key factors that led to the decision to explore alternative options discussed in 8.5.2. The final processing tool was ROS, and the code written for object detection is written in C++ .

The object detection code is a lightly modified version of [42] to fit the project requirements. This code detects objects of all sizes in the surrounding environment, while the project specifies on detecting balloons. The code also lacks the capability to output the coordinates of the detected objects, so an implementation was required for transmitting correct data and corresponding coordinates to the Effector.

The surrounding area of the LiDAR are scanned and plotted with points that make up a point cloud. The point cloud corresponds to the environment within the LiDAR FoV of 15 ° below and above the horizontal axis. As the Effector has a vertical FoV of 0-40 °, all points plotted below the Effectors FoV is considered out of range. The AXIS camera categorizing and identifying the detected objects has a range restricted to 0.5 m above ground, which results in objects below the camera range also considered out of range.

Given this information, there are parameters set in the code which only allows points within the Effectors and cameras FoV within 3 m. The LiDAR is placed on an Enclosure 0.8 m above the Effectors top structure, and as mentioned in 10.3.5, 0.5 m above the camera, and given this information the parameters for detecting objects are set to above -0.5 m in the z-axis.

The Euclidean clustering method is employed for the object detection method, meaning that a cluster is considered an object based on the Euclidean distance between the plotted points [43].

Given that the objective of the system is to detect balloons, a procedure was carried out to estimate the number of points associated with a balloon. Upon measurement, an object is determined to possess a significantly higher density of points within its immediate proximity. An object positioned at a greater distance will exhibit a reduced number of points, requiring a significant range between the minimum and maximum number of points that make up a specific object. As seen in fig. 8.9, a balloon in proximity to the LiDAR carries a higher point density than the balloon detected at a greater distance.

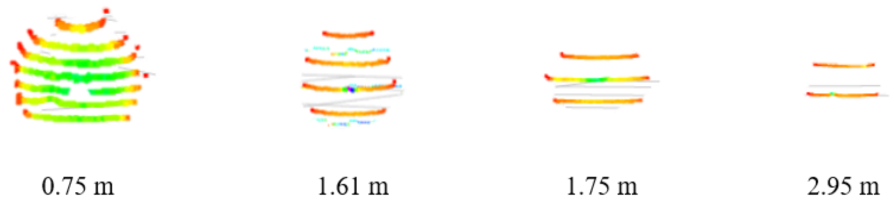


Figure 8.9: Balloon detected from different distances

The object detection code utilized for the LiDAR concerns point density in the clusters detected. Focusing on the shape of the objects was not a requirement for the LiDAR considering the cameras categorization and identification of the detected objects.

As objects categorized as Hostile would be fired upon, a code checking if objects are in frame was implemented. The system thus check if any objects remain present in frame after firing. If no objects are present, it could be inferred that the target was successfully hit.

For further code relating to the object detecting, please refer to the Doxygen folder.

Robot Operating System

VJ

Robotic systems has common denominators such as actuators, sensors and a control system. The actuators which perform an action corresponds to the Effector of the system, moving to a set direction and firing upon Hostile targets when receiving instructions. The system has a sensor system comprising an AXIS camera, a LiDAR VLP-16 sensor and a control system in the GUI. The system corresponds to a robotic system and therefore Robot Operating System (ROS) is a suitable solution. ROS is a set of software libraries and tools, and utilizing it as a tool to seamlessly integrate the LiDAR sensor into the system leads to a system that performs its task successfully [44]. Due to the amount of documentation available for ROS version 1 as well as resources, it became the best solution for the project, given the constrains of time. When referring to ROS further in the thesis, version 1 is the one referred to.

ROS allows for tools such as messages and topics that connects the components and layers of data easily. In the ROS framework, it is essential to create a master before initiating any ROS actions. It acts like a server for the nodes so that they can reach each others information. The communication is enabled through nodes being launched. When nodes are launched, the messages sent are published to a topic which is available for other nodes to subscribe to. This way, the nodes communicate by publishing and subscribing to messages and topics. Fig. 8.10 demonstrates ROS's communication properties.

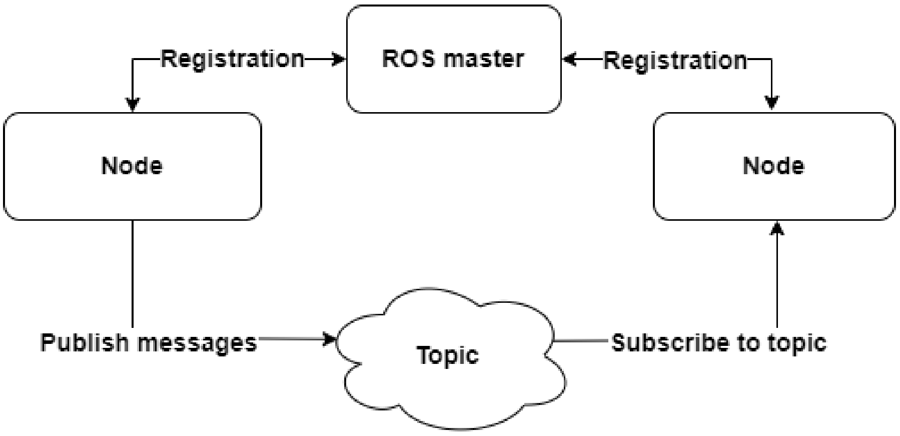


Figure 8.10: ROS communication

RViz The ROS framework is used to process the LiDAR data and has useful tools for visualization of the captured point cloud. RViz is a 3D visualization tool for the framework and is utilized for the project [45]. The topics in ROS can be added to RViz and visualizes the output of the topics. If the captured point cloud is filtered and published to the topics, a filtered point cloud is visualized in RViz. This enables the opportunity to remove unnecessary points in the cloud and visualizing the desired data. For this project, the object detection code is filtered, meaning that certain points within certain parameters are removed from the processed point cloud. The object detection code in ROS filters the point cloud captured and publish markers in RViz where the objects are detected.

8.2.2 Camera for Identification

JM

In order to fulfill functional requirements FR-01 which states that *"The system's Sensor shall detect objects"* and FR-02 which states that *"The system's Sensor shall identify and categorize the detected objects"*, various options were considered. This included object categorization using either a camera or a LiDAR sensor, each requiring different approaches depending on the chosen method. Utilizing a camera could involve using pre-existing libraries, frameworks and algorithms such as OpenCV [46], Tensorflow [47], YOLOv5 [48] and PyTorch [49] to perform the categorization. In contrast, as mentioned in the previous Chapter 8.2.1, object categorization with LiDAR involves utilizing point cloud data obtained from the sensor to identify and classify objects within the environment. During the research Tensorflow, OpenCV, YOLOv5 and PyTorch was explored for the purpose of object categorization.

TensorFlow, OpenCV, YOLOv5 and PyTorch are all popular to use in machine learning and computer vision applications, but they have different focuses and purposes.

TensorFlow is a powerful and flexible open-source platform for building and training machine learning models, primarily for deep learning. It is often used for tasks such as image and speech recognition, natural language processing, and other types of pattern recognition. TensorFlow provides a variety of high-level APIs for building and training models, as well as low-level APIs for more advanced users who want to customize their models [47].

OpenCV, is an open-source computer vision library that provides tools for analyzing and manipulating visual data, primarily images and videos. OpenCV provides a wide range of functions for tasks such as image filtering, feature detection, object tracking, and machine learning-based object recognition. It is often used in applications such as robotics, surveillance, and augmented reality [46].

YOLOv5 is a specific object detection algorithm developed by Ultralytics [48]. It is designed to identify objects in an image or video and locate them with bounding boxes. YOLOv5 uses a deep neural network architecture to detect objects in real-time, with high accuracy and efficiency. YOLOv5 is often used for tasks such as object detection and tracking in surveillance systems, autonomous vehicles, and robotics [48].

PyTorch is an open-source framework for machine learning that focuses on deep learning tasks. It falls under the category of deep learning frameworks, similar to TensorFlow. PyTorch provides a dynamic computational graph and a Python [50] interface, making it popular for building custom deep learning models.

To summarize, OpenCV is a library for image processing, TensorFlow and PyTorch are deep learning frameworks for building models, and YOLOv5 is a specific object detection algorithm for identifying and locating objects in an image or video.

Given that the existing system is built on the Qt framework [51], it would be more convenient to employ a combination of OpenCV and YOLOv5 for the necessary categorization task. OpenCV is a well-established computer vision library that is specifically designed for such tasks and integrates seamlessly with Qt [52]. In terms of real-time performance, simplicity, compatibility, and integration with Qt, OpenCV presents several advantages over TensorFlow. Which means that a combination of YOLOv5 and OpenCV would be the optimal choice for this project.

8.2.3 Training Object Detection Models

JM

Training object detection models is the process of teaching a computer vision algorithm or model to identify and locate objects within images or videos. Object detection is a computer vision task that involves two main steps: identifying the presence of objects in an image and determining their precise locations by drawing bounding boxes around them [53]. These steps were taken when training the model:

- Creating a dataset: A labeled dataset was created that contains images with annotations specifying the objects' locations. The annotations consist of bounding boxes around the objects of interest.

- Model selection: A suitable object detection model architecture was considered. The two considerations was Single Shot Multibox Detector (SSD) and You Only Look Once (YOLO).

- Preprocessing: Resizing the images, normalizing pixel values, and augmenting the data. This was done with Roboflow which will be discussed later on.

- Training process: The training dataset was fed through the model, which produced predictions for each image.

- Evaluation: Evaluated the model's performance on a validation set to monitor its progress during training. mean Average Precision (mAP), Precision and Recall metrics was obtained. This will be discussed at a later point.

The first training process involved using Tensorflow. Initially, training a model with Tensorflow using the Single Shot Multibox Detector (SSD) MobileNet V2 FPNLite 640x640 architecture [54] was selected as part of the research. The training was done following a YouTube tutorial [55] However, using PyTorch with YOLOv5 to train an object detection model for balloons was preferred over TensorFlow with SSD MobileNet V2 FPNLite due to higher accuracy, customization flexibility, a lot of resources for faster experimentation and the growing popularity of PyTorch. This sub chapter will provide an overview of the training procedures used for both models and the outcomes achieved.

Tensorflow

The TensorFlow Object Detection API presents pre-trained models for object detection tasks and enables users to train customized models utilizing their data sets. A custom model was trained in our research with images of balloons using Python and Jupyter Notebook [56]. The objective was to identify the most appropriate methodology for our project. During the training process, we discovered that Tensorflow was not suitable for our project due to its reliance on Python and Jupyter Notebook, which posed challenges for integrating it with our existing code written in C++. Furthermore, although Tensorflow demonstrated accurate object detection in images, as depicted in fig. 8.11, its real-time performance was suboptimal.

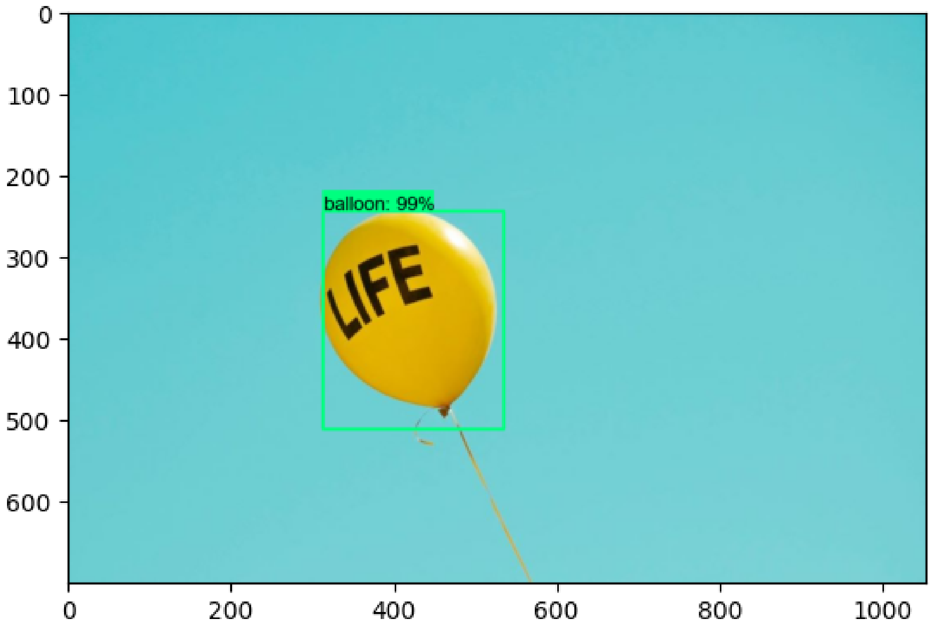


Figure 8.11: Object detection with Tensorflow model.

YOLOv5

Identifying an object using machine learning is one of the methods that were proposed during the project. The idea was to train a custom model, using publicly available tools and since object detection is a crucial task in various industries, including autonomous vehicles, retail, and security systems, this technology was of great interest for the project.

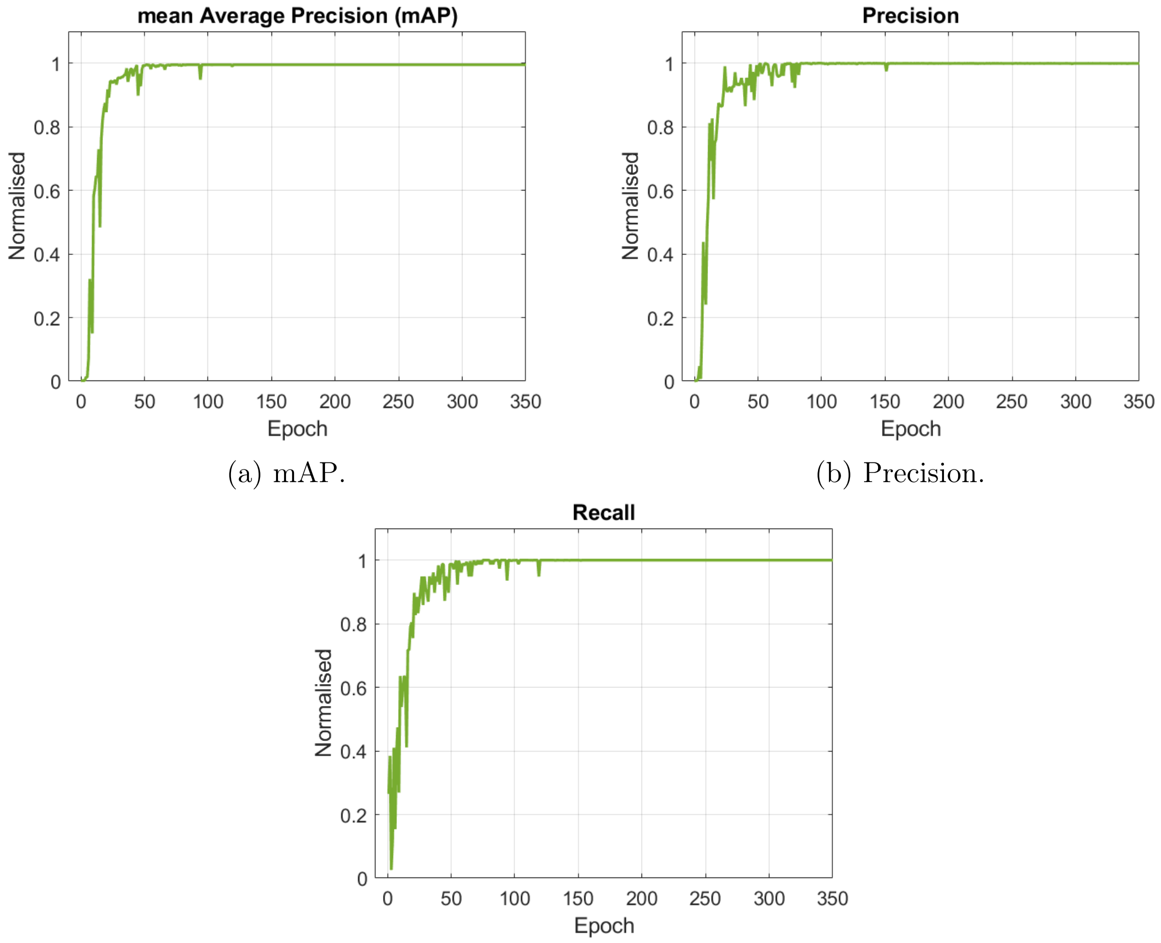
YOLO is a popular algorithm for object detection due to its speed and accuracy, and even though YOLOv5 [48] is not the latest version of this algorithm, it has shown significant improvements in both speed and accuracy over its predecessors, and it is widely used. Therefore, the decision to use this algorithm was done with consideration of the amount of publicly available information.

The process began with downloading a large number of balloon images, followed by annotation of each image by outlining a rectangle around each individual balloon. This annotation process was conducted using a tool called LabelImg [57]. Subsequently, both the annotated images and their corresponding annotations were uploaded to the online platform, Roboflow [58], for further processing.

Roboflow is a computer vision platform that provides developers and businesses with a suite of tools to create and manage computer vision models. The platform enables users to preprocess, annotate, augment, and deploy computer vision models with ease. It supports a variety of popular deep learning frameworks such as TensorFlow [47] and PyTorch [49].

By utilizing Roboflow, creating a dataset and exporting it for training became a relatively effortless task. After configuring the model, the training process was initiated using the train.py script provided by the YOLOv5 repository, and was employed in Google Colab [59] using PyTorch. The training process was repeated several times to achieve a satisfactory outcome.

Google Colab is a free tool that provides access to Graphics Processing Unit (GPU) resources, making it a popular choice for training machine learning models. Following completion, the model was displayed and exported as a '.png' file using Netron [60], which can be found in Appendix A. Once the training was complete, the performance of the model was evaluated using the test.py script. Metrics such as precision, recall, and mean Average Precision (mAP) was obtained, which indicate how well the model is performing. These diagrams can be seen in fig. 8.12.



(a) mAP.

(b) Precision.

(c) Recall.

Figure 8.12: Metrics of the custom trained model.

Average Precision (AP) is a metric that evaluates the precision-recall curve of an object detection model. It summarizes the model's ability to rank objects correctly based on their predicted probabilities. AP is calculated by taking the average precision value at different recall levels, typically ranging from 0 to 1. The precision-recall curve is generated by varying the confidence threshold for object detection. AP summarizes the precision-recall curve using a single value [61].

mean Average Precision (mAP) in fig. 8.12a is determined by considering recall values ranging from 0 to 1. It represents the average of the individual Average Precision (AP) values. In object detection tasks, where multiple object classes are involved (e.g., person, car, dog), the model's performance is assessed separately for each class. To calculate mAP, AP is computed for each class, and then the average is taken across all classes. This metric offers a comprehensive evaluation of the model's overall performance across all object classes, providing a consolidated measure of its effectiveness [61].

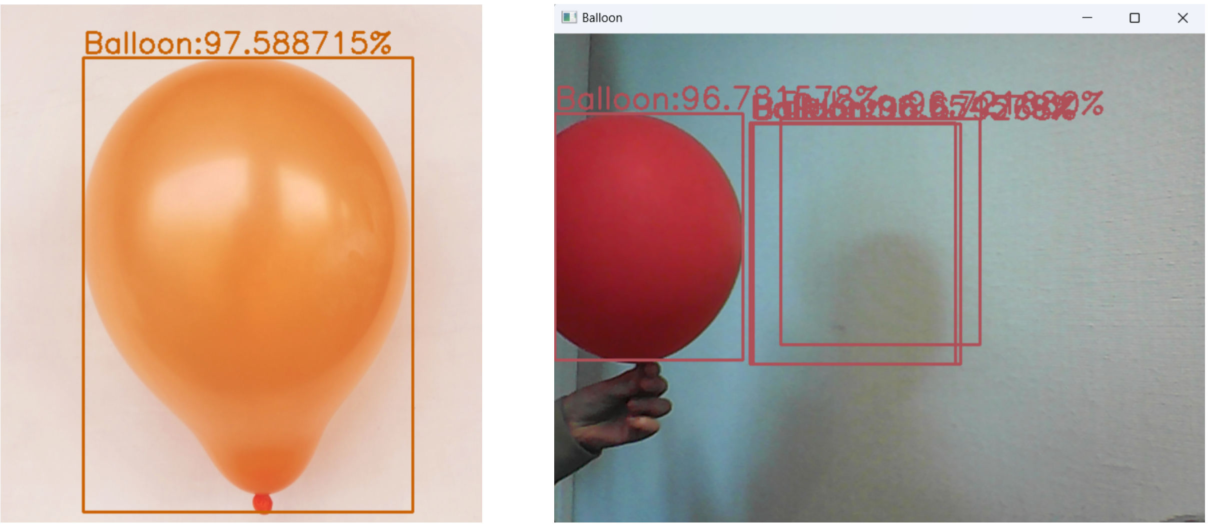
Precision in fig. 8.12b is a metric that indicates the accuracy of detections made by a model. It is determined by dividing the number of true positives by the sum of true positives and false positives. Precision measures the ratio of correctly predicted positive instances (true positives) to the total number of instances predicted as positive. The precision value ranges between 0 and 1. This metric helps assess the reliability of the model's positive predictions [61]. The precision formula is as follows:

$$\text{Precision} = \text{True Positives}/(\text{True Positives} + \text{False Positives}). \tag{8.1}$$

Recall fig. 8.12c, also referred to as sensitivity or true positive rate, reflects the ability of a model to capture relevant instances. It is computed by dividing the number of true positives by the sum of true positives and false negatives. Recall measures the proportion of correctly predicted positive instances (true positives) out of all actual positive instances. It provides an indication of how well the model can identify all the relevant objects. The recall value lies between 0 and 1, where higher values indicate a better ability to capture relevant instances [61]. The recall formula is as follows:

$$\text{Recall} = \text{True Positives}/(\text{True Positives} + \text{False Negatives}). \tag{8.2}$$

While attempting to integrate the custom model in Visual Studio [62], errors were encountered and significant troubleshooting was required. It was discovered that during the export of the model from '.pt' to '.onnx', a line of code within the detect.py file needed to be set to false in order to prevent interference with Deep Neural Network (DNN) inference when using PyTorch. Specifically, the code line "do_constant_folding=True" needed to be changed to "do_constant_folding=False" [63]. After making this modification, the custom model was able to successfully process an image, as illustrated in fig. 8.13a. However, when attempting to utilize the model in real-time with a web camera, performance was slow and the outcome did not meet expectations which can be seen in fig. 8.13b.



(a) Image.

(b) Real-time.

Figure 8.13: Testing the model on an image and in real-time.

fig. 8.13b shows a scenario where the balloon was moves to the side of the frame, resulting in the bounding boxes, which are the colored boxes that is being displayed around the balloons being shifted along with the balloon. However, the bounding boxes were also stacked on top of one another within the frame. The underlying code employed for this task, was taken from an open-source repository on GitHub [64]. It was concluded that the code employed was not optimal for the custom model and the computational resources of the machine utilized did not meet the required specifications. Training a custom YOLOv5 model using Google Colab can be a powerful way to build accurate object detection systems. In this project the time constraint made it hard to realize the idea of integrating the model to the system.

8.2.4 Identifying Objects with Color

JM

Early in the project, a color detection algorithm was also considered as a possible method for categorizing detected objects. However, there was no previous knowledge on how to implement this algorithm, and a tutorial was followed to gain an understanding. The tutorial, which utilized C++ with OpenCV and was compiled in Visual Studio, only demonstrated how to run the algorithm on an image [52].

To incorporate the algorithm with the webcam, the same method used in Qt framework to implement the camera feed in the GUI was employed. Integrating the algorithm into the existing code required multiple attempts and troubleshooting. Furthermore, additional functions were created, and the code was separated to enhance readability and comprehension.

The main function of the code is to detect and recognize objects of interest from the video feed and display information about the detected object on the GUI.

object_detect(): This function is the main loop of the application that runs continuously while the GUI window is open. It reads frames from a video capture device (webcam, file, etc.), finds colored regions in the frame using the *findColoredRegions* function, updates the color display using the *updateColorDisplay* function, and displays the video frame in the label_camera_stream widget.

findColoredRegions(): This function takes a OpenCv Mat object representing a video frame as input and returns a 2D vector *coloredObjectPoints*, where each element is a 3-tuple containing the (x, y, color_index) of a colored object detected in the frame. The color index refers to the index of the color in the *colorValues* vector used to find the object.

getContours(): This function takes a binary image as input and returns the centroid of the object with the largest area in the image as a Point object. It does this by finding the contours of the image using OpenCV's *findContours* function, approximating the contours using *approxPolyDP*, and finding the bounding box of the contours using *boundingRect*. It then calculates the centroid of the bounding box and returns it.

getCategoryAndStyleSheet(): This function takes a scalar object representing the color of an object as input and returns a pair containing the category and stylesheet of the object. The category can be "Friendly," "Hostile," "Unknown," or "Waiting for status," depending on the color of the object. The stylesheet is a string that represents the Cascading Style Sheets (CSS) style of the text used to display the category in the GUI.

updateColorDisplay(): This function takes the *coloredObjectPoints* vector returned by *findColoredRegions* and the *colorValues* vector containing the colors used to find the objects as inputs. It loops over each point in *coloredObjectPoints*, extracts the corresponding color from *colorValues*, and uses *getCategoryAndStyleSheet* to get the category and stylesheet of the object. It then updates the object_category label in the GUI to display the category and stylesheet and adds a log entry to the LoggingList_2 widget. Finally, if the object is categorized as "Hostile," it enables several buttons in the GUI that are used to control the projectiles.

8.3 Graphical User Interface

JM

A Graphical User Interface (GUI) is a type of User Interface (UI) that allows a user to interact with a system through the use of icons, menus, buttons etc. It provides a more user-friendly way to interact with a computer system compared to command-line interfaces. GUIs often display information in a graphical form, such as images, charts, and diagrams, making it easier for users to understand and interpret data.

In Chapter 6.1 and 6.2.1 the state of the GUI and overall software was briefly explained when taking over the project from last year [1].

This chapter however, describes the new functionality and some improvements made to the original GUI.

8.3.1 Upgrades for Improved Functionality

JM

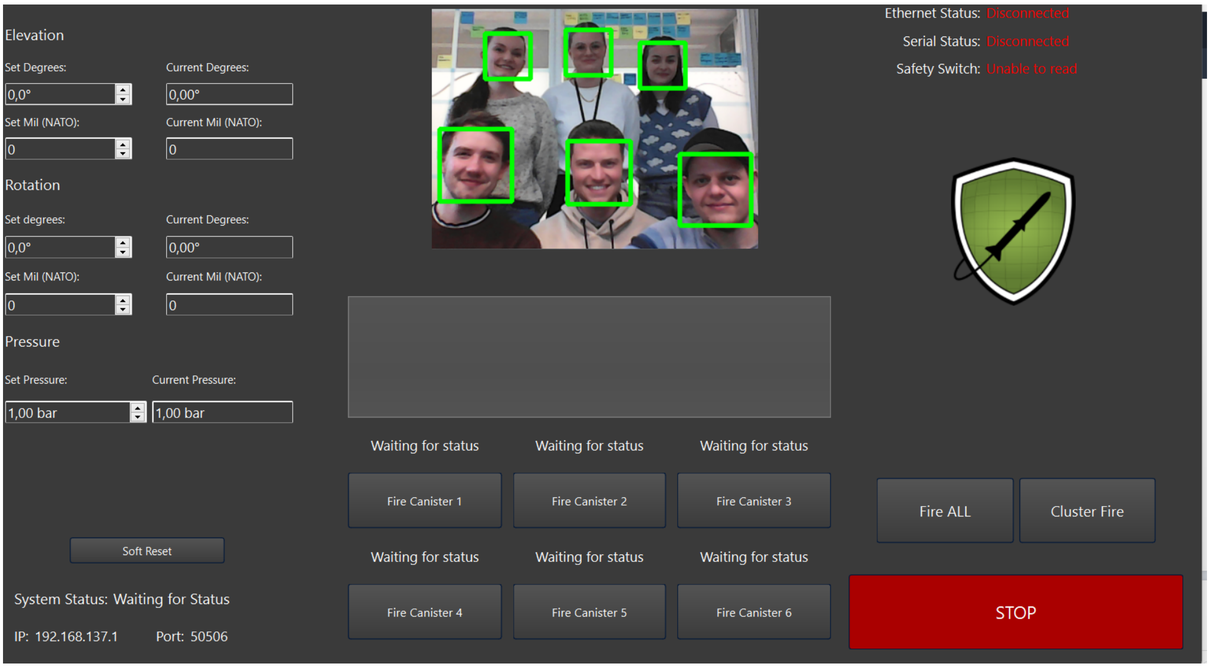
Various modifications have been made to improve the design and functionality of the system. These changes cover several aspects, including the integration of OpenCV for streaming a web camera, adjustments to the buttons, the implementation of functions and improvements in tracking and visualization of the code base.

Streaming Webcamera

JM

The source code from the previous years project [1] was used, which was then enhanced with the integration of the OpenCV library. To configure OpenCV, the library paths were added to the GUI.pro file. In addition, a GitHub repository code [65] was used as an example of how to write code to open the webcam with OpenCV in Qt. A few debugging sessions were required, but ultimately, integration with the updated source code from last year was successful. Integrating face detection was also done using OpenCVs pre-trained Haarcascade facetracking model, which is a machine learning-based algorithm used for object detection in images or videos. This was not a customer requirement, it was pursued as a means of furthering research into object detection and to gain familiarity with computer vision programming and machine learning techniques. This can be seen in Fig. 8.14, where the face tracking can be seen on all group members. For a more detailed explanation of using an object detection model and color detection algorithm with the camera, please refer to Chapter 8.2.2.

An implementation of the camera were also done using Qt C++ classes and modules in Qt Framework. Researching various options to implement a camera feed provided a better understanding of the Qt Framework, and see its variety of functionalities. As requirements for the project was to identify an object, the lack of object detection possibilities with Qt C++ classes and modules alone made OpenCV the ultimate choice.



Elevation

Set Degrees: 0,0°

Current Degrees: 0,00°

Set Mil (NATO): 0

Current Mil (NATO): 0

Rotation

Set degrees: 0,0°

Current Degrees: 0,00°

Set Mil (NATO): 0

Current Mil (NATO): 0

Pressure

Set Pressure: 1,00 bar

Current Pressure: 1,00 bar

System Status: Waiting for Status

IP: 192.168.137.1 Port: 50506

Ethernet Status: Disconnected

Serial Status: Disconnected

Safety Switch: Unable to read

Waiting for status Waiting for status Waiting for status

Waiting for status Waiting for status Waiting for status

Figure 8.14: Face tracking on all group members using OpenCV.

Fire Canister Buttons

JM

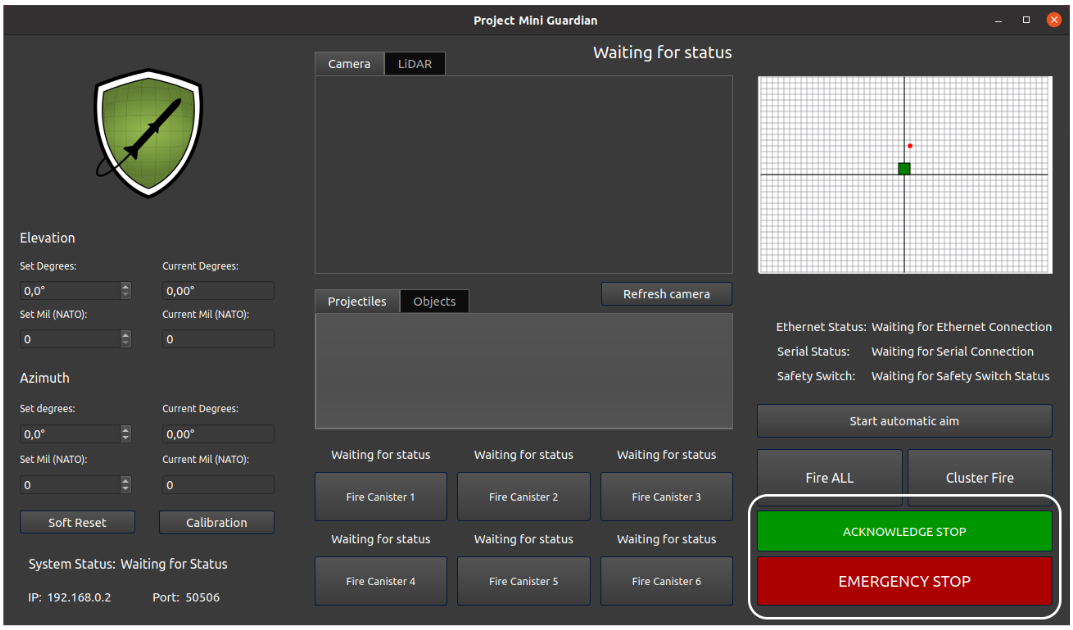
Firstly, the button order has been adjusted to ensure that pressing the "Fire Canister" buttons logs the corresponding canister, which was not the case in the original source code due to button misplacement. Similarly, pressing the "Fire Canister" buttons now displays that the right canister is empty.

Several buttons also underwent name changes due to a lack of correspondence between their names and associated actions. For instance, the "Fire all" button now provides an indication that all the canisters were fired upon pressing it. The "Cluster fire" button was also found to be logging the fired canisters in an incorrect order, which has since been rectified.

Emergency and Acknowledgment Stop Buttons

JM

An emergency stop button has been implemented, controlled by the GUI, along with an "acknowledge stop" button that needs to be pressed before the motors can be driven again. The emergency button initiates the deactivation of the stepper motor, while the acknowledge button initiates the activation of the stepper motor. In Fig. 8.15, the buttons are highlighted in white. There was a flaw in the design that caused the Effector to first move to the original angle after pressing the confirm button and entering a new azimuth degree. This behavior persists unless the new degree is greater than the previous one. The emergency button has been implemented as per the safety requirement NFR02, which states that *"The system shall not be a risk for health and safety"*.



Waiting for status

Elevation

Set Degrees: 0,0°
Current Degrees: 0,00°
Set Mil (NATO): 0
Current Mil (NATO): 0

Azimuth

Set degrees: 0,0°
Current Degrees: 0,00°
Set Mil (NATO): 0
Current Mil (NATO): 0

System Status: Waiting for Status
IP: 192.168.0.2 Port: 50506

Waiting for status Waiting for status Waiting for status
Waiting for status Waiting for status Waiting for status

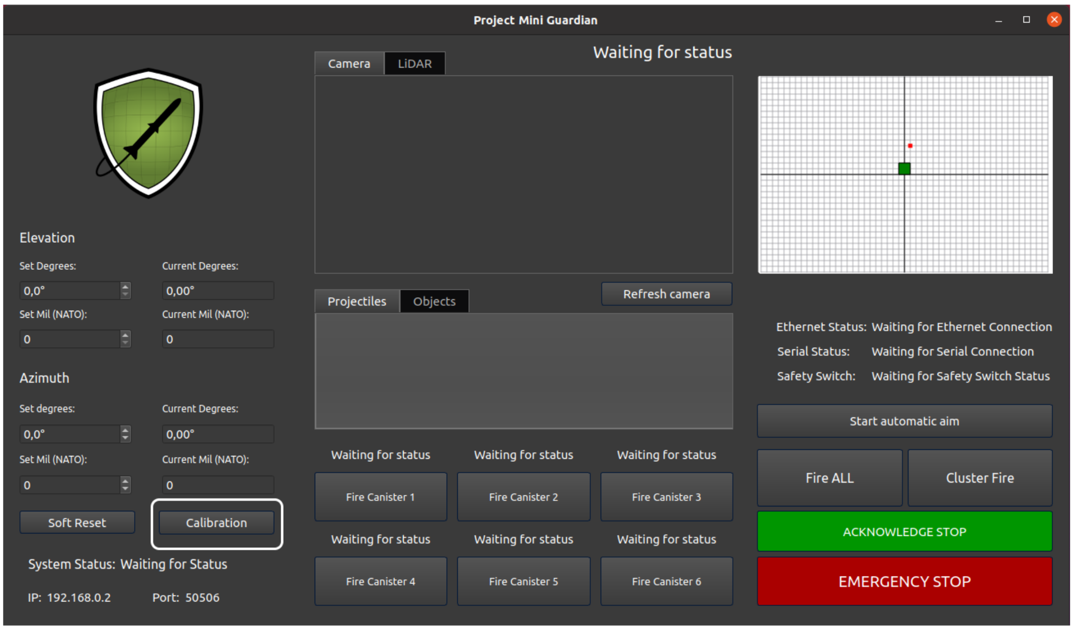
Ethernet Status: Waiting for Ethernet Connection
Serial Status: Waiting for Serial Connection
Safety Switch: Waiting for Safety Switch Status

Figure 8.15: GUI with the emergency and acknowledge button highlighted.

Calibration Button

JM

Additionally, a calibration button has been added to address the issue of uncertainty about the extent of rotation each time the elevated base was manually moved, or the system was shut down before the base was reset to the zero point. The button initiates the stepper motor and rotates the base until the zero point switch is activated. Fig. 8.16 shows this button highlighted in white. More on this function can be read in Chapter 9.1.7.



Waiting for status

Elevation

Set Degrees: 0,0°
Current Degrees: 0,00°
Set Mil (NATO): 0
Current Mil (NATO): 0

Azimuth

Set degrees: 0,0°
Current Degrees: 0,00°
Set Mil (NATO): 0
Current Mil (NATO): 0

System Status: Waiting for Status
IP: 192.168.0.2 Port: 50506

Waiting for status Waiting for status Waiting for status
Waiting for status Waiting for status Waiting for status

Ethernet Status: Waiting for Ethernet Connection
Serial Status: Waiting for Serial Connection
Safety Switch: Waiting for Safety Switch Status

Figure 8.16: GUI with the calibration button highlighted.

Overtravel Function

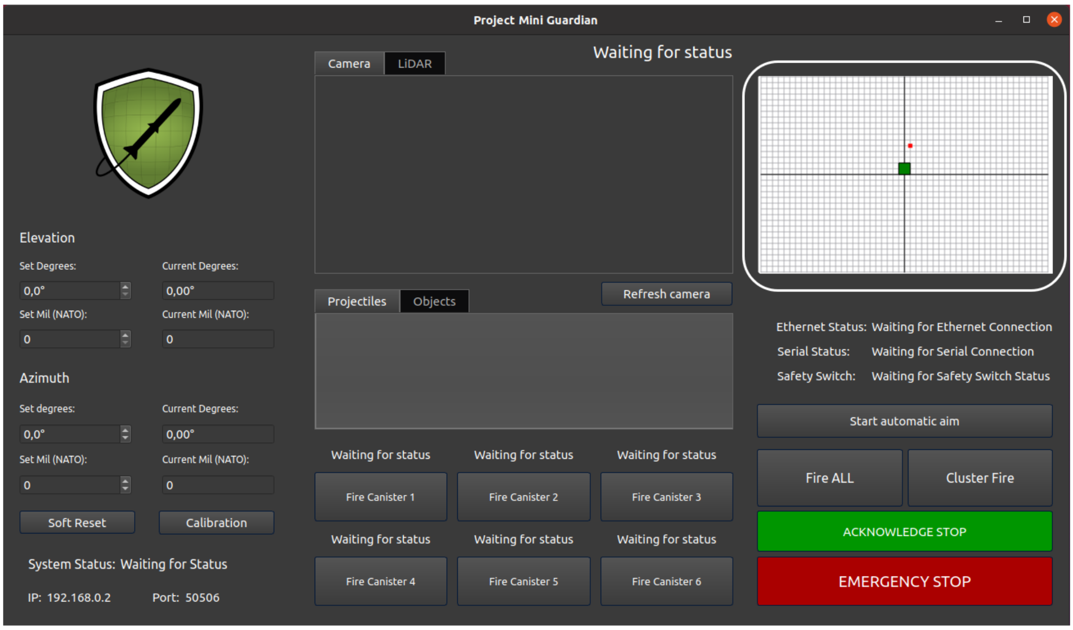
JM

A new function has been added into the Graphical User Interface (GUI) to notify the operator when the Effector exceeds its rotational limit of 450°. This safety function operates in the background, ensuring that the Effector is monitored without requiring any action from the operator during rotation.

2D Coordinate system

JM

To enhance the operator's understanding of the detected object's position relative to the Effector, a coordinate system widget has been implemented. This widget provides a visual representation of the coordinates obtained from the LiDAR sensor and the placement of the Effector. The Effector is represented by a green square located at the origin of the coordinate system. This addition allows the operator to easily perceive the spatial relationship between the object and the Effector. This widget is highlighted in Fig. 8.17.



Project Mini Guardian

Waiting for status

Elevation

Set Degrees: 0,0° Current Degrees: 0,00°

Set Mil (NATO): 0 Current Mil (NATO): 0

Azimuth

Set degrees: 0,0° Current Degrees: 0,00°

Set Mil (NATO): 0 Current Mil (NATO): 0

System Status: Waiting for Status

IP: 192.168.0.2 Port: 50506

Ethernet Status: Waiting for Ethernet Connection

Serial Status: Waiting for Serial Connection

Safety Switch: Waiting for Safety Switch Status

Waiting for status Waiting for status Waiting for status

Waiting for status Waiting for status Waiting for status

Figure 8.17: GUI with the coordinate system highlighted.

Retrieving LiDAR Coordinates

VJ

A button incorporating the retrieval of coordinates for detected objects from the LiDAR was implemented to the GUI. The button calls a function which opens an UDP socket, processing the data sent from ROS which process the LiDAR data. Given the real-time capabilities of the LiDAR, the detected objects coordinates are updated continuously and its position data is sent accordingly. Incorporating a button which has to be pressed once to get position data ensures that data is transmitted in smaller increments, preventing the risk of overwhelming Qt with a large amount of data at once. If the position data were to be sent to Qt without any form of downscaling would result in a significant data load, and could overload or significantly delay the GUI. As delays and overload of the GUI could lead to a system not detecting objects within a timeframe, the controlled button is important for the systems functionalities.

Converting LiDAR Coordinates

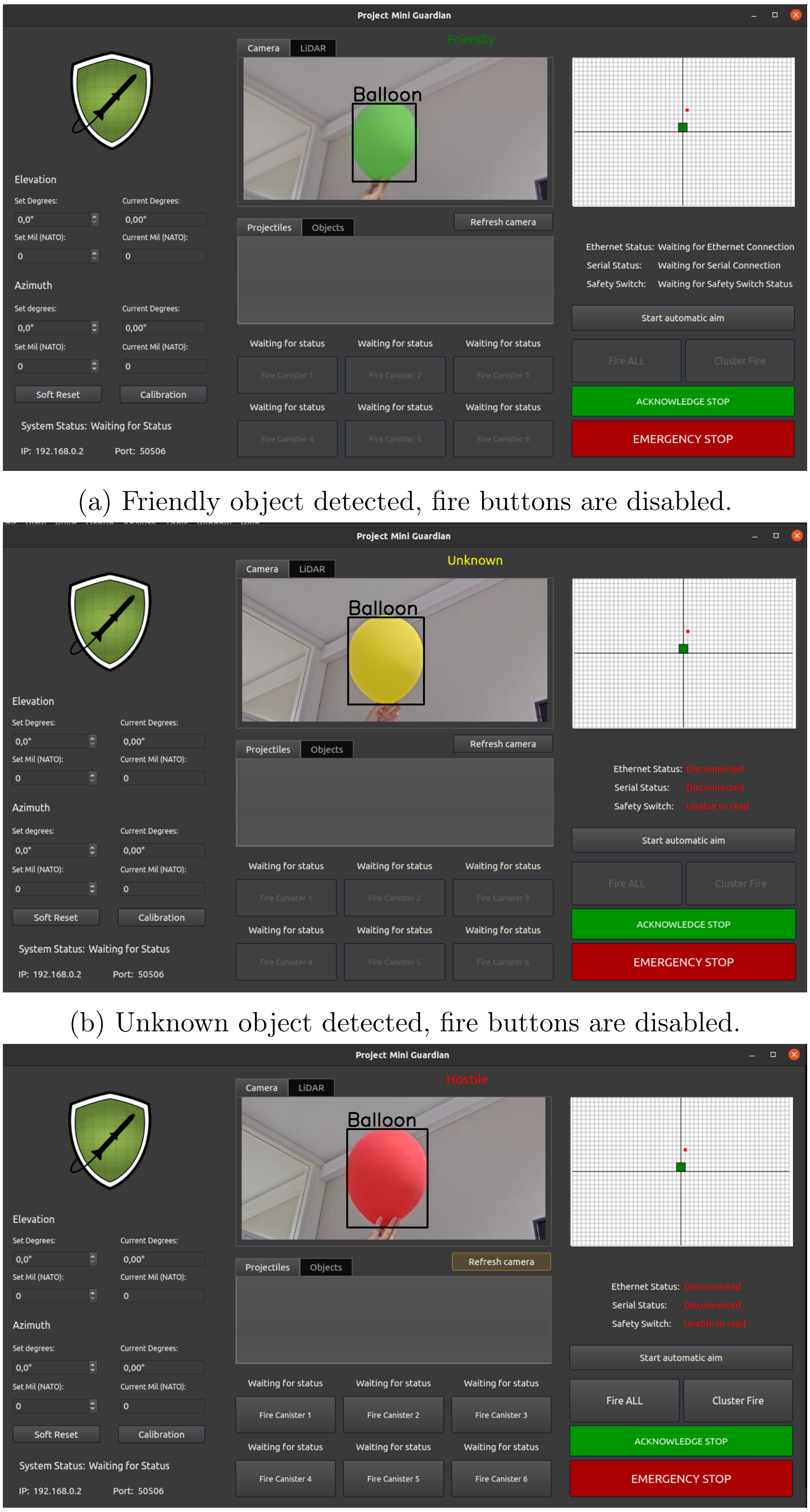
JM

To detect the balloons using LiDAR, a function was developed to extract x, y, and z coordinates from the LiDAR data. These coordinates were then converted from Cartesian to spherical coordinates. To read more about the conversion, refer to 8.4.3. Additionally, an automatic feature was implemented in the GUI, triggered by pressing a button, that sets the elevation and rotation angle. This required the addition of a new function, which incorporates new code and calls the existing functions responsible for converting angles into steps for the stepper motor.

In addition to the modifications previously mentioned, a second logging list has been incorporated for objects detected by the camera. The fire buttons are now also disabled if the camera detects an Unknown or Friendly object, to prevent unintended firing, as per the requirements of FR-05 which states that *"The system's Effector shall only trigger against targets that are categorized as Hostile."* Further details about the color detection algorithm can be found in Chapter 8.2.4. Fig. 8.18 shows the enabled and disabled buttons when different objects has been categorized.

To optimize the code and make it more clear, extra code that did not align with the project requirements has been removed, and a general cleanup of the code has been performed.

All the code for the added functionality mentioned above can be found in the Doxygen files. They are labeled with the names of the person who wrote them.



Project Mini Guardian

Friendly

Elevation
Set Degrees: 0,0°
Current Degrees: 0,00°
Set Mil (NATO): 0
Current Mil (NATO): 0

Azimuth
Set degrees: 0,0°
Current Degrees: 0,00°
Set Mil (NATO): 0
Current Mil (NATO): 0

System Status: Waiting for Status
IP: 192.168.0.2 Port: 50506

Ethernet Status: Waiting for Ethernet Connection
Serial Status: Waiting for Serial Connection
Safety Switch: Waiting for Safety Switch Status

(a) Friendly object detected, fire buttons are disabled.

Project Mini Guardian

Unknown

Elevation
Set Degrees: 0,0°
Current Degrees: 0,00°
Set Mil (NATO): 0
Current Mil (NATO): 0

Azimuth
Set degrees: 0,0°
Current Degrees: 0,00°
Set Mil (NATO): 0
Current Mil (NATO): 0

System Status: Waiting for Status
IP: 192.168.0.2 Port: 50506

Ethernet Status: Disconnected
Serial Status: Disconnected
Safety Switch: Unable to read

(b) Unknown object detected, fire buttons are disabled.

Project Mini Guardian

Hostile

Elevation
Set Degrees: 0,0°
Current Degrees: 0,00°
Set Mil (NATO): 0
Current Mil (NATO): 0

Azimuth
Set degrees: 0,0°
Current Degrees: 0,00°
Set Mil (NATO): 0
Current Mil (NATO): 0

System Status: Waiting for Status
IP: 192.168.0.2 Port: 50506

Ethernet Status: Disconnected
Serial Status: Disconnected
Safety Switch: Unable to read

(c) Hostile object detected, fire buttons are enabled.

Figure 8.18: Displaying Friendly, Unknown and Hostile objects in the GUI.

Streaming IP Camera

JM

In the later stages of the project, the purchase of an AXIS M5074 PTZ Camera resulted in an integration process that took longer than expected. Thorough research was conducted, which included checking the firewall and debugging the code to ensure its functionality. Multiple platforms were tested to access the stream, including Visual Studio and VideoLAN Client (VLC), and it was found that the stream could be accessed through various Universal Resource Locator (URL)s, including Hypertext Transfer Protocol (HTTP) and RTSP.

After conducting a detailed analysis, it was determined that the issue was either related to a format problem with the OpenCV library (even though it was supposed to support the stream) or a configuration problem with Qt. The problem was resolved by changing the kit (which consists of the compiler, debug files, Qt version etc.) in Qt and rebuilding the code, which enabled the system to function properly. The kit was changed from "Desktop Qt 6.4.2 MSVC2019 64bit" to "Desktop (x86-windows-msvc2019-pe-64bit). This was performed on a Windows computer. In the later stages of the project, the system was transferred to the dedicated computer. For additional details about the computer, refer to 8.5.4.

Streaming LiDAR

VJ

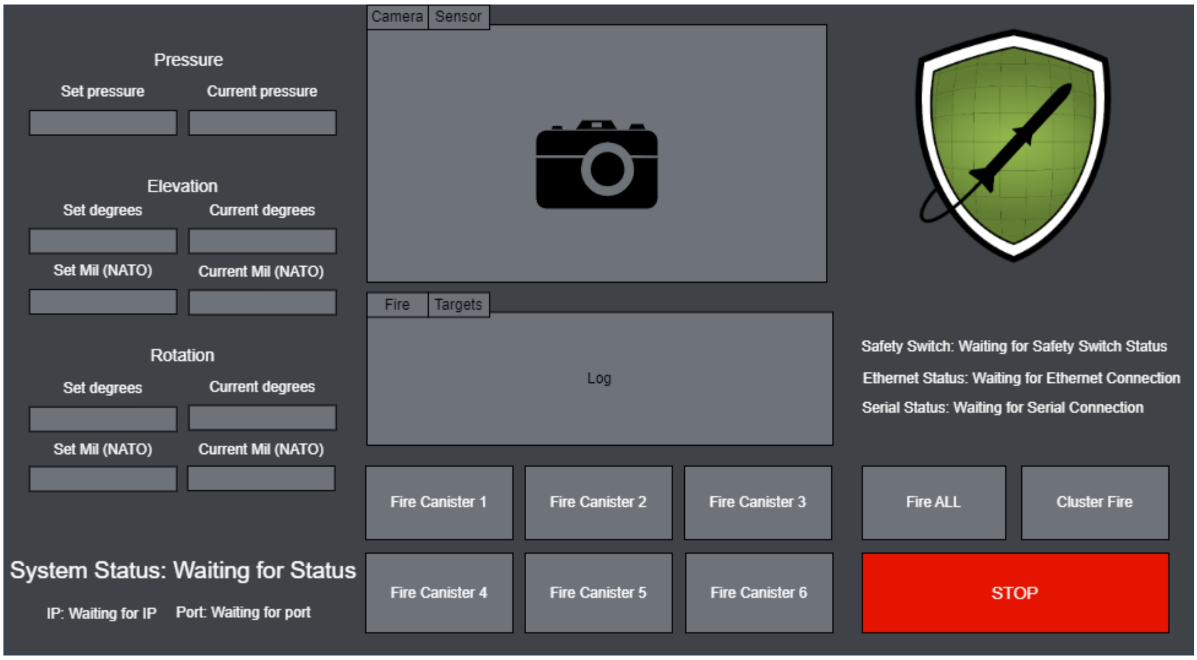
Visualizing the detected objects from the LiDAR in the GUI would lead to an interface with a better overview of all components, as well as a more seamless workflow for the operator. When the LiDAR feed is shown in a ROS visualization tool called RViz, the operator requires both GUI and RViz open in two separate windows to visualize the object detection simultaneously as handling the other system functionalities. Currently, the GUI has camera feed only and the LiDAR feed is shown in RViz.

GUI Layout

VJ

As the new system got new features, the GUI required a new layout. When adapting the GUI layout to be suitable for the system, some proposals were made as seen in fig 8.19 and 8.20. There could be a feed widget with the opportunity to switch between the camera and LiDAR sensor feed, similar to the Fire Distribution Center (FDC) for the NASAMS. As requirements states that information shall be displayed in the GUI, there arises a need for additional space in the log widgets. These requirements support the claims that the provided GUI from [1] is not suitable for the current system, hence the proposals for GUI layouts.

There is only one final layout, and it is adapted from the proposals as well as the new functionalities in the GUI discussed above. The final layout of the GUI can be seen in fig. 8.21.



Safety Switch: Waiting for Safety Switch Status

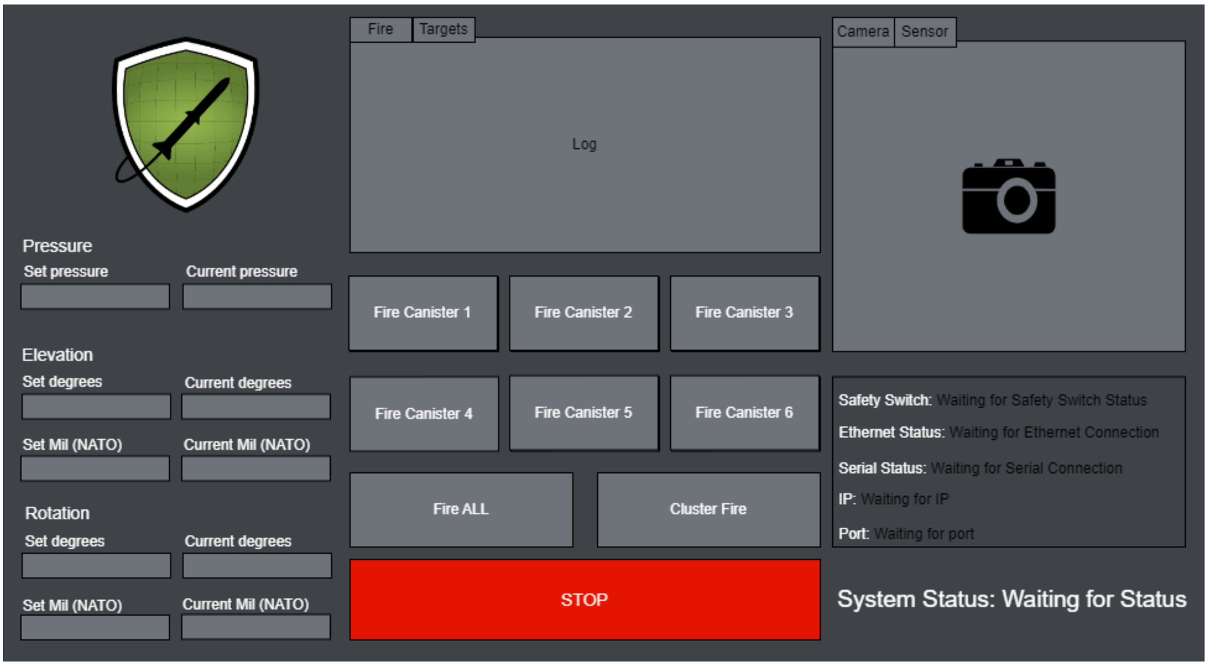
Ethernet Status: Waiting for Ethernet Connection

Serial Status: Waiting for Serial Connection

System Status: Waiting for Status

IP: Waiting for IP Port: Waiting for port

Figure 8.19: First Proposed GUI Layout



Safety Switch: Waiting for Safety Switch Status

Ethernet Status: Waiting for Ethernet Connection

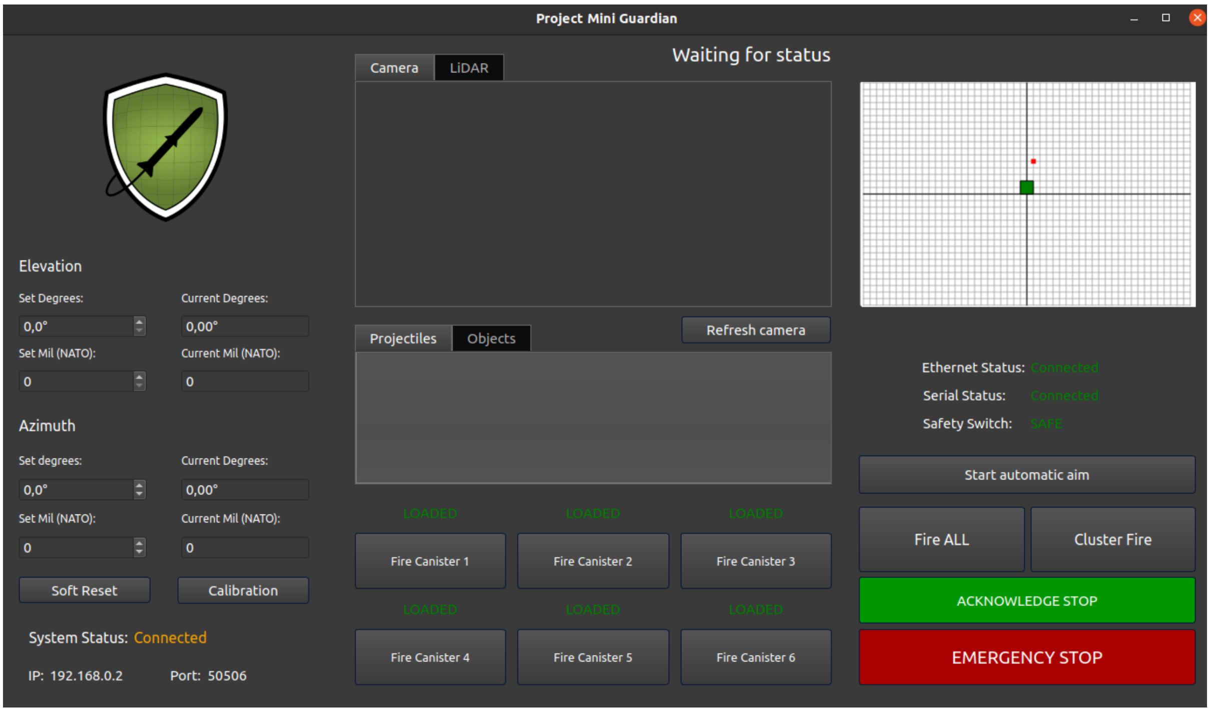
Serial Status: Waiting for Serial Connection

IP: Waiting for IP

Port: Waiting for port

System Status: Waiting for Status

Figure 8.20: Second Proposed GUI Layout



Waiting for status

Elevation

Set Degrees: 0,0°
Current Degrees: 0,00°
Set Mil (NATO): 0
Current Mil (NATO): 0

Azimuth

Set degrees: 0,0°
Current Degrees: 0,00°
Set Mil (NATO): 0
Current Mil (NATO): 0

System Status: Connected

IP: 192.168.0.2 Port: 50506

Ethernet Status: Connected
Serial Status: Connected
Safety Switch: SAFE

Figure 8.21: Final GUI layout

8.4 Device Communication

JM | VJ

The system comprises of four components; a computer with a GUI, a LiDAR sensor, an AXIS camera and the Effector. To establish communication between the devices and the computer, an Ethernet connection is required for all components. Managing the communication requires a switch for all devices to send and receive messages simultaneously. The switch deployed is Tp-Link TL-SG105PE [66].

The switch automatically assigns IP addresses to connected devices. However, in this scenario, only the LiDAR and the Ethernet shield on the Arduino Mega did not receive their default IP address, while all other devices obtained it. In table 8.1 the IP addresses for the different devices are shown.

Table 8.1: Device IP addresses

Device	IP address
LiDAR	10.19.35.30
AXIS Camera	192.168.0.90
Effector	192.168.137.1
Network Switch	192.168.0.1

When connecting the LiDAR to the computer, a problem arose. The borrowed LiDAR from the summer internship project known as Coastal Shark lacked any accompanying information or user guide. Since the LiDAR utilized an Ethernet connection, determining its specific IP address became necessary. By using the tool Wireshark [67] to analyze the data source, it was discovered that the IP address had been modified from its default setting to a highly unique one. Initially assuming the LiDAR had its default IP address, attempts to establish a connection naturally failed. Eventually, the connection was successfully established using the IP address obtained from Wireshark.

Since the Effector relies on the Arduino Mega board's Ethernet shield for communication, it requires a configuration of two IP addresses: one for the Ethernet shield itself and another for an external UDP socket in the GUI. These IP addresses are also defined in the GUI socket code. If the Effector is connected to the switch, the external IP address should match the switch's IP. Alternatively, if the Effector is directly connected to the computer, the IP address needs to be set to match the Ethernet shield's IP address on the computer. Fig. 8.22 illustrates a straightforward network diagram over the connection between the different devices. The system assigned had serial and Ethernet connections between the deployed components and required no external component to acquire communication between them. As the current system has an Ethernet switch deployed to acquire communication, the communication connection diagram is updated with the new system in Fig. 8.23

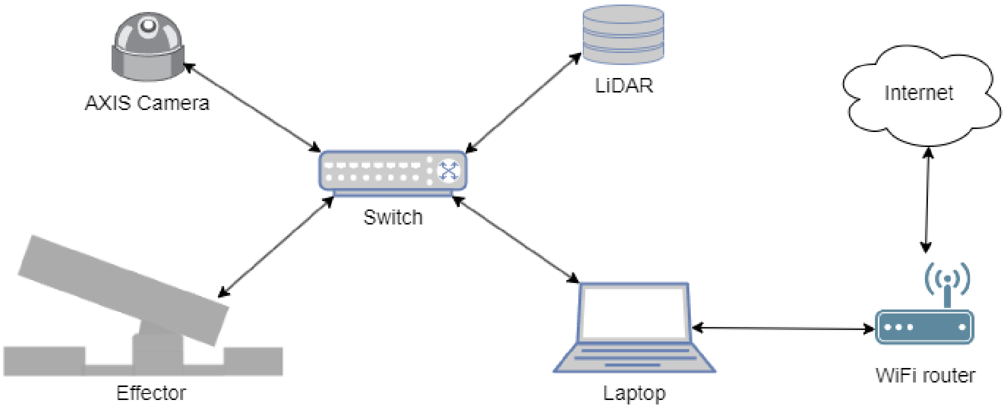


Figure 8.22: Network diagram

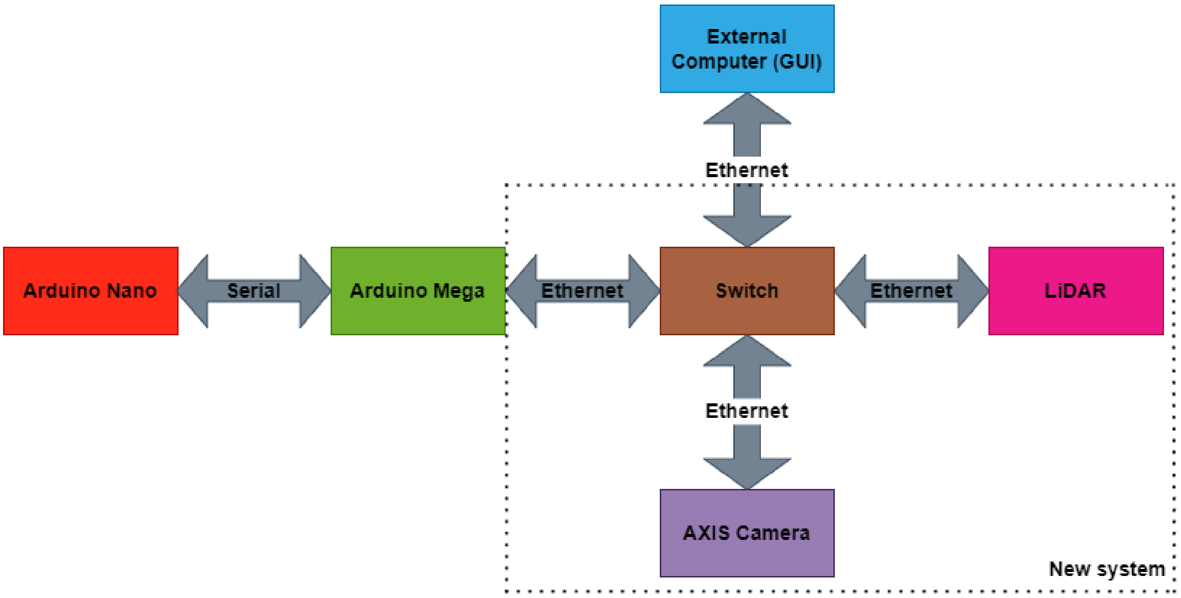


Figure 8.23: Communication connections

8.4.1 TTL Serial Communication

JM

The system, from the previous year's project [1], employed an Arduino Mega and an Arduino Nano for controlling its operations. These hardware components have been retained and continue to serve a crucial role in the system's functionality. Both the Arduino Mega and Nano boards are equipped with (Universal Asynchronous Receiver-Transmitter (UART) hardware [68], which allows them to communicate with other devices using serial communication protocols. UART is an asynchronous serial communication protocol that uses two data lines: one for transmitting data (TX) and one for receiving data (RX). Transistor-Transistor Logic refers to the voltage levels used in the communication, typically 5V for logic high and 0V for logic low [69].

The communication values from the project of the previous year has undergone some updates, and these changes are denoted by bold text in Table 8.2. The report does not include the list of previous values that have not been updated.

Table 8.2: Commands from GUI to Arduino mega

Command	Info
10	Request an "ON" ping from Nano
11	Set LED to green (System Ready)
12	Set LED to orange (System Active)
13	Set LED to red (System not ready)
50	**Request "Emergency stop"**
55	**Request "Acknowledge emergency stop"**
56	**Request "Calibration"**
66	Request “Fire ALL”
67	Request Cluster Fire
100	Request canister status
200	Request FIRE from Canister 0
201	Request FIRE from Canister 1
202	Request FIRE from Canister 2
203	Request FIRE from Canister 3
204	Request FIRE from Canister 4
205	Request FIRE from Canister 5

8.4.2 Communication ROS, LiDAR and GUI

VJ

To acquire communication between the LiDARand GUI for them to instruct the Effector to rotate and elevate, ROS and User Datagram Protocol (UDP) sockets were deployed. ROS were used to process the LiDAR data, as well as opening an UDP socket to send the required information to the GUI, which sends instructions to the Effector. Fig. 8.24 demonstrates the communication in ROS when the object detection for the LiDAR is enabled. The LiDAR and GUI are present in the diagram to visualize the flow of input and output data.

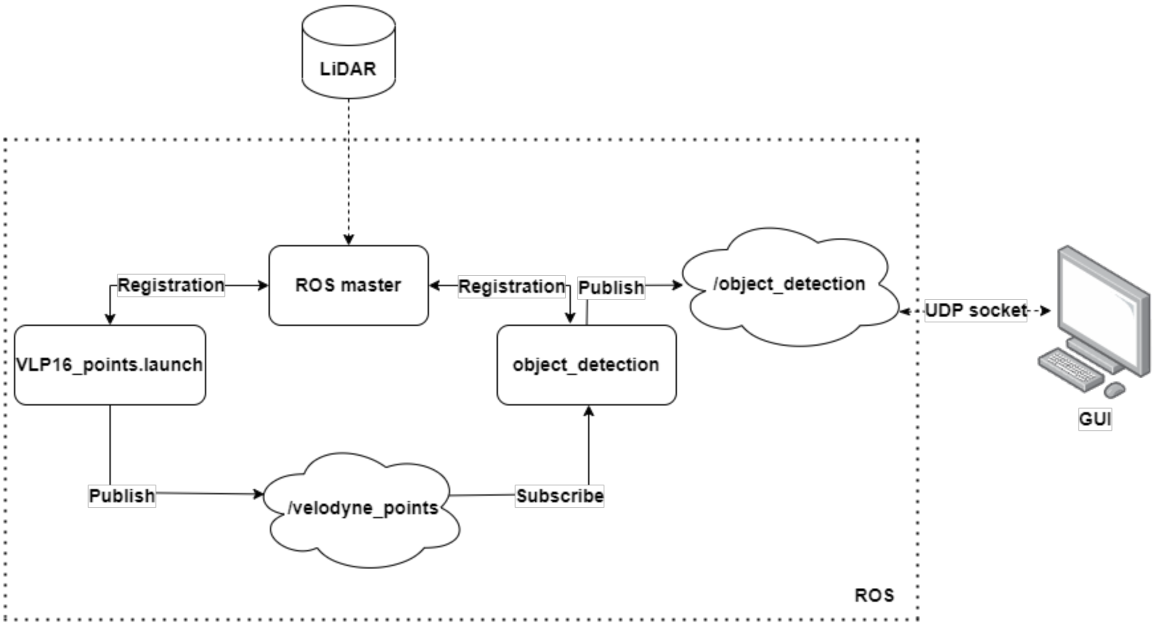


Figure 8.24: Current system ROS communication

The LiDAR sensor captures environmental data which is processed by ROS. ROS launches a node called “VLP16_points.launch”, which launch the LiDAR point cloud and publish its outputs to a topic called “velodyne_points”. *This node is provided by the VLP16-driver [70] and is required for launching the necessary nodes for displaying the LiDAR data.*

This topic is subscribed to by the “object_detection” node, which divides the point cloud into clusters to facilitate object detection. The resulting clusters are published to the “/object_detection” topic who has three subnodes called "cloud_filtered", “markers” and “poses”.

Cloud_filtered displays the filtered point cloud, while markers will visualize a cuboid around the detected object, and poses publish the centroid of the cuboid visualized. This visualization can be seen in Fig. 8.25. Poses depends on markers, meaning that the cuboid position will not be published unless markers are published. As the GUI and Effector requires position of a detected object, it was deemed suitable to place the code for a socket sending positions within "poses". In the case of no objects being detected, the socket sends an empty position within "cloud_filtered".

Sockets allows for communication between programs or processes over the network, based on IP addresses and ports. An UDP socket has fast transmission speed and offers low latency, making it suitable for real-time applications [71]. As requirement SR-01-05 states that *“The Sensor should detect newly appeared or relocated objects within 5 s”*, a socket offering fast transmission speed was appropriate. The UDP socket creation in both the GUI and ROS code is influenced by [72].

The GUI is made in Qt, meaning that Qt process the messages sent to the GUI. The UDP socket enabled transmission of data from the ROS topics, and the received messages are processed in Qt which communicates the detected objects positions to the Effectors motor and initiates its operation.

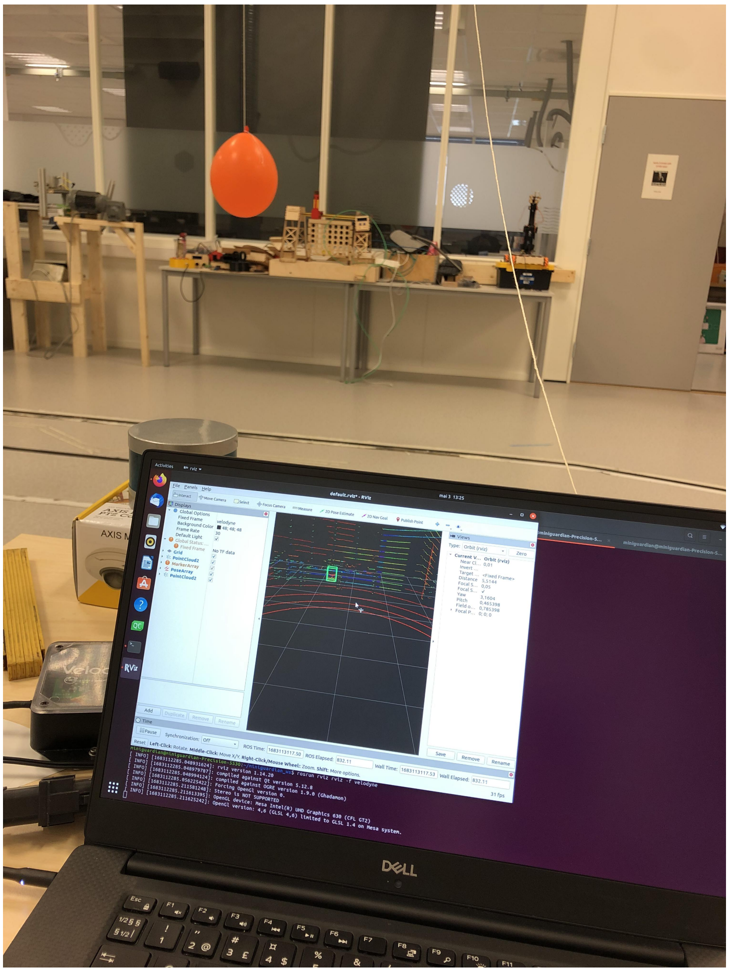


Figure 8.25: Balloon detected with visualized cuboid in RViz.

8.4.3 Converting coordinates

JM

When incorporating a LiDAR sensor into the system, enhancing its functionality for aiming the Effector became important. Manual addition of LiDAR coordinates by the operator is impractical due to their large number, requiring conversion to spherical coordinates before utilizing them with the system's existing functionality. Consequently, converting the Cartesian coordinates from the LiDAR sensor became a necessary step to enable automatic targeting of the detected objects. In this scenario, the conversion is essential since it involved handling both azimuth and polar angles when aiming the Effector at an object. Cartesian to spherical coordinate conversion is a mathematical transformation that converts a point's coordinates from the Cartesian coordinate system (x, y, z) to the spherical coordinate system (r, θ, ϕ) [73].

In the Cartesian system, a point is represented by the (x, y, z) coordinates. These coordinates represent, the distance along each axis [73]. In the spherical system, a point is represented by its coordinates (r, θ, ϕ), where r represents the distance from the origin to the point, θ represents the angle between the positive z-axis and the line connecting the origin and the point (polar angle), and ϕ represents the angle between the positive x-axis and the projection of the line connecting the origin and the point onto the horizontal xy-plane (azimuth angle). Fig. 8.26 illustrates the spherical coordinate system [73].

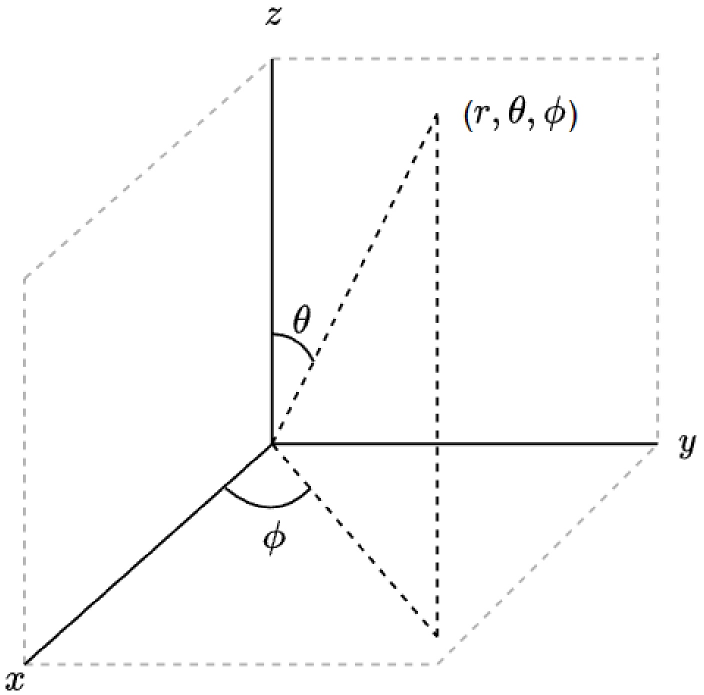


Figure 8.26: Spherical Coordinates

The conversion from Cartesian to spherical coordinates in degrees, was done using the following equations:

$$\begin{aligned} r &= \sqrt{x^2 + y^2 + z^2}, \\ \theta &= \arccos\left(\frac{z}{r}\right) \cdot \frac{180}{\pi} \text{ and} \\ \phi &= \arctan\left(\frac{y}{x}\right) \cdot \frac{180}{\pi}. \end{aligned} \tag{8.3}$$

The ranges of the spherical coordinates are as follows:

$$\begin{aligned} &r \geq 0 \\ &0 \leq \theta \leq 180 \\ &0 \leq \phi < 360. \end{aligned} \tag{8.4}$$

When the system is assembled, the LiDAR sensor is placed approximately 0.80 m above the Effector. This was taken into consideration when converting the z coordinate, by adding 0.80 m in the conversion function. The Effectors maximum elevation is 35° and ϕ is therefore limited to that value in the software implementation.

In the code implementation for the conversion, that can be seen in the code example below, the standard C library header math.h is included. This is a conventional library too use when working with trigonometric functions, making atan equivalent to arctan and acos equivalent to arccos. For the code relating to this conversion, please refer to the list below, and the Doxygen folder.

```cpp
void MainWindow::cart2sph(const vector<double>& coords) {
    double x = coords[0];
    double y = coords[1];
    // Adding 0.80 due to the placement of the LiDAR sensor in relation to the Effector
    double z = coords[2] + 0.80;

    // Compute radial distance
    double r = sqrt(x*x + y*y + z*z);
    // Compute polar angle (in degrees)
    double phi = acos(z/r) * 180 / M_PI;
    // Compute azimuthal angle (in degrees)
    double theta = 90 - (atan2(y, x) * 180 / M_PI); // subtracts 90 due to the z axis being considered as zero in the vertical plane

    if(phi <= 35){
        // call the function that converts degrees to steps
        on_elevationDegSpinBox_valueChanged(phi); // calls the elevation function with the Elevation angle
        sleep_for(milliseconds(2000));
    }else{
        // call the function that converts degrees to steps
        on_elevationDegSpinBox_valueChanged(35); // calls the elevation function with the maximum Elevation angle
        sleep_for(milliseconds(100));
    }
    if(theta <= 360){
        // call the function that converts degrees to steps
        on_rotationDegSpinBox_2_valueChanged(theta); // calls the rotation function with the Azimuth angle
        sleep_for(milliseconds(100)); // Wait 100 milliseconds for mega to be ready
    }
}
```

8.5 Preliminary Research

8.5.1 Qt

VJ

Project Mini Launcher [1] had deployed the Qt framework for the development of their GUI. To extend the GUI's capabilities, research was conducted on the application, involving familiarization with its libraries and features. Gaining an understanding of Qt and interpret the existing code in a new application required both time and patience. As mentioned earlier in 8.3.1, the GUI had some flaws in which required attention. Through studying the code and learning more about the Qt framework, a deeper understanding of the given system was acquired, as well as paving the way for the upcoming implementation of new functionalities.

8.5.2 Virtual Machine vs. Docker

JM

In order to facilitate the data extraction process from a LiDAR, it may be necessary to consider the implementation of a containerization tool for the project at hand. Following discussions with subject matter experts, a number of viable options for data extraction have been identified. While there has been considerable discourse surrounding the use of ROS, communication with the sales team at the LiDAR company (which has since been acquired by Ouster) has revealed that data extraction may be achieved through a variety of different methods. The primary concern currently relates to identifying the optimal approach for integrating the tool with the existing system. To this end, research efforts are focused on identifying a suitable containerization tool, although direct data extraction from the Veloview interface of the LiDAR is also being explored.

When considering containerization tools, it was a challenge to determine which option will best suit the project. While Docker is known for its efficiency and lightweight containerization, there are times when a Virtual Machine (VM) may be a better choice, especially if one is pressed for time and have experience with VMs.

Based on prior familiarity and experience with VMs, a VM-based approach was deemed more efficient in our case if it was necessary. The decision would result in significant time savings and reduced the learning curve associated with adopting a new technology such as Docker.

In terms of isolation, hardware emulation, flexibility and complexity, there is no specific reason to why a VM should be used over Docker for this project.

Ultimately, the choice between a VM and Docker does not depend on the specific requirements and constraints of the project, but of the individual's knowledge and experience with each technology, and the time constraints.

The choice to not use a containerization tool was made towards the end of the project timeline. To facilitate this decision, a dedicated computer was purchased for the project. Prior research was conducted for the virtual machine to determine the appropriate operating system, as well as the compatible distributions of OS, ROS 1, and Qt. This was essential to ensure seamless integration of the new system with the existing one.

8.5.3 ROS as a LiDAR processing tool

VJ

Upon choosing ROS as the processing tool for the LiDAR data, there was little to no knowledge about the operating system among the group members. Tutorials delivered by ROS itself were done [74], trial and error as well as research were conducted which led to an understanding of the system. As time constrains were a liability, the understanding of ROS were not as deep as desired, considering the scale of ROS as a system. To display and connect to the LiDAR through ROS, a tutorial provided by ROS were also followed [75], although some issues arose as mentioned in 8.4. This tutorial included installing the VLP16 driver, which is found in [70]. At first hand, the idea was to use ROS for both the LiDAR and GUI, but converting the code provided from last years project [1] to a code in the ROS workspace revealed some issues in terms of compatibility, so the final solution was to only process LiDAR data in ROS.

8.5.4 Choosing a Computer

VJ

Considering the decision of using ROS Noetic for processing the LiDAR data, the computer OS would be converted to Ubuntu 20.04 to be compatible with ROS Noetic.

The processing of the LiDAR data requires a lot of memory, and a computer that will process a large amount of data and large data sets should have a Random Access Memory (RAM) of at least 16 GB. As the project will consist of more than LiDAR data, 32 GB is the best on the basis of having a well-functioning and fast system. For future extension of the project, a computer being able to process even more than the current system is also highly necessary.

On the basis that the project will run several processes and process data simultaneously on several levels, several cores would be ideal as it requires a certain quality and speed. As the computer were supposed to process data rather than having great graphics, a better Central Processing Unit (CPU) over a good graphics card were prioritized.

The final choice became a 15" Dell Precision 5530 Mobile Workstation with a 6 core Intel i9 processor, 32 GB RAM, 512 Solid State Drive (SSD) and a Nvidia Quadro P2000 graphics card [76]. The workstation has also performed a MIL-STD 810G- test which ensures robustness in correlation to a robust and reliable system [77].

8.6 Code hosting platform

VJ

8.6.1 GitHub

Collaborating across a team on a project regarding the same software code requires a platform adapted for collaboration and control. Using GitHub, a code hosting platform allowing version control and collaboration among several users, provides an efficient workflow for monitoring the software modifications [78]. As multiple members of the group will be collaborating and adapting the same code assigned from [1], GitHub allows modification and for the different versions of code being merged into one. All group members are members of an organization called Mini Guardian on GitHub, where the GUI source code, Arudino Nano and Arduino Mega code are made as separate Git repositories. Using Git repositories is ideal for structure and provides an overview of the changes made to the different code variations.

Git is a Version Control System (VCS) hosted by GitHub and contributes to collaboration across teams by tracking changes to software [79]. Using Git allows cloning repositories to a local folder on a personal machine, giving collaborators the ability to modify code and getting the correct and updated version of the code to their computer. GitHub organizations assigns roles and permissions to certain collaborators assigned by an organization administrator. Having the software engineers of the group as admins of the organization gives the opportunity to control and permit changes as well as merges to the source code. A tutorial was made for the group members to follow along for publishing code to GitHub E.

Chapter 9

Electronics

The Electronics chapter explores various aspects related to the electronic components and systems of project Mini Guardian. This chapter begins with a detailed exploration of the zero-point azimuth, examining different cams, cam profiles and sensing options to accurately determine the zero-point in the rotational direction. Furthermore, the power of the Effector is thoroughly discussed, providing an overview of the power requirements and specifications. The chapter also explains the process of creating the electrical schematic of the System and the implemented solutions. Additionally, it includes a section dedicated to the conversion of the Fire Distribution Center into the Power Supply. The chapter features a section of the simulation of the control system, showcasing the intricacies and functionalities of the system through a simulated environment. Lastly, a vibrational analysis was performed.

9.1 Zero-point azimuth

MH

System requirement SR-07-03 as seen in table 3.5, says that the Effector shall have a fixed reference/ zero point. The reason for this requirement is that a known reference point will make the Effector start in the same position. To make the reference position for the azimuth axis a set of steps has been taken, and in this part of the report they will be described. Firstly some preliminary research will be described, it includes information about cams, and sensors. This research was then used when the cam profiles was designed, and the sensors was chosen. Furthermore this section will also go into how a second set of switches was utilized to create the hardware that stops the Effector, in case over-travel on the azimuth axis occurs.

9.1.1 Cams

In fig. 9.1 a cam is seen, a cam is a body that creates an oscillatory or reciprocating motion onto a second body when it is rotated. The second body, also seen in fig. 9.1 is a follower, the follower is in contact with the cam. When the cam rotates a motion is created upon the follower making it rise, dwell and fall over the duration of the rotating motion. Each of the different positions has different time periods which depends on the shape, also known as the profile, of the cam. The rise part of the cam is the section that moves the follower away from the center of the cam, furthermore it is the profile in this section that determines how fast the follower is lifted. As seen in fig. 9.1 the next part is the dwell section, this section lets the follower stay at the same location for a given period of time. The dwell section is created by making the cam profile have a circular radius that is not changing. Lastly the fall part is the section that moves the follower back towards the center of the cam, how fast the follower moves is determined by the profile [3].

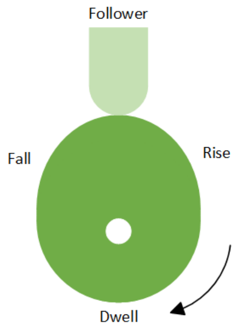


Figure 9.1: A cam and follower, created with inspiration from [3].

To produce a particular motion with the use of a cam and a follower a set of adjustments can be made, the cam can have different profiles, and different followers can be used. An example of a particular motion with the use of a point-shaped follower is seen in fig. 9.2. Here the cam profile is a circular cam with an offset on the center of rotation which is called an eccentric cam, the profile moves the follower which in turn creates a simple harmonic motion. The displacement of the follower, in other words how far the follower moves up and down, is determined by the radial distances. In particular the radial distance from the followers point of contact, onto the cam profile, to the axis of rotation for the cam. In fig. 9.2 the resulting follower motion, at various angles around the rotation of the cam, is shown in what is called a vertical displacement diagram. The diagram is drawn by taking the radial distance from the cam surface to the point of rotation and projecting the resulting distance around the profile to get the displacement at the given angles [3].

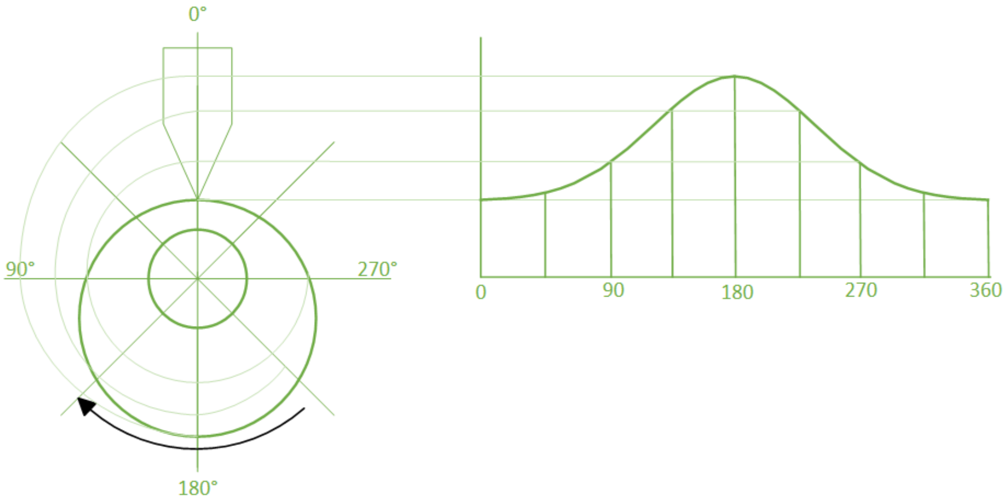
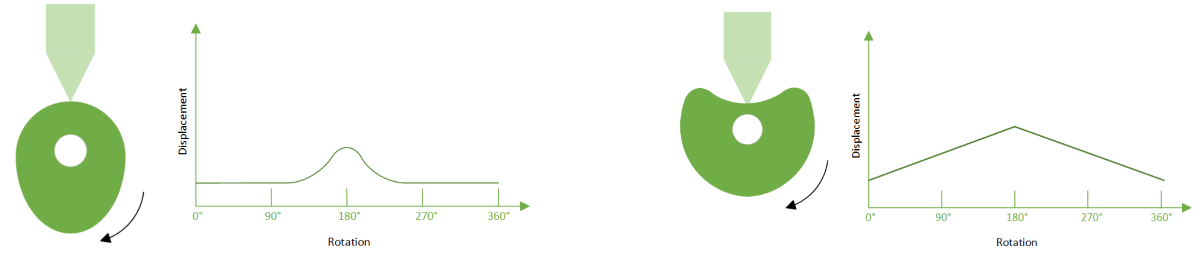


Figure 9.2: Displacement diagram for an eccentric cam profile, created with inspiration from [3].

Another set of cam profile is seen in fig. 9.3, this shows the resulting follower motion for a pear shaped cam, seen in fig. 9.3a. The pear shaped cam results in the follower motion to stay stationary for about a half revolution, and the in other half revolution it rises and falls with an harmonic motion. Pear shaped cams are often used to control valves in internal combustion engines. The heart shaped cam, seen in fig. 9.3b, gives a resulting follower motion that increases and decreases at a constant rate along with the rotation of the cam.

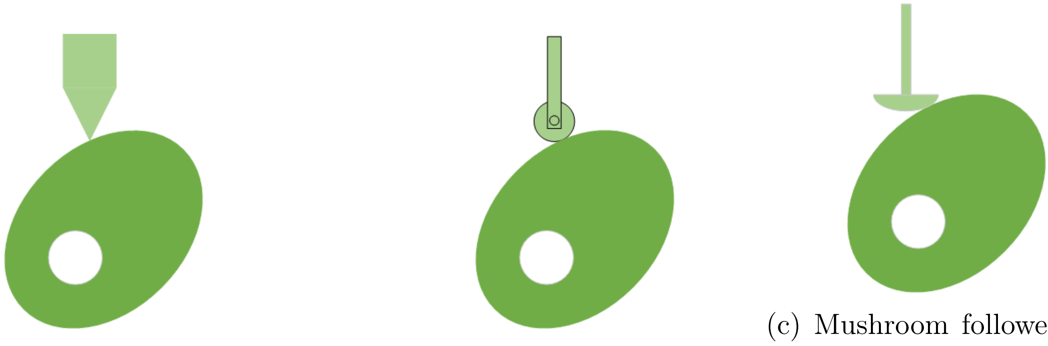


(a) Pear shaped cam, created with inspiration from [3].

(b) Heart shaped cam, created with inspiration from [3].

Figure 9.3: Pear and heart shaped cam with follower motion, created with inspiration from [3].

A set of different follower types is seen in fig. 9.4, each follower has its own benefits and drawbacks. The knife follower is seen in fig. 9.4a, a roller type follower is seen in fig. 9.4b. The roller follower are usually made up of a roller or ball bearing, as a result they create less friction than a sliding follower such as the knife follower. The roller follower are also more expensive. A mushroom shaped follower is shown in fig. 9.4c, they are often used because they are cheaper and can be made smaller than the roller follower [3].



(a) Knife follower, created with inspiration from [3].

(b) Roller follower, created with inspiration from [3].

(c) Mushroom follower, created with inspiration from [3].

Figure 9.4: A sample of follower types, created with inspiration from [3].

Cam location

The azimuth axis of the Effector is a rotational axis, making it so that an individual position, by the use of one cam, can easily be achieved. But if the axis is to have multiple positions of interest a second cam can be incorporated to make more specific positions. Seeing as an over-travel position also was of interest the option feel upon creating two cam. One cam will be mounted to the Effector motor sprocket, and the other cam will be mounted onto the bigger sprocket which turns the Effector in the Azimuth direction.

The cam plates can be seen in Figure 9.5. This is an illustration which shows the cam plates mounted onto the Effector, this is just an illustration, and was used as a guide when choosing the sensor.

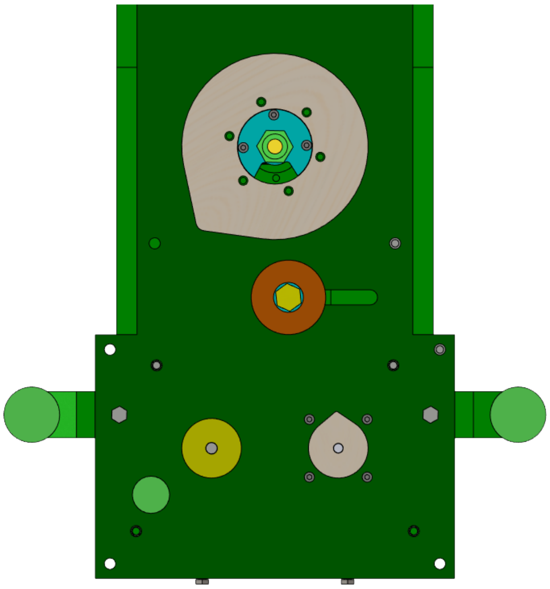


Figure 9.5: Cam plates for reference point

9.1.2 Sensing options

An important factor to look into when making the Effector have a known position where it can calibrate its measuring system, is a sensor that can actuate upon this position. There are a number of different sensors that can be used for the application, such as limit and proximity sensors. Each sensor has its pros and cons when it comes to detecting objects. Below is a list which show a set of criteria to be taken into account when choosing a limit sensor [3]:

1. If contact with the target is possible and acceptable then a limit switch can be used.
2. If the sensing distance is less than 65 mm and the target is metal then an inductive sensor can be used, otherwise if the target is non-metal a capacitive or photoelectric sensor.
3. If the target is transparent an ultrasonic or photoelectric sensor can be used.
4. If the target is a solid object a photoelectric or ultrasonic sensor can be used. If the sensing distance is greater than about 6 m the photoelectric sensor is used.
5. If the target is a liquid and can be sensed from its top surface an ultrasonic sensor can be used, otherwise a capacitive sensor.

Because the system will utilize a set of cams, which can be made out of metal, for the reference position an inductive proximity sensor can be used. A proximity sensor is built up of a coil, the coil is supplied with an alternating current. When alternating current moves through the wires in the coil an oscillating magnetic field is created. If an object made up of conducting material, in other words metallic material, moves into the magnetic field small eddy currents will build up in the object. This will in reduce the effectiveness of the oscillating magnetic field, resulting in a reduction of current moving in the coil. The sensor will detect a change in the current flow, which in turn will trigger an output signal when the change in current flow is sufficient. Proximity sensors have a typical sensing range from around 3 to 10 mm, they are both reliable and cheap, making them ideal for many applications [3].

Seeing as the proximity sensor can have various sensing ranges another form of sensor was looked as. As a result research went into a set of sensors known as a limit switch. These types of sensor are mechanically activated and comes in many different sizes and configurations. Because the sensors are mechanically activated they are in direct contact with the object they are supposed to detect, furthermore the electrical contacts inside the sensor are mechanically activated by the object. The sensor then gives and output signal in regards to the state of the switch [3].

The limit switches electrical contacts are closed or opened, in regards to its configuration, when an object comes in contact with the sensor. The object comes into contact with the actuator which is the part of the limit switch that closes or opens the electrical contacts. When the electrical contacts are acted upon a they can make or break an electric circuit. As a result the limit switch can be used to decide if an object has come in contact with the sensor, and in turn determine the objects position. Other uses for the limit switch is that it can utilized to determine when a given object has reached the end of its travel [3].

Limit switches are primary used were switch actuation and wear are minimal. The sensors are made up off a switch body, electrical contacts, and the operating head which is generally a plunger or a lever arm. When the object applies pressure to the lever arm or plunger, a resulting movement occurs. This movement causes the electrical contacts located inside the switch body to make or break a circuit. The limit switches can have different configurations according to the suitability of the applications. An example is that the lever arm can include a roller bearing, these limit switches are well suited were a sliding contact causes the bearing to rotate. A plunger style limit switch is suited for applications where short and controlled movements are required [3].

Sensor choosing and placement

Taking into consideration the benefits and the list from the book, the choice fell upon using limit switches that the cam profile can actuate upon. There are many configurations of the limit switch, and as a result the plunger style, long and short lever arm, both with and without a roller bearing, was looked into. Because our system will use two rotating cams the choice fell upon the short lever arm with a roller bearing called V-165-1C5 manufactured by Omron [80]. This is because the roller bearing creates the least amount of friction upon contact with the cam resulting in minimal wear, it is also cheap to buy, and it is compact in size.

To make sure that the limit switch sensors are located in the correct position a Matlab script has been made, this is seen in Appendix A. This takes the relationship between the motor driver gear, and the Effector gear, and uses it to plot where the limit switches needs to be placed around the Effector gear. This is in order to achieve the desired zero-point, and over-travel positions. The resulting plot is seen in fig. 9.6

The plot seen in fig. 9.6 shows that the three limit switches will be placed with

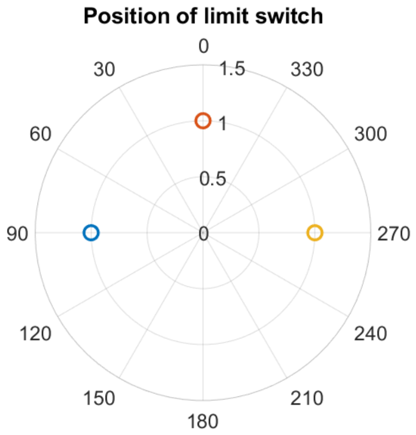
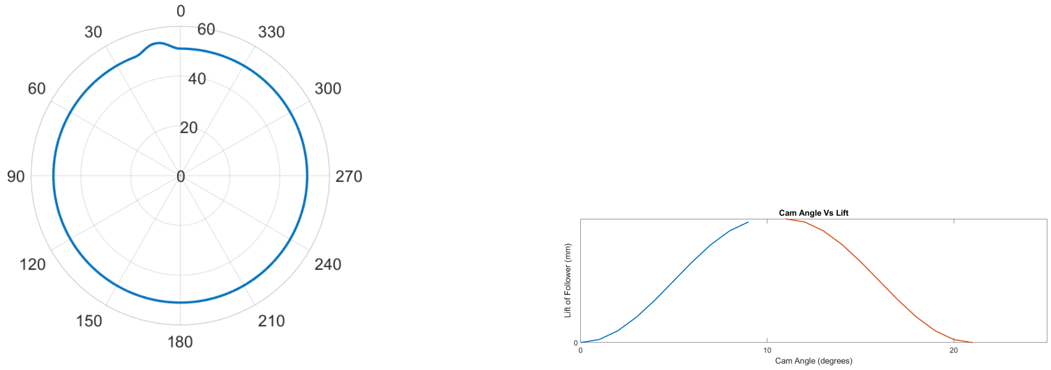


Figure 9.6: Placement of limit switches.

90° spacing between them. Each of the position will be actuated together with the limit switch on the motor driver gear at the given locations. This makes it so that each position will only give a signal to the Arduino at the desired angles.

9.1.3 Cam profile

To design the cam profiles a cam and follower displacement diagram was used, as these provide in great detail the movement information on the movement of the follower in regards to the cam profile. After researching a Matlab script was found [81], the code is seen in appendix B, this script plots the cam profile and resulting movement of the follower based on the given specifications for the cam. When the script is run, data for follower lift, base radius, ascent angle, dwell angle and descent angle is inputted, and the result is plotted. The resulting plot for the cam profile and follower movement for the Effector gear cam is shown in fig. 9.7.

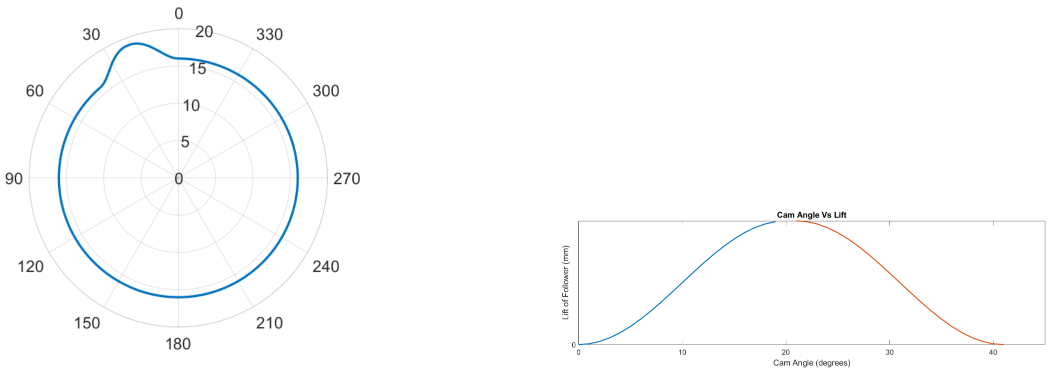


(a) Effector cam profile.

(b) Effector cam follower movement.

Figure 9.7: Effector cam profile data.

In fig. 9.7a the cam profile is plotted for the Effector gear. This profile has a follower lift movement of 3 mm, a base radius of 51 mm, an ascent angle of 10° a dwell angle of 1° and a descent angle of 10°. The following follower movement from this cam profile is shown in fig. 9.7b, it shows an harmonic movement in both the fall and rise section. Furthermore the follower has a dwell time of 1°.



(a) Motor cam profile.

(b) Motor cam follower movement.

Figure 9.8: Motor cam profile data.

In fig. 9.8a the cam profile is plotted for the Effector gear. This profile has a follower lift movement of 3 mm, a base radius of 16 mm, an ascent angle of 20° a dwell angle of 1° and a descent angle of 20°. The following follower movement from this cam profile is shown in fig. 9.8b, it shows an harmonic movement in both the rise and fall section. Furthermore the follower has a dwell time of 1° By using Matlab the cam profiles was easy to design and changes was incorporated quickly, the plotted data was extracted into a spreadsheet. The spreadsheet was then imported into SolidWorks, where the profile was sketched and the correct center of rotation was added. This was in order to get the correct hole positions to mount the cam onto the Effector driver sprocket. The cam was then cut out in plywood using a laser cutter, and it was later assembled onto the Effector The cam for the motor sprocket was incorporated onto the existing sprocket, making it one part that was 3D printed. The resulting cam configuration is seen in fig. 9.9, the 3D printed sprocket is located in the upper right corner.

9.1.4 Placing limit switches

To get the right placement of the limit switches a set of 360 ° protractors was printed out in different scales. They were then cut to shape and glued onto the bottom side of the Effector, as seen in fig. 9.9. When placing the protractors it was really important to get the location correct. The switches was fastened using double sided tape, and the cam profiles was mounted onto the Effector aswell. When all the parts were mounted, the Effector was then spun around to verify that the calculated locations were correct. To be verify that the limit switches were placed in the correct location this was triple checked. The limit switch brackets, which was 3D printed, mounting holes was marked and then the switches was removed. The marked holes was then drilled and tapped. The Effector was then assembled with the new limit switches and the desired locations for the reference point and over-travel position was checked again.

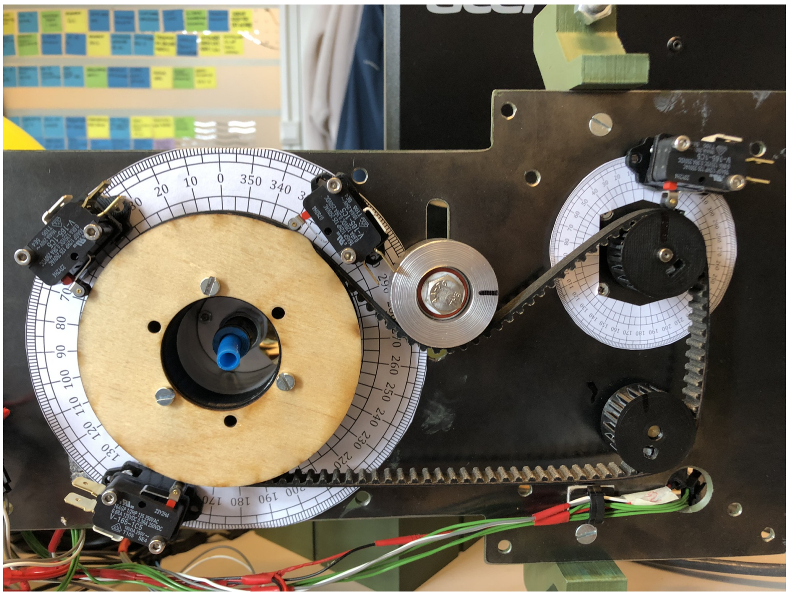


Figure 9.9: Placing limit switches

9.1.5 Wiring of switches

The limit switches has a set of Normally Open (NO) and Normally Closed (NC) contacts, because the way they are connected, they can translated into logic, and as a result Boolean algebra can be used. This is in order to show how the switches needs to be connected in order to get the wanted logic functions. The NO contacts will get letter A as an annotation, and the NC will get $\bar{A}$. In table 9.1

Table 9.1: Limit switch data for wiring.

Name	NO annotation	NC annotation
-S11	A	$\bar{A}$
-S12	B	-
-S16	C	$\bar{C}$
-S17	D	$\bar{D}$

To wire the reference position the logic in (9.1),

$$\text{Refernce position} = AB, \tag{9.1}$$

is used. Which shows that to get the correct logic for the zero-point the switches used, need to have the NO contact sets connected in series. This makes it so that when the Effector is searching for its reference position, both the -S11 and -S12 switch needs to be actuated.

Originally it was intended that the enable signal from pin 27 on the Arduino Mega would run trough the limit switch wiring logic. As a result when the over-travel position was reached the drive for the azimuth axis motor would then cut the enable signal which in turn makes it so that the motor stops. When researching it was discovered that the azimuth stepper motor is allowed to move when the enable signal is low, as result a different signal had to run trough the over-travel limit swithces. A good solution then became to run the 24V DC power to the motor driver trough the NC contact set on the limit swithces. The resulting logic for the over-travel is seen in (9.2),

$$\text{Over-travel} = \bar{\text{A}} + \bar{\text{C}}\bar{\text{D}}, \tag{9.2}$$

which show that to cut the power for the motor drive both the -S11 or -S16 and -S17 need to be actuated.

To get feedback from the system to the Arduino Mega, the wiring was performed in accordance with the logic seen in (9.3)

$$\text{Feedback} = \text{A} + \text{C} + \text{D}. \tag{9.3}$$

After all the Boolean equations were finished a proposed wiring of the switches was drawn in OrCad Capture [82], the resulting diagram is shown in fig. 9.10. Here the -S11 and -S12 switch are connected in series which creates the feedback to the Arduino that the reference position has been reached. The NC contact set on -S16 and -S17 are powering the motor driver together with the NC contact set on -S11 connected in parallel. This results in a breakage of power to the motor driver if the Effector reaches the over-travel position. Furthermore the NO contact set on -S11, -S16 and -S17 are connected to the Arduino. The full wiring diagram for the Effector is seen in Appendix A, and will be explained in the section 9.3.3

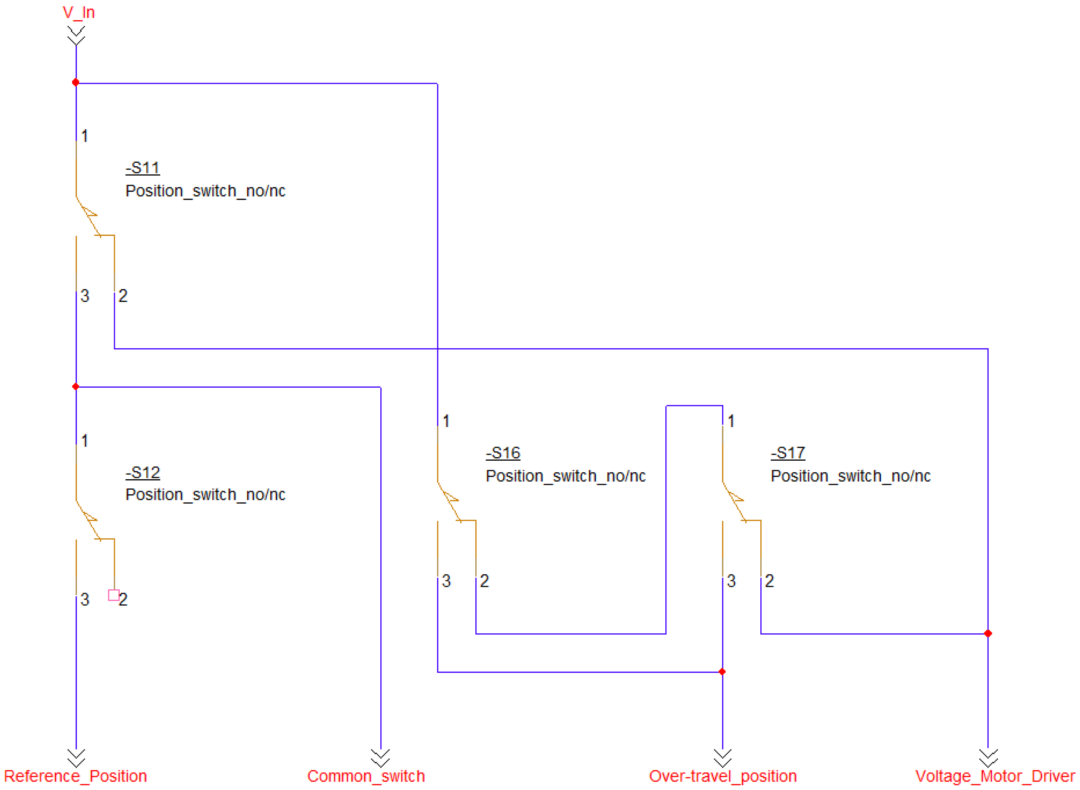


Figure 9.10: Wiring of limit switches

9.1.6 Pins on Arduino

To get important information to the Arduino Mega, a set of interrupt pins can be used. Interrupts are important signals that makes the micro controller chip on the board stop its normal operation and respond to tasks that require immediate handling [69].

To get all the information about what is happening in the azimuth axis the Arduino Mega needs to be connected to the limit switch logic. Therefore the 24V DC signals are sent into a voltage divider circuit, which will be explained later in the report. This voltage divider then sends signals, with correct voltage levels, into the Arduino. The zero-point logic is sent into Pin 39, the common switch sends information to pin 18, and the over-travel switches send information into pin 19. The reason for using pin 18 and 19 is because they are interrupt pins.

9.1.7 Programming Zero-point

The intended purpose to program the zero-point is that when the Calibration button is pressed in the GUI the Effector will start to move in a CCW direction. This makes it so that when pressing the Calibration button the Effector needs to be located sufficiently in the CW direction. Furthermore the speed with which the Effector moves, is reduced to make sure that the correct position is established. After the position is reached the Arduino Mega 2560 stores it as the new zero-point position, which is used when calculating the number of steps for repositioning. The azimuth position status is also updated in the Graphical User Interface (GUI). The code is documented in the Doxygen folder.

9.2 Power

SH

Project Mini Guardian inherited power supply issues from the previous group, where the power supply was found to be underdimensioned. This meant that it was unable to adequately meet the power requirements of the system, leading to potential operational limitations and challenges. Addressing and resolving these power supply issues became an important task for the current project to ensure the reliable and efficient functioning of the System. The process to get there is described in the following sections.

9.2.1 Power consumption

The original Effector used two power supplies to give the correct voltage to different parts of the system, the name and size can be seen in Table 9.2.

Table 9.2: Power supply

Power Supply	Nr.	Input	Max output [W]	Max output [A]
24 V power supply	LCS150US24	85-264 VAC	150 W	6.5 A
5 V power Buck	Xl4015 5A	4-38 VDC	75 W	5 A

The original Effector is made up of several different components which consume power. The name and current drawn of each component can be seen in Table 9.3.

Table 9.3: Power consumption of different components.

User	Nr.	Input V	Max current input [A]	Amount	Total current [A]
Arduino Mega	A000067	6-20 VDC	65.6 mA	1	65.6 mA
Arduino Nano	A000005	7-12 VDC	19 mA	1	19 mA
Motor driver	DC9-40V 5A	9-40 VDC	5 A	2	10 A
Azimuth motor	892-8732	2.8 VDC	1.68 A	1	1.68 A
Elevation motor	302-23-848	24 VDC	1 A	1	1 A
Encoder	RES20D-50-201-1	5 VDC	50 mA	2	100 mA
IR-detector	184-5081	3.3-5.5 VDC	15.35 mA	6	92.1 mA
RGB module		5 VDC	7 mA	1	7 mA
Valve	893-0029	24 VDC	0.1875 mA	6	1.125 A
Ethernet Arduino	A000024	5 VDC	150 mA	1	150 mA

USB-B connection to Arduino Mega

SH

To properly power the Arduino Mega, Project Mini Launcher connected an external USB-B to it from the computer. Viewing this as an unnecessary connection, project Mini Guardian aimed to solve this. Initially, the Arduino's were connected through the 5 V ports, powered by a buck converter from 24 V to 5 V, see the electrical schematic of Project Mini Launcher [1]. As the only components being powered by 5 V were the Arduino Mega and Nano themselves which respectively draw 65.6 mA and 19 mA, the encoders which draw a maximum 50 mA each, the Ethernet-shield drawing 150 mA, and the six infrared reflective sensor with a current draw of maximum 15.35 mA, the total current draw on the 5 V buck converter would be 426.7 mA.

The idea was to power the Arduino's with 12 V on the 'Vin' pins, and then use the 5 V pins as output on the Arduino's instead. This would remove 65.6 mA and 19 mA from the equation and the total current draw would be 342.1 mA out from the Arduino ports. The Mega would have to supply one encoder and the Ethernet shield, a total of 200 mA. The Nano would have to supply one encoder and six infrared reflective sensors, a total of 142.1 mA. Both well beneath the 800 mA max current draw limit that each the Arduino's can supply. After the wiring was finished we had some heat issues from the internal regulator of the Mega, so we reduced the input voltage to 7 V, as the recommended input voltage is between 7-12 V. In Table 9.4 the new Power Supply (PS) components are listed.

Table 9.4: Power supply new System.

Power Supply	Nr.	Input	Max output [W]	Max output [A]
24 V power supply	LCS150US24	85-264 VAC	150 W	6.5A
7 V power Buck	Xl4015 5A	4-38 VDC	75 W	5A
5 V power Buck	Xl4015 5A	4-38 VDC	75 W	5A

9.3 Wiring diagram

SH

Due to the need for significant changes to the wiring diagram inherited from Project Mini Launcher, the decision was made to completely redraw it. The original wiring diagram, created using Visio or a similar software, had an overall structure that was difficult to read, and the file was not available in the file system delivered to our project. This left us with the option of either starting from scratch or attempting to integrate the new parts onto the existing diagram, which would have resulted in even more chaos. Considering the plan to redo the power distribution and the lack of a systematic existing drawing, it was determined that a complete redesign was necessary. The new wiring diagram can be found in Appendix A.

9.3.1 Wiring Diagram Software

MH

To document the electrical wiring for the project a few different software applications were researched. Seeing as three of the group members are studying electronics, a few requirements were made. The requirements were that it should be easy to use, easy to learn, and free of charge or usable with a school license. The fact that the project has a limited time frame made it so that the most important requirement is that we already know how to use the software. This made it so that there were a few different options of electrical wiring schematic software applicable to us. The most promising drawing software were from SolidWorks [83], haBit [84], OrCad Capture [85], and Autodesk [86]. The software from SolidWorks known as SolidWorks Electrical Schematic is free to use with the license provided by University of South-Eastern Norway (USN). When looking into the software it seemed intuitive and easy to use. Many features are also incorporated such as existing symbol libraries. The symbol library makes it easier to draw new electrical diagrams as it saves time from having to create the symbols and parts used for the project. The big downside with SolidWorks Electrical Schematic is that no one in the group is familiar with the software from earlier usage, therefore making it unfit for the project. The software from haBit was researched because it was a known software from previous experience. This makes it so that it takes a minimal amount of time to get started drawing the electrical wiring schematic. The drawback with this software is that USN has no license, a license is expensive and it therefore became unsuitable for the project. The electrical drawing software provided by Autodesk is free for students to use, it contains many finished libraries with completed parts and symbols to use. The interface seems intuitive and easy to use, but the downside is that this software is unfamiliar for the group. The final software which was researched is Capture CIS which is made by OrCad. This software is used in a subject which the electronics students can choose in the fifth semester. With a school license from USN the software is free to use. Seeing as this software is familiar, it makes it such that little to no time is required before the drawing of electrical schematics can begin. Capture is mainly a software which is used to draw circuits intended to be printed on Printed Circuit Board (PCB). However, new parts and symbols can be created into libraries, making it so that the first time the software is used for an electrical wiring diagram, a bit of time is required to create the parts and symbols needed. After they are created the wiring can begin. Seeing as project Mini Guardian likely will continue next year the software from OrCad is seen as the best choice.

9.3.2 Standard

MH | SH

A standard was used when drawing the symbols used in the electrical schematic. The NEK 144:2017 [87] is the Norwegian standard for graphical symbols for electric- and e-communication grids. It was therefore the preferred standard. A set of symbols used in the wiring diagram of the project can be seen in fig. 9.11.

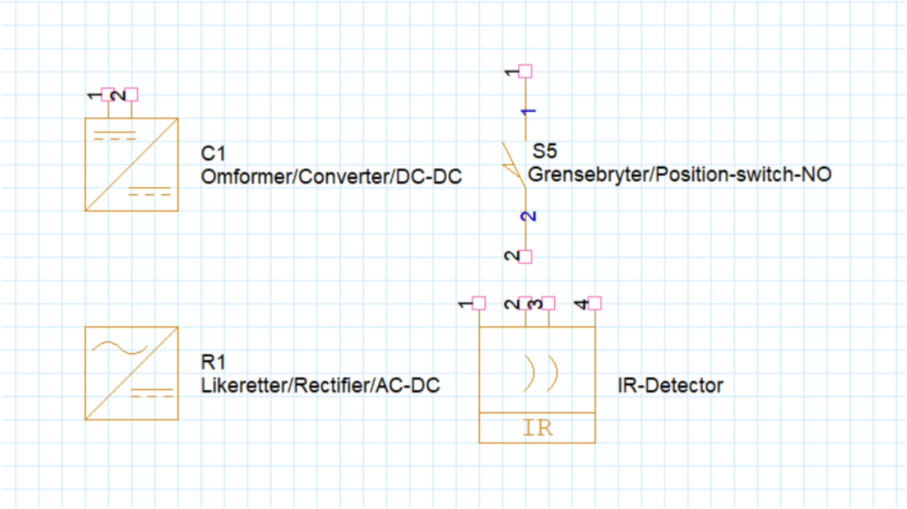


Figure 9.11: Standard symbols from NEK 144:2017.

NEK 144:2017 includes a lot of symbols. If the exact symbol one is looking for is not represented, it can often be put together by the right selection of symbols. An example of this shown in fig. 9.12. Here multiple symbols consisting of two breakers, a connection line, thermal- and electromagnet triggers and a ground fault symbol create the two pole circuit breaker triggered by thermal- and electromagnetic limits with ground-fault safety fuse.

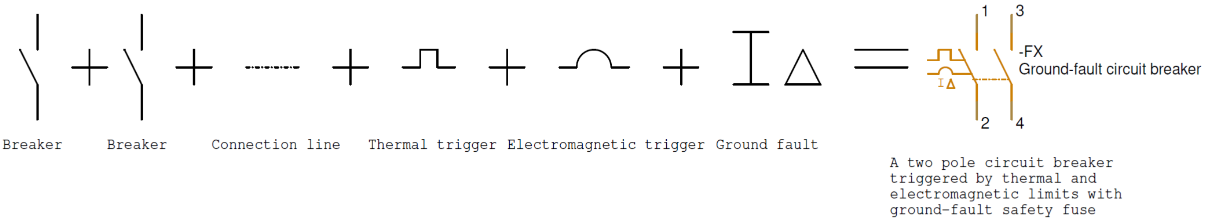

Breaker
Breaker
Connection line
Thermal trigger
Electromagnetic trigger
Ground fault
-FX
Ground-fault circuit breaker
A two pole circuit breaker triggered by thermal and electromagnetic limits with ground-fault safety fuse

Figure 9.12: Putting together symbols to create a ground-fault circuit breaker.

9.3.3 Reading the wiring diagram

SH

A lot of thought has gone into the wiring diagram. It consists of nine pages, found as mentioned in Appendix A. The front page introduces what can be found, a table of contents of sorts. The other pages go into detail in how the wiring is done. It starts at the grid supply from a socket outlet into our Power Supply (PS), where it further branches out to the rest of the System.

All components are designated a reference designator. If the symbol came directly from NEK 144:2017, the symbol usually has a standard reference designator. If not, one is created. For example circuit breakers and fuses are designated -F#. If there is no standard, e.g. the socket outlet. It is not a plug, neither a jack, but still a socket. It receives a -X with the description of what it is, often with initials. A socket outlet becomes -XSO.

As the wiring diagram consists of multiple pages, it can be difficult to comprehend which wires lead where. This is overcome with the grid references on the left side and top, going from A through D and 1 through 5. These grid references in addition to the off-page connectors should simplify the reading of the diagram. Each off-page connector features its own reference. The references are listed in Table 9.5. Upon going from one off-page connector to the next, the wire it connects to is also labeled with a /X.Y. The X is a number referring to the page it leads to/came from. Y is the grid reference number, referring to the column it is placed in, a number from 1 through 5. If the wiring diagram would have been compacted further and been less readable, the A through D lettering could be included in the Y reference. It was deemed comprehensible enough to not include the letter reference as well.

Table 9.5: References to off-page connectors and their meaning.

Off-page connector reference	Meaning
-PE	Protective earth
-L1	Grid phase 1
-L2	Grid phase 2
-PS Ethernet switch	Power Supply into Ethernet switch
-PS24	24 V power supply
-PS7	7 V power supply
-PS0	Ground from power supply
-ES0	Ground through emergency stop to reset Arduino in emergency
-PS5	5 V from power supply
-Redundant	Spare
-24V_-P35	24 V from plug 35, interrupted by a safety switch
-Driver24V	24 V to driver through limit switches

The wiring diagram also features the circuitry of two circuit boards, -X1 and -PFC1. This is more detailed than most wiring diagrams, delving more into electrical schematics. This is included to get a better understanding of what happens within the circuits and to easily identify them. As OrCAD is primarily a PCB software it mostly features the components for smaller electronic devices, so OrCAD makes wiring circuitry easier as the symbols already are in libraries and not needed to be created as the rest of the symbols.

9.3.4 Hardware emergency stop

SH

A hardware emergency stop button has been added. This disconnects the 24 V power to the motors and pressure solenoids, instantly disabling the Effectors moving parts. It is also connected to the Reset pin of the Arduino, to reset it of any code it is still trying to run. It is placed on top of the new Power Supply.

9.3.5 Voltage divider

SH

The project frequently utilizes signals that the Arduino Mega should read, but come from a source of 24 V. The input ports of an Arduino can receive a maximum of $V_{CC} + 0.5$ V on an input pin, read from the datasheet [68]. The solution for this is to utilize a voltage divider, where 5 V is taken out of the divider and sent into the Arduino as the signal. In room temperature, the input pins the project operates with have a V_{CC} of 5 V. The lower limit of a "Input High Voltage" is $0.6V_{CC}$, which equals to 3 V. Therefore the voltage divider should reduce the voltage to somewhere between 3-5.5 V. Considering these are the limits, a middle value is appropriate. Making the voltage divider of standardized resistor values is also sensible. See (9.4) for the output voltage of the voltage divider. See fig. 9.13 for the appropriate context.

$$V_{out} = V_{in} \cdot \frac{R_2}{R_1 + R_2} = 24 \cdot \frac{47k}{220k + 47k} = 4.22 \text{ V} \tag{9.4}$$

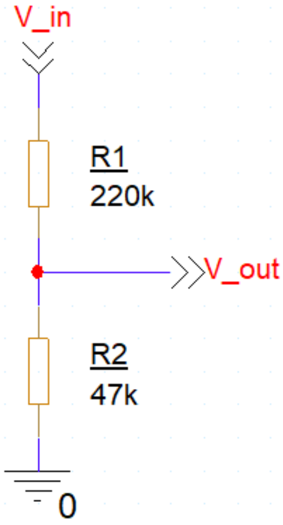


Figure 9.13: A voltage divider.

As shown a voltage divider with resistor values 220 kΩ and 47 kΩ gives an output voltage of 4.22 V. This is almost perfectly in the middle of the limits range the input pins desire.

The datasheet for the 24 V power supply used, the LCS150US24, states that it has an overvoltage protection in the form of cutting power if peaking above 33.6 V. This potentially means that the voltage divider could peak at 5.91 V, which is above the limit of the Arduino pins. To circumvent this, an overvoltage protection system is implemented into the voltage divider. A zener diode is used. Zener diodes behave like ordinary diodes up to some breakdown voltage when they become conducting. The zener diode follow the same principle as the diode, which is blocking the current in the reverse direction. However, the zener diode only does this up to the breakdown voltage. For this project, the maximum limit of the Arduino Mega is 5.5 V, such that an overvoltage protection is desired of below this value. The breakdown voltage is a limited voltage specified by the zener diode. The limit desired is 5.1 V as this is a common value for a zener diode. This zener diode will block current through it up to the limited 5.1 V it is rated for. A higher voltage and it will let the current through. As the other end of the zener diode is connected to ground, the Arduino is safe. Additionally, the zener diode acts as a protection mechanism against reverse polarity due to its low resistance for current flow in one direction and high resistance in the opposite direction. When connected with the correct polarity, it exhibits high resistance across the output, leading to a substantial voltage drop. However, in the case of reversed supply polarity, the diode exhibits low resistance, resulting in little to no voltage drop across the output.

To test this potential circuit it was simulated in OrCAD. The circuit drawn and the resulting simulation is shown respectively in fig. 9.14 and 9.15.

$$V_{out} = V_{in} \cdot \frac{R_2}{R_1 + R_2} = 24 \cdot \frac{47k}{180k + 47k} = 4.97 \text{ V} \tag{9.5}$$

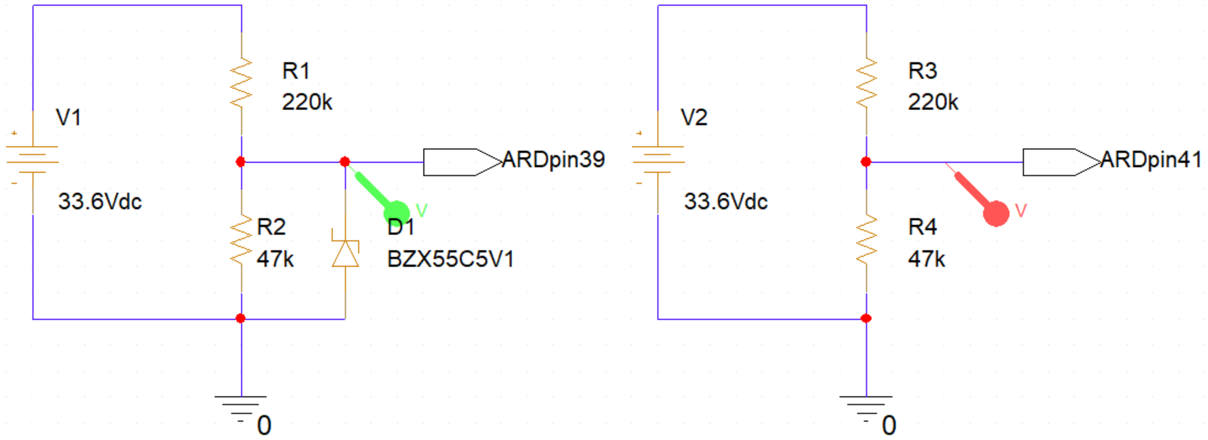


Figure 9.14: A voltage divider with overvoltage protection in the form of a zener diode.

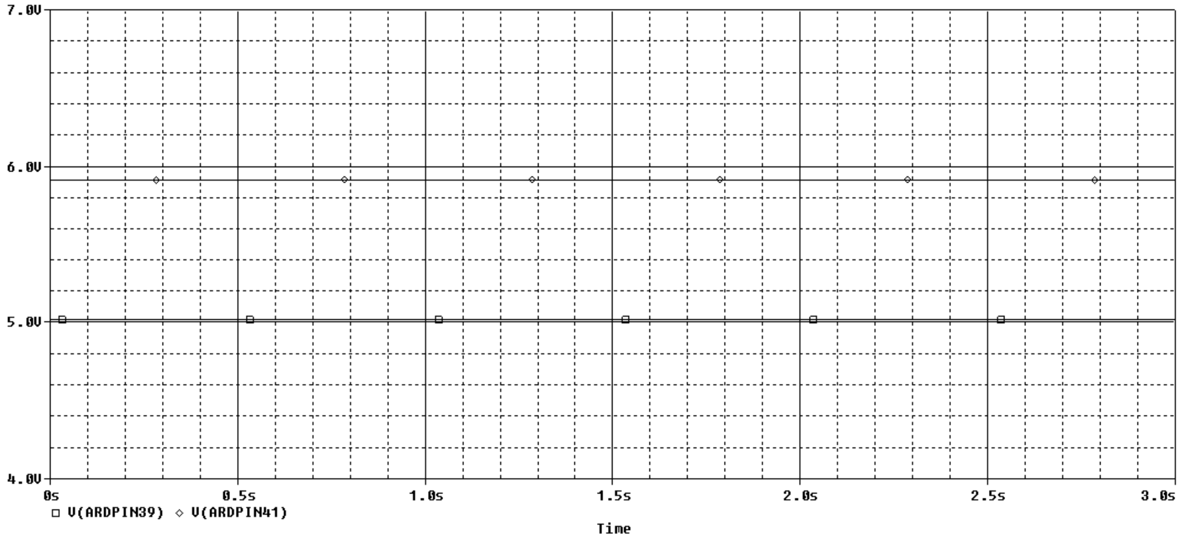
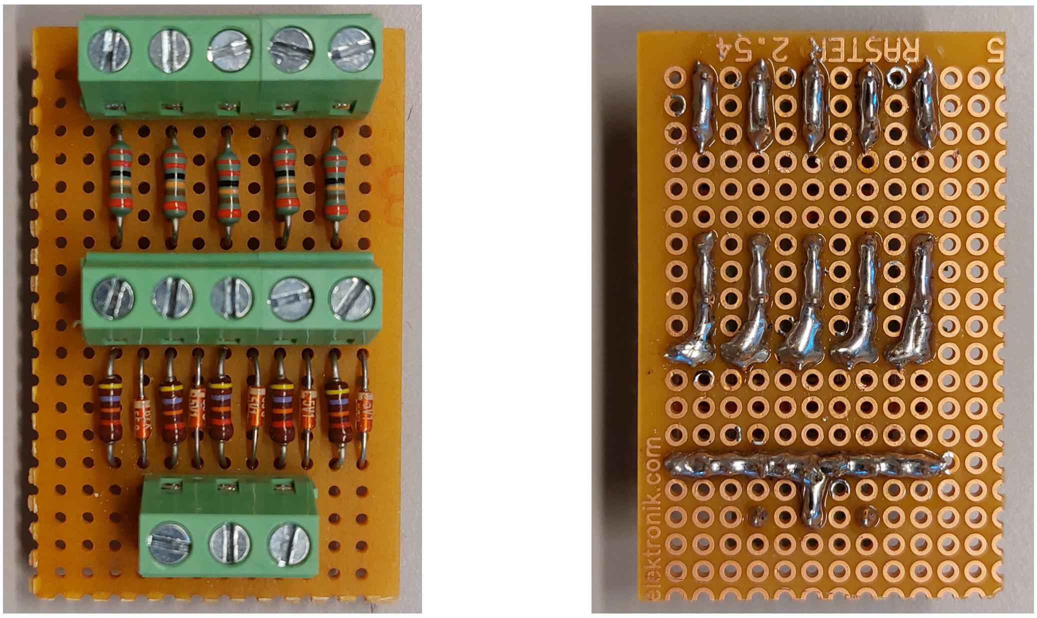


Figure 9.15: Simulation of a voltage divider with overvoltage protection in the form of a zener diode.

What was not shown in the simulation of the circuit, is that the zener diodes found in a laboratory at USN, the BZX55C5V1 has a forward voltage. Before soldering the circuit it was not physically tested on a breadboard. When testing the finished circuit board, an unexpected voltage was shown on the multimeter. It showed voltages between 3.5-3.6 V on the output of the five voltage dividers on the circuit board. The mistake of not physically testing the circuit on a breadboard, and blindly trusting the simulation - even though the exact zener diode used was found in the OrCAD library - was unfortunate. However, after the forward voltage is taken into account, the Arduino will still read a high signal as it is above the lower limit of 3 V. If it had been lower it still could have been fixed. Switching out R_1 from a 220 kΩ to a 180 kΩ would solve the problem, see (9.5).

The reason the simplest solution is to switch out the resistors and not another zener diode, is the way the voltage divider circuit board was soldered. Pictures of the finished circuit board are shown in fig. 9.16. The circuit board was difficult to hold on to. There were many things needed to be held together - whilst soldering - with a shortage of arms. Therefore some component legs were twisted in between others to be held in place. The zener diodes are the components most twisted and likely stuck. The consequence of switching out R_1 to a smaller resistor is that the output voltage will be higher. Without the zener diode it will be closer to the maximum limit, yet below it. With the zener diode it will be around the originally planned voltage level, 4.3 V.



(a) Soldered voltage divider top side.

(b) Soldered voltage divider bottom side.

Figure 9.16: Soldered voltage divider circuit.

For easier soldering, cleaner design with less noise, the prototype circuit board can be made into a Printed Circuit Board (PCB). This has been designed in OrCAD and can be seen in Circuit board CAD, section 9.3.7.

9.3.6 Rewiring

SH

When the wiring diagram was in order and most of the material needed was collected, the rewiring could begin. Due to budgetary reasons it took a lot of time to find cheap enough components if they had to be bought, but a lot of time was invested in finding parts available to us through other sources. We have scoured USN, KIC and workplaces to find what we needed. An overview of the System can be seen in fig. 9.17.

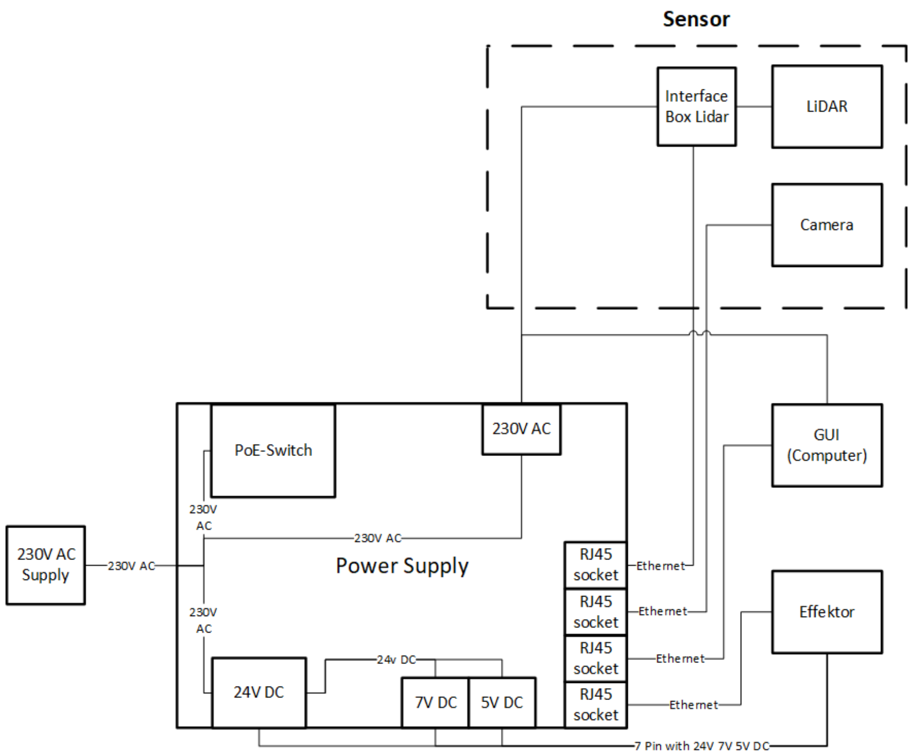


Figure 9.17: Block diagram of the System.

While wiring, it became consequently clear in what shape the Effector was in. It seems it was a rushed endeavor. The Effector uses a Molex system as connections between wires. The crimping of the male and female plugs on the wires were mostly performed wrong, leading to wires breaking off the plugs. Also, often when disconnecting the Molex plugs and pushing them back together afterwards, the plugs could pop out. Sometimes fully and visibly, other times just a bit and almost not visible at all, giving this project trouble when troubleshooting.

The first problem that occurred after the rewiring was that the 24 Vpower supply - LCS150US24 - would shut down after startup. It seemed weird as the first time we connected the System after rewiring it worked for 5-6 minutes before shutting down. After that it would shut down immediately after trying to turn it on. We had connected the Effector to Protective Earth (PE) from the grid. The Effector chassis is made of conductive parts, so it should be grounded. The exception is if it is double insulated, which it is not. The 24 V power supply sensed the PE as an error and went into a shut down mode trying to protect the System. The reason for it doing this is unclear, but it could be linked to our office space - Innovasjonsloftet - having different potentials between each phase to ground. It was measured to be 106 VAC between one phase to ground and 148 VAC to the other phase and ground, pointing towards the idea that there might be a ground fault in the building. The potential that should be between each grid phase and PE is $\frac{230\text{ V}}{\sqrt{3}} = 132$ V.

Another problem after getting the System to initially thought as functioning, was when starting to rotate or elevate the Effector. Unwanted messages started filling the GUI, overloading it and crashing it after a few seconds. At the same time multiple firing canisters did not function properly either. The problem was treated as a intertwined problem at first. The Effector was picked apart, testing each component individually, going systematically forward testing and rule out components one by one. The problem with the canisters not functioning properly can be read about in Projectile firing circuit.

After three days of troubleshooting the System in a structured and clear fashion, a conclusion of what was wrong was made, and some more rewiring performed. Now the 5 V buck was introduced again, removing all load from the outputs of both Arduino's, powering the encoders and infrared reflective sensors through the buck. The conclusion is that the internal regulator in the Arduino Mega overheated in such a way that the power output it could deliver was reduced drastically to not be able to deliver the 5 V encoders enough power (although only needing a maximum of 50 mA each). This made them unstable which led to the overload of messages filling the memory and ultimately crashing the GUI. After this solution was implemented the System worked satisfactory.

Projectile firing circuit

SH

When looking at the problem not in an intertwined fashion, but as potentially multiple faults happening (and revealing themselves) at the same time, a further look into the projectile firing circuit was performed. A picture of it can be seen in fig. 9.18.

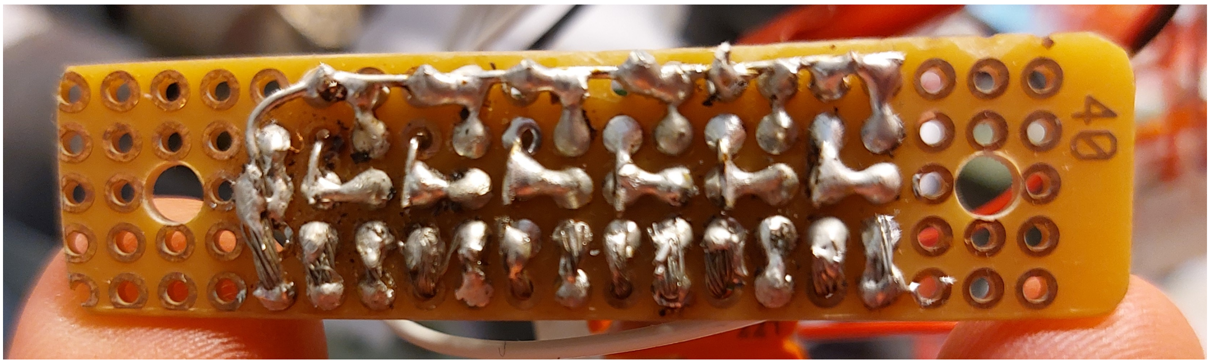


Figure 9.18: Soldering of the old projectile firing circuit.

As the picture implies, the soldering was not perfect. A strand from one of the cords is touching another connection among others. However, this was not the only problem. The previous design and what has been documented from Project Mini Launcher can be seen in fig. 9.19.

What this circuit fails to show is what it is connected to through Q1. It is connected to the solenoid valve that opens for the air pressure when firing, an inductive load. When the power supply of the solenoid is suddenly removed (turned off) a voltage spike occurs because the current flowing through the solenoid can not change instantly. Implementing a flyback diode over the solenoid will solve this problem. It will prevent the voltage spikes from occurring when the power supply is disconnected. The solenoid stores energy which has to go somewhere [3]. Without the flyback diode it will go through the transistor, potentially damaging it, which is what has happened with the previous design. The new implementation can be seen in the wiring diagram in Appendix A.

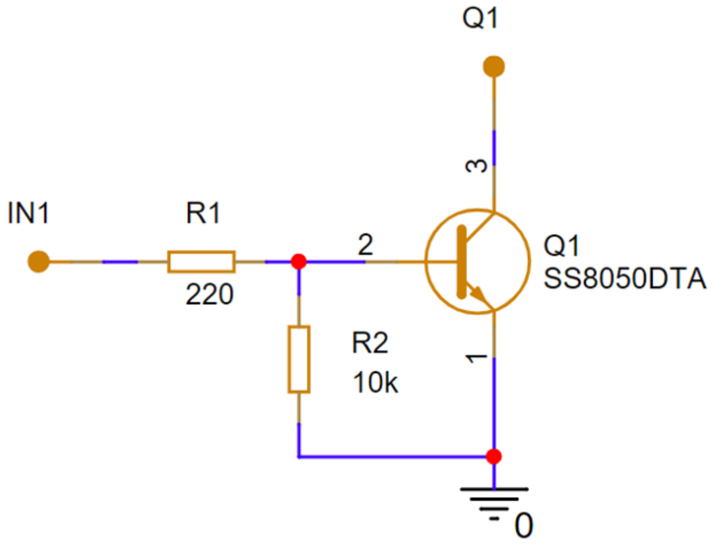


Figure 9.19: Project Mini Launcher projectile firing circuit design.

Since it was poorly documented from the previous project, it was discovered in a late phase of the project. A new one is therefore not made as of delivery of this report, but will be after delivery.

Troubleshooting

SH

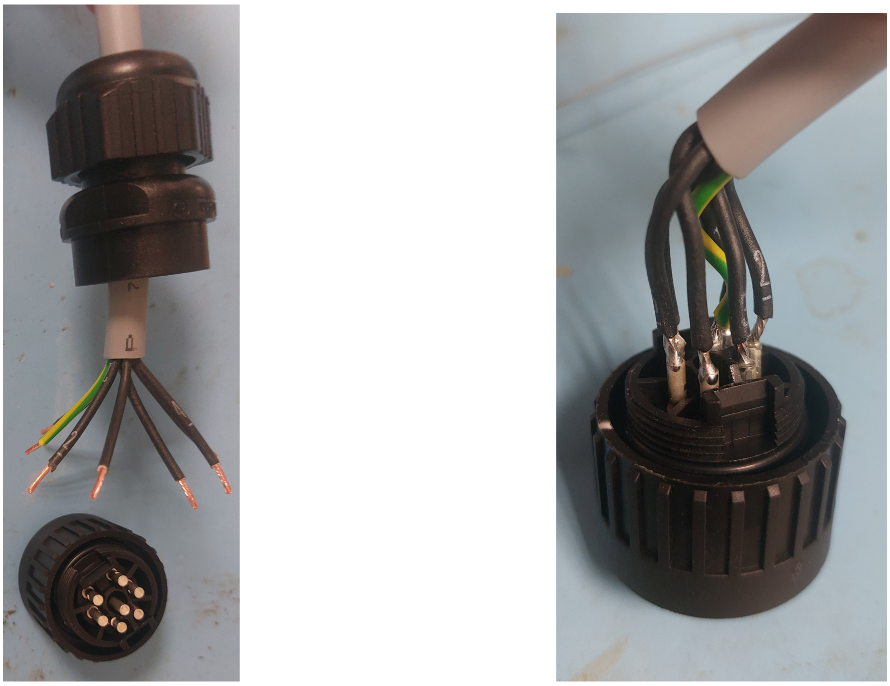
In general, it is difficult to troubleshoot electrical systems when the addition of new components or changes in the system result in the breaking or malfunctioning of other elements. This "domino effect" can make it challenging to identify the root cause of the issue and fix it effectively. Each time a new element is introduced or modified, the interaction introduces other consequences like unexpected failures or performance issues. Troubleshooting these situations require a systematic approach, careful analysis of the Systems behavior, and an understanding of the interdependencies. By methodically examining the changes made and their impact on the overall Systems, we were able to unravel the issues and restore the functionality of the electrical system.

Wire dimensioning

SH

Wire dimensioning is important in electrical systems as it ensures the safe and efficient transmission of electrical current. Properly sized wires help prevent issues such as overheating and electrical failures. Undersized wires can lead to excessive resistance, resulting in energy losses and potential damage to equipment. However, oversizing wires can be inefficient and unnecessarily increase costs. By selecting the appropriate wire gauge based on the current requirements of the system, wire dimensioning ensures the performance, reliability and safety in electrical installations.

The main consideration was to use the same American Wire Gauge (AWG) as Project Mini Launcher [1]. They concluded that an AWG 22, corresponding to a 0.327 mm^2 and can deliver a maximum of 7 A. Previously though, the connecting cable between their FDC and the Effector was an audio cable. Project Mini Guardian has changes this cable to a 6+1 Ölflex cable with a cable dimension of 0.75 mm^2. A build of this cable can be seen in fig. 9.20 and fig. 9.21. The amount of wires was chosen to ensure redundancy, as the bachelors project will likely be continued later. With redundancy the System is easier to develop further in the future. When introducing the 0.75 mm^2 into the System through the connecting cable, we also used this dimension to rewire the the motor wiring as this has the potential to draw the most power. For chassis wiring the 0.75 mm^2 can handle 15 A.



(a) Cable ready for soldering.

(b) Soldered cable in plug.

Figure 9.20: Process of making cable between PS and Effector.

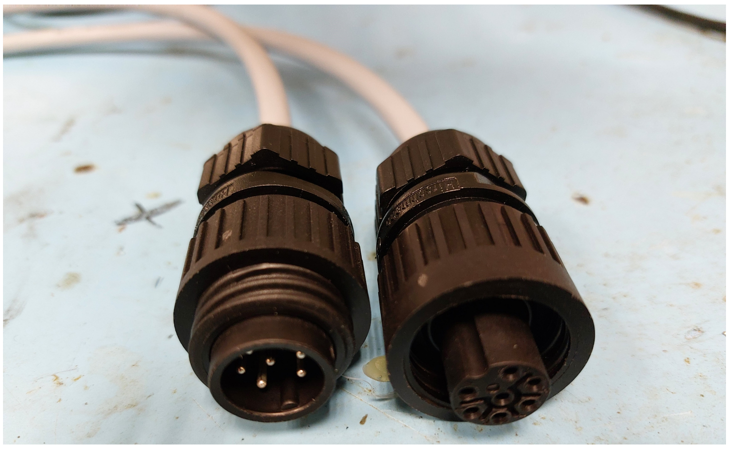


Figure 9.21: Finished cable.

The cable is made intentionally to plug the male contact pins into the PS and the female plug into the Effector. This is due to safety reasons as open connections that could be touched by for example a finger should not be able to have power on them. With this solution it can not.

9.3.7 Circuit board CAD

SH

As mentioned, OrCAD was utilized to make the prototype circuit board into a Printed Circuit Board (PCB). As it takes time to produce and receive such a PCB it was not ordered, but if wanted, next years group can order and purchase it - or expand on it, the files are available. The layout is planned in such a way that it has a streamlined design. The screw terminals are in place such that there are many options for which wires can be connected. It is shaped intuitively, voltage in at the top, voltage out in the middle and ground at the bottom, as one is familiar with when drawing a voltage divider in general.

The padstacks and footprints were previousely made for another project. The padstacks are made is such a way that they are easy to solder onto by hand. The screw terminal blocks decide the width of the PCB, so bigger padstacks makes it more comfortable to work with. It could be argued that the PCB should be smaller as it should fit into a confined space inside the Effector, but then a different option than the ruggedness of screw terminal blocks needs addressing. When screw terminal blocks are favored, the padstacks can be a bit bigger for easier soldering. The footprints are designed for intuitive placing of the components. The wiring diagram in Appendix B should be used when soldering, but intuitive footprints makes it easier.

The constraints for the project file are listed in Table 9.6.

Table 9.6: Constraints for PCB design.

Constraint	Value
Min line width	0.635 mm
Min line to line spacing	0.305 mm
Min line to pad spacing	0.305 mm
Min pad to pad spacing	0.305 mm
Route keepin	1 mm

The producer in mind when ordering the PCB was SmartPrototyping [88]. They have a set of constraints for producing the PCBs. The constraints for this project is well within the limits. The line width of the tracks in the design is 1.27 mm, equivalent to 50 mils, as this is recommended for power tracks. The constraints are set in the file to be reminded of when doing something wrong. The thickness of the PCBs cross section is altered. It is as a standard set to 0.2032 mm in OrCAD. The standard for a two-layer PCB is 1.6 mm. It could have been made as a one-layer PCB, but two layers was chosen due to better current carrying capacity and less noise. This was placed in the bottom layer.

The Gerber-files and drill file (needed to order the PCB) are viewed in ViewMate [89] and shown in fig. 9.22.

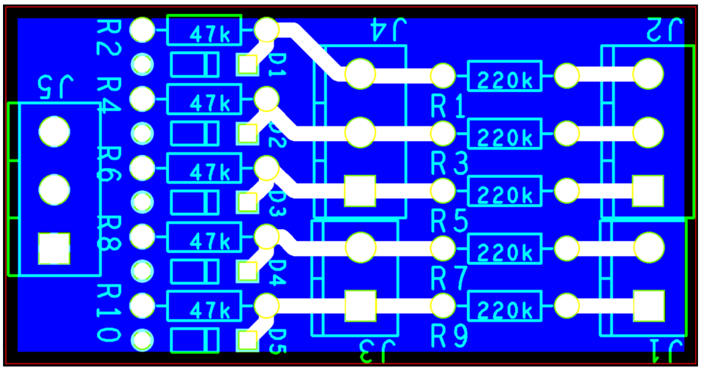
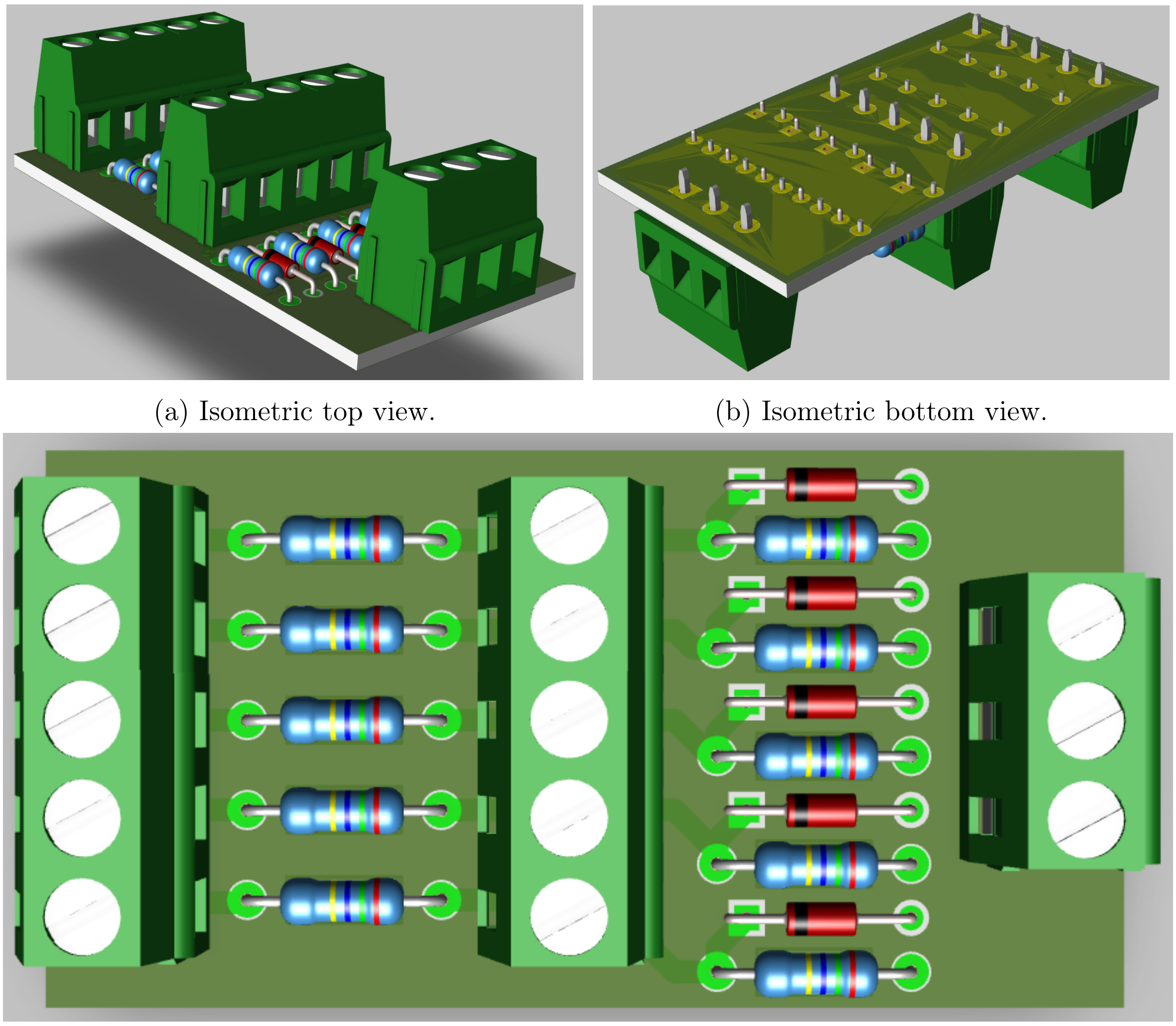


Figure 9.22: Gerber-files for the design.

The 3D-model of the PCB is shown in fig. 9.23 from different angles. Note that the resistor color code is not correct and is for illustrative purposes only.



(a) Isometric top view.

(b) Isometric bottom view.

(c) Top view.

Figure 9.23: PCB 3D-model.

9.4 Power Supply

SH

The decision to remodel the Fire Distribution Center into a dedicated Power Supply was driven by several key factors. Firstly, the removal of the air pressure tank necessitated a reevaluation of the FDCs (box's) purpose. Additionally, it was already planned to improve on the power supply to the Arduino's, so a revamp of the FDC had to be done either way. Due to project requirements, safety features are important. A red emergency button to stop the moving parts of the hardware has been incorporated. It is placed on top of the PS. As it is meant to be struck the PS box needs to be robust to withstand the impact without taking damage. It was therefore made out of metal instead of wood. As a result, the FDC was transformed into strictly a Power Supply. Incorporating a socket outlet and Power over Ethernet (PoE) switch inside the box as well, makes it so that the System can draw all the cables needed out from a single source, the Power Supply.

9.4.1 Modeling of the Power Supply box

KC

The new Power Supply has been modeled in SolidWorks, based on the previous FDCs dimensions. Since there were no CAD-model of the FDC, the Power Supply was made from scratch. Simplified models of the components to be included in the Power Supply were also made and arranged in their respective places in the model. This was done to make sure every component would fit within the Power Supply at a reasonable distance from each other. To make sure every component could be fastened at the desired place, holders for some components were also designed. These were to be 3D-printed. The holders, and the components not in need of one, was placed on a base plate and fastened through holes in this plate. The base plate was designed to be laser cut from 6 mm plexi-glass. The Power Supply itself was to be built in aluminum. Fig. 9.24 and 9.25 shows the final model of the Power Supply in ISO-view and from above.

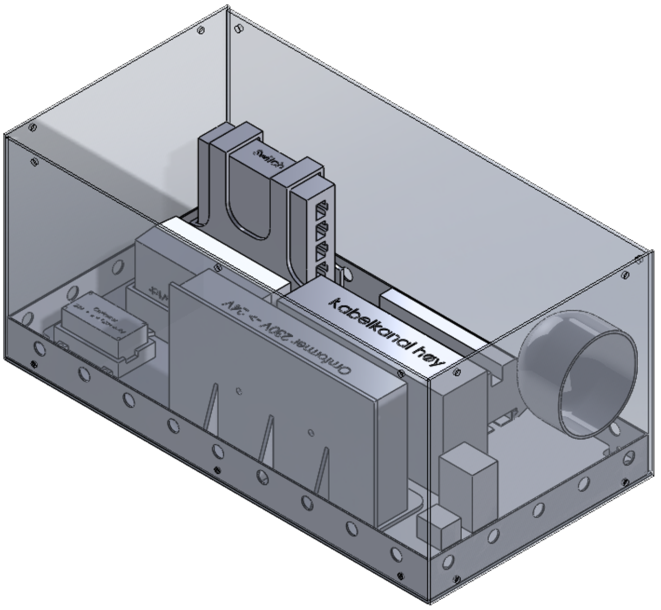


Figure 9.24: CAD-model of Power Supply seen in ISO-view.

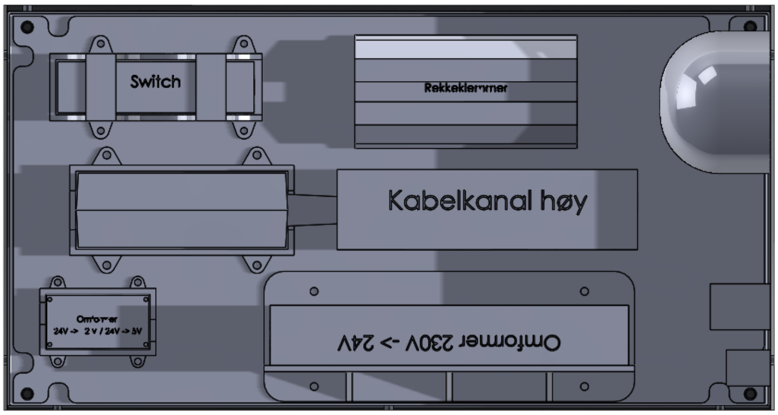


Figure 9.25: CAD-model of Power Supply seen from above.

9.4.2 Modular removable bracket system

SH

For easy extraction of each component a modular bracket system was developed. The concept was to use inserts in a lasercut Plexiglas plate, with 3D-printed brackets tailored to fit the components in them. The brackets were designed with screw holes that aligned with the inserts, enabling simple attachment. This makes it so that every component can be removed individually from the Power Supply box, without the need to disassemble the entire Plexiglas plate to get access to the underside, where the alternative was to have nuts instead of inserts, and finally remove them when access to the nuts had been granted.

A picture of the insides of the Power Supply is shown in fig. 9.26. The components inside the Power Supply are listed below:

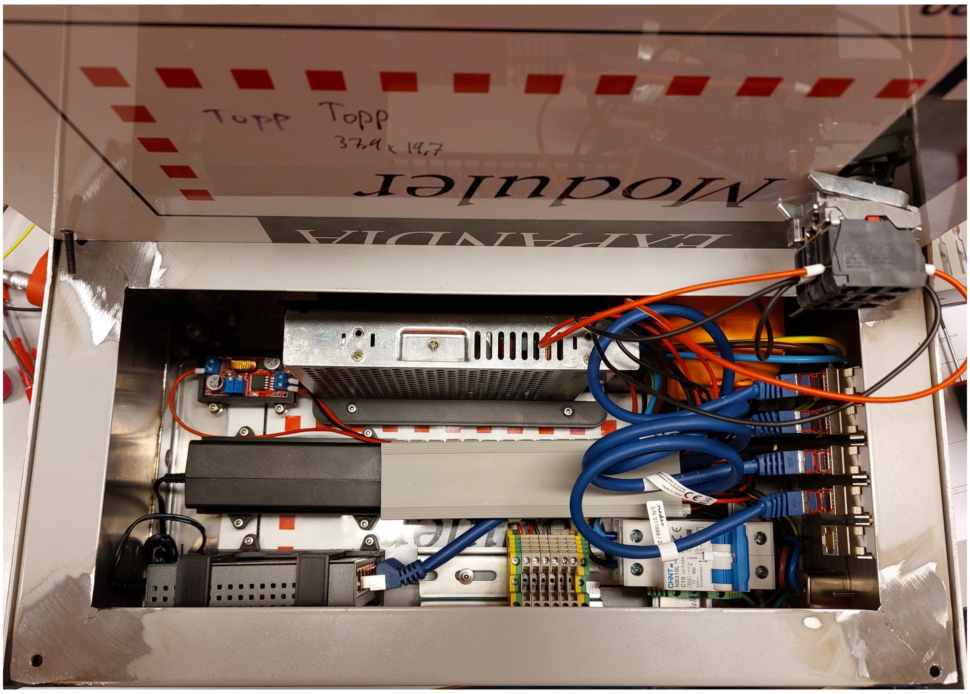


Figure 9.26: Inside the Power Supply box seen from above.

- 24 V power supply
- Two Buck converters, one for 5 V and one for 7 V
- PoE switch
- Power supply for the PoE switch
- Six terminal blocks and three PE terminal blocks
- Cable channel for cable management
- Five through wall Ethernet connections
- Socket outlet
- Grid input plug
- Grid power on/off switch
- Ground-fault circuit breaker
- Emergency stop button

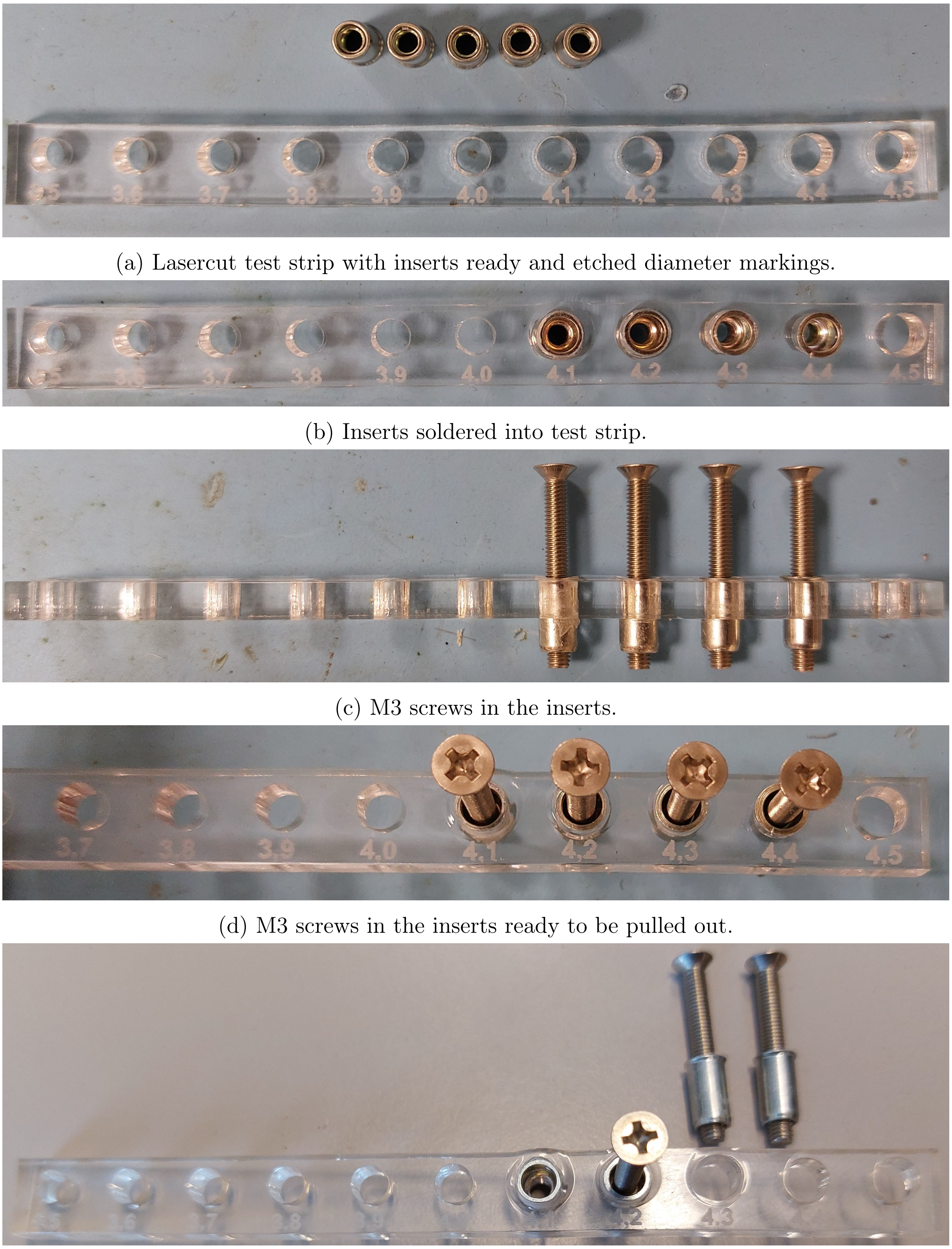
Laser cutting plate for inserts

SH

To make the lasercut Plexiglas plate with inserts, the hole diameters had to be tested. Since the lasercutter burns a hole in the plastic, the input diameter in Solidworks in not what comes out of the lasercutter in reality. The test performed is shown through the pictures in fig. 9.27.

The test was performed using a soldering iron to burn the inserts down into the plastic. Only the hole diameters with potential were used, not all of them as the inserts acquired were limited. After the inserts were in place, M3 screws were screwed into the inserts.

The screws were then attempted to be pulled out and a non-scientific way. An adjustable wrench was used to pull out the screws. The inserts in diameter holes 4.3 mm and 4.4 mm came out, whilst the inserts in the holes with a diameter of 4.2 mm and 4.1 mm stayed in place. As the 4.1 mm hole displaced a lot of plastic, and 4.2 displaced little to none, but the insert still being stuck after the pull test, the 4.2 mm hole was deemed as the best fit. The Plexiglas plate to be lasercut would be cut with 4.2 mm holes.



(a) Lasercut test strip with inserts ready and etched diameter markings.

(b) Inserts soldered into test strip.

(c) M3 screws in the inserts.

(d) M3 screws in the inserts ready to be pulled out.

(e) Pulled out M3 screws with inserts, leaving the survivors.

Figure 9.27: Test process of which diameter of lasercut hole fits the inserts.

9.5 Control system

DS

The control system setup can be viewed in fig. 9.28. It is set up in the Simscape Electrical in Simulink [90]. This package is a good tool to develop control systems and testing of system-level performance. The models can be parameterized using Matlab variables and expressions, whilst the control systems are designed in Simulink. Paramters such as mechanical, hydraulic, thermal and other physical systems can be integrated into the model using the components from the Simscape products. To deploy models to other simulation environments, including hardware-in-the-loop systems, Simscape Electrical supports C-code generation.

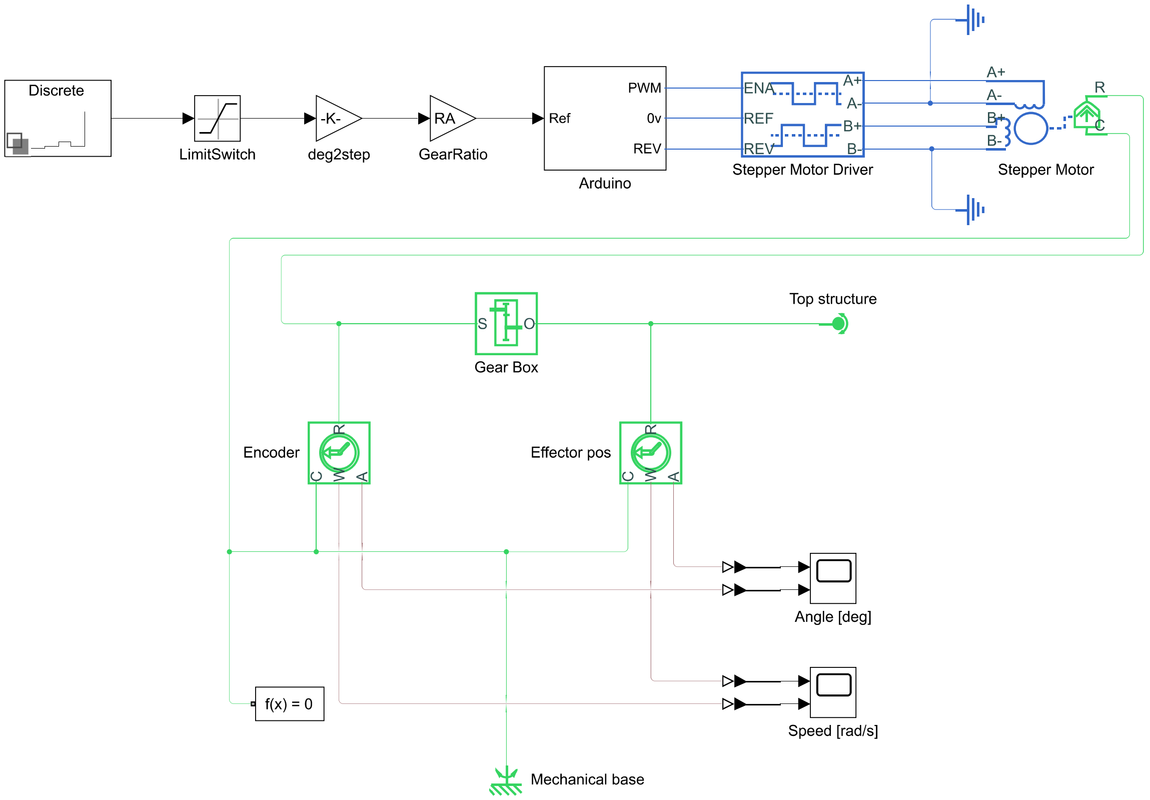


Figure 9.28: Stepper control simulation setup.

Fig. 9.29 shows the simulation generated by the setup shown above. The Matlab code for this environment is shown in Appendix F. In yellow is the angle of the Effector position, whilst in blue is the angle of the the Encoder in the azimuth direction.

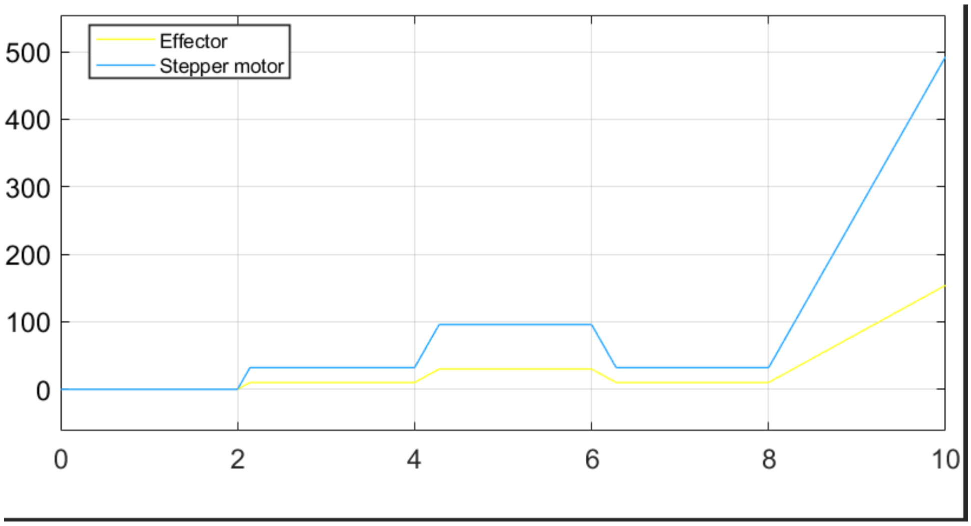


Figure 9.29: Simulated angle.

Fig. 9.29 shows the simulation of the simulated speed. In yellow is the Effectors speed in rad/s and the Encoders speed in rad/s is shown in blue. For both the simulations, the ratio between Effector and Encoder is 3.1997, which is the expected ratio as this is the gear ratio between the two.

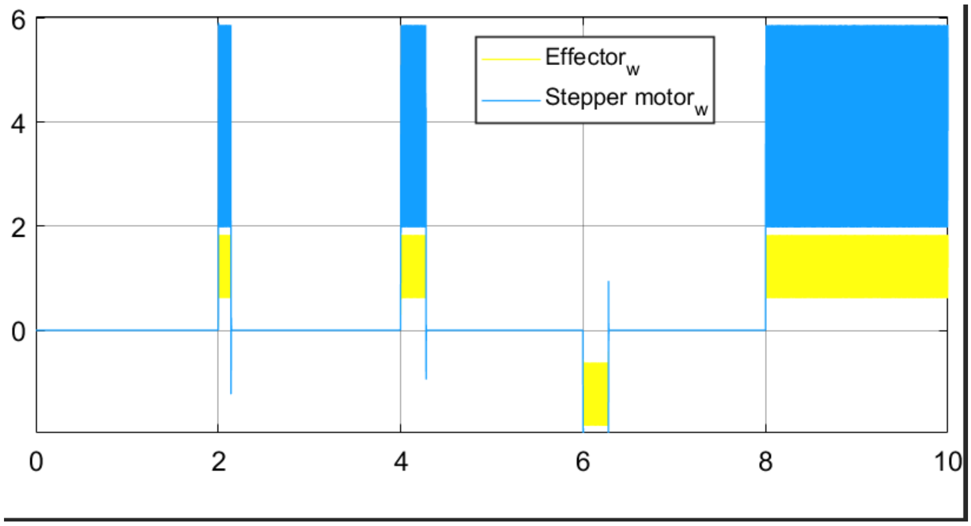


Figure 9.30: Simulated speed.

9.6 Vibrational analysis

MH | SH

This section of the report will go into the vibrational measurements that was carried on the Effector. The reason for these measurements is that the Finite Element Analysis (FEA) can perform a computer generated vibration analysis on modeled objects. To compare theoretical vibrations with the physical vibrations on the Effector a vibrational measurement has to take place. A Matlab script was written in order to analyze the accelerometer data.

A vibrational analysis can be performed by shaking a mechanical structure with an exciter and the measure the resulting vibrational motion. The response is then analyzed, furthermore the resulting signal can be a chirp, and impulse or made up of random noise. To excite the mechanical structure a shaker table, an eccentric mass and other tools can be used [91].

9.6.1 Equipment used for measuring

To measure the vibration a accelerometer was used. An accelerometer is a device which can be used to measure the acceleration of the structure which it is fixed to. The accelerometer has an internal seismic mass, that moves relative to the housing, and hence relative to the structure whose acceleration is to be measured, an example figure of a piezoelectric accelerometer is shown in fig. 9.31. Which show the mass and piezoelectric crystal, the piezoelectric crystal develops an electric charge when it is subjected to compression or shear deformation. The electric charge produced is minimal and as a result an amplifier has to be used [92].

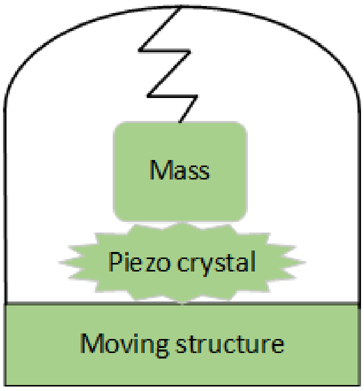


Figure 9.31: Illustration of piezoelectric accelerometer.

Dytran DY 3224A1 accelerometer

Our internal supervisor provided the group with a Dytran series 3224A1 accelerometer [93], this is sensor has a planar shear element [94]. The Dytran 3224A1 is an ultra miniature teardrop Integrated Electronics Piezo-Electric (IEPE) accelerometer, that is designed to perform vibration measurements. Furthermore it uses adhesive, such as beeswax, to be mounted onto structures. A good adhesion is important as vital data can be lost if the sensor is adhered improperly [93]. The used accelerometer with beeswax lump is shown in fig. 9.6.1.

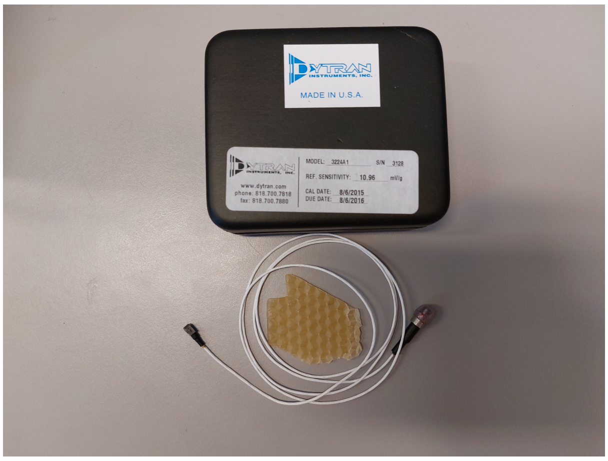


Figure 9.32: Dytran DY 3224A1 accelerometer

Dytran DY 4105C amplifier

To amplifier the signal given from the accelerometer the internal supervisor provided the group with a Dytran DY 4105C amplifier [95], the 4105C is a single channel, battery powered portable unit that is designed to be used with IEPE accelerometers, the amplifier is shown in fig. 9.33.

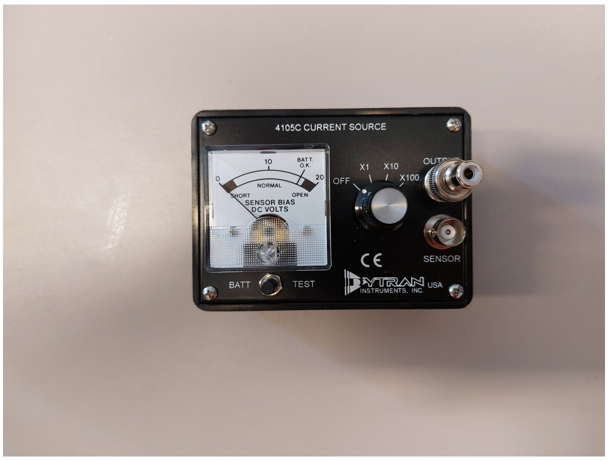


Figure 9.33: Dytran DY 4105C amplifier

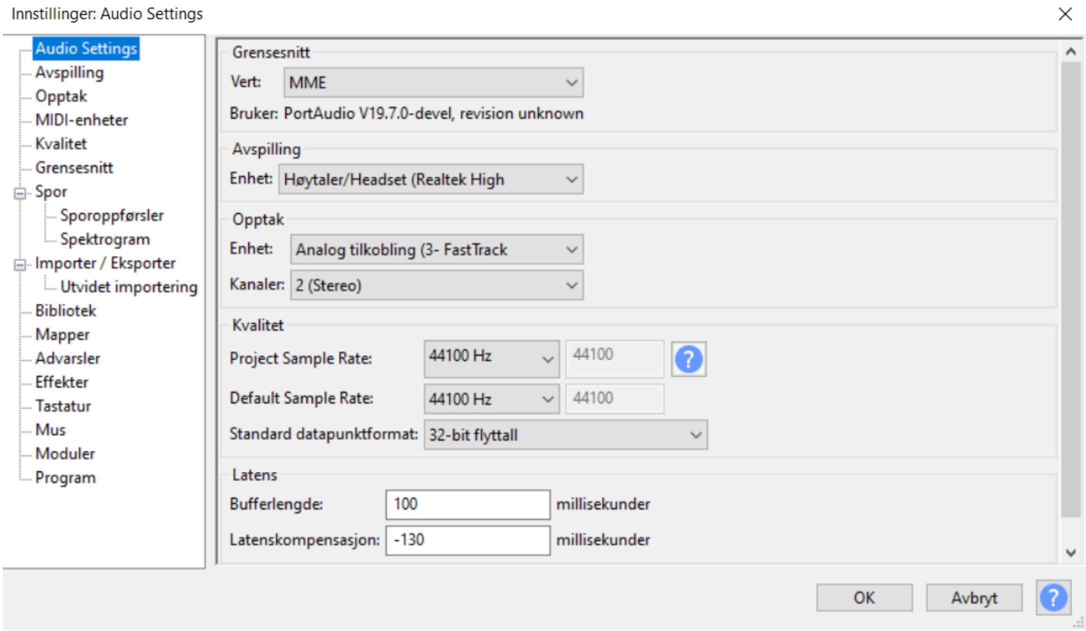
M-AUDIO Fast Track Pro

To make the recordings usable for a computer the measurement was put trough an external sound card called M-AUDIO Fast Track Pro [96]. The Fast Track Pro is a portable audio and Musical Instrument Digital Interface (MIDI), and the Dytran DY 4105c amplifier was connected with the use of coaxial cables. The sound card was then connected to the computer using a Universal Serial Bus (USB) A to USB B cable.

Audacity

To record the measured vibrations Audacity [97] was recommend by our internal supervisor. We also looked at Adobe Audition [98], but this software seemed more complicated, and we struggled to get the software to pick up the measurements from the accelerometer.

As a consequence Audacity was the preferred choice for the recording software. Audacity is a free open source audio recorder, editor and mixer that can be used on most operating systems. Audacity is very intuitive and easy to use, and it can work with a wide number of file formats used for audio recordings [99]. The settings used when configuring the recording device in Audacity is shown in fig. 9.34.



Grensesnitt

Vert: MME

Bruker: PortAudio V19.7.0-devel, revision unknown

Avspilling

Enhet: Høytaler/Headset (Realtek High

Opptak

Enhet: Analog tilkobling (3- FastTrack

Kanaler: 2 (Stereo)

Kvalitet

Project Sample Rate: 44100 Hz 44100

Default Sample Rate: 44100 Hz 44100

Standard datapunktformat: 32-bit flyttall

Latens

Bufferlengde: 100 millisekunder

Latenskompensasjon: -130 millisekunder

Figure 9.34: Settings in Audacity

9.6.2 Measurements

SH

Using the equipment described in section 9.6.1, the measurements were performed in a set of five. The five locations measured are shown in fig. 9.35. The accelerometer was adhered using honeycomb. A measurement setup is shown in fig. 9.36. Here a measurement in position 1 is performed. One set of measurements consisted of elevating the Effector from 0° to 30°, lower it from 30° to 0°, rotate it from 180° to -180°, rotate it from -180° to 180°, fire the canisters in sequence at 0° and fire the canisters in sequence at 30°. An excitation signal which shakes the Effector at the start is to be expected as this is during the starting phase of the electrical motors.

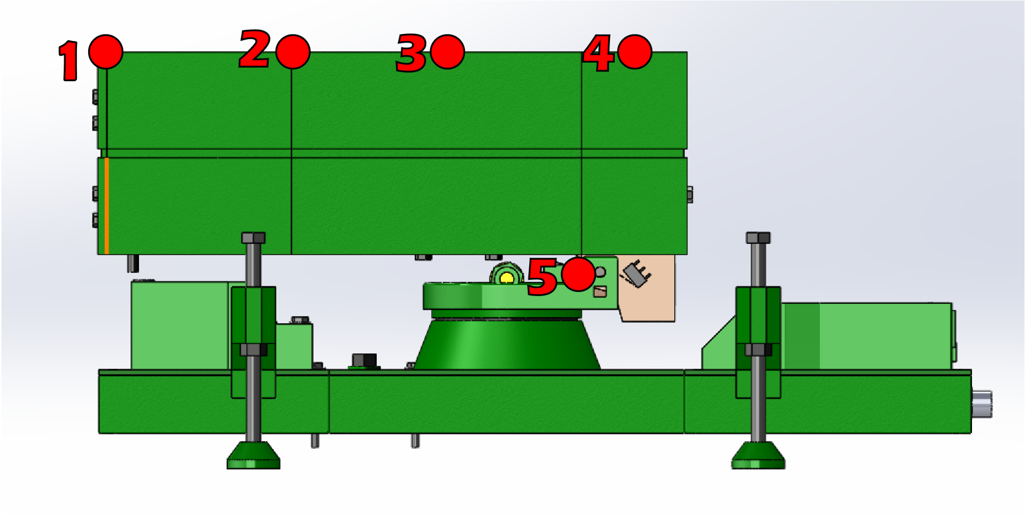


Figure 9.35: Measurements positions

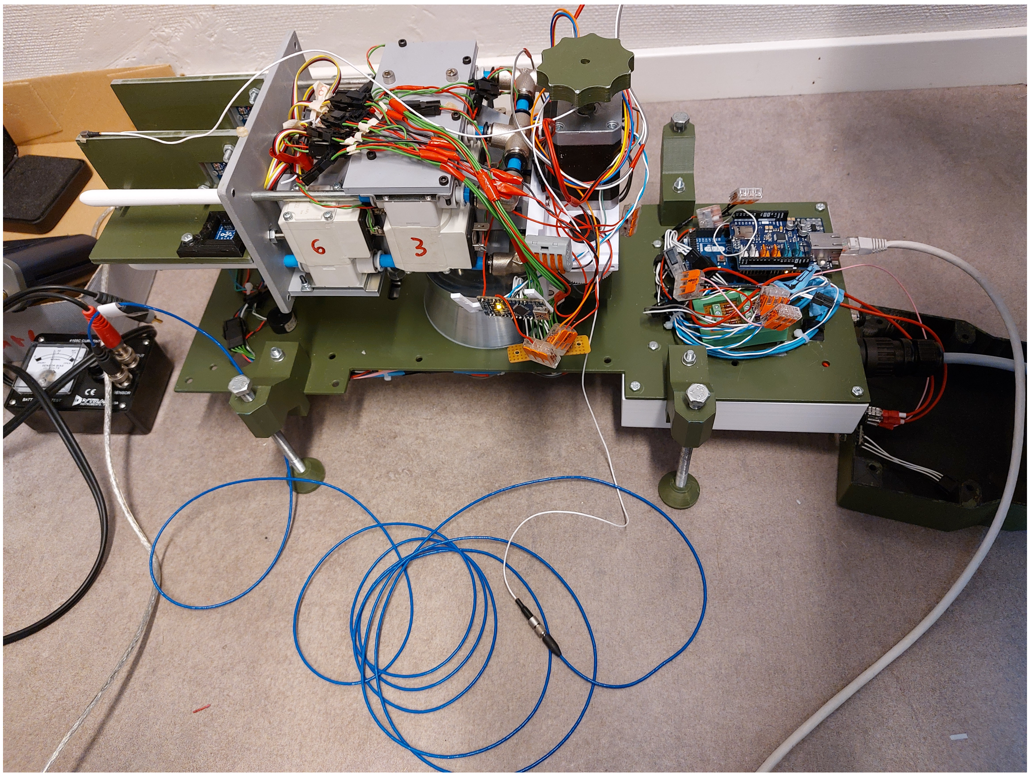


Figure 9.36: Accelerometer on Effector

9.6.3 Analyzed results

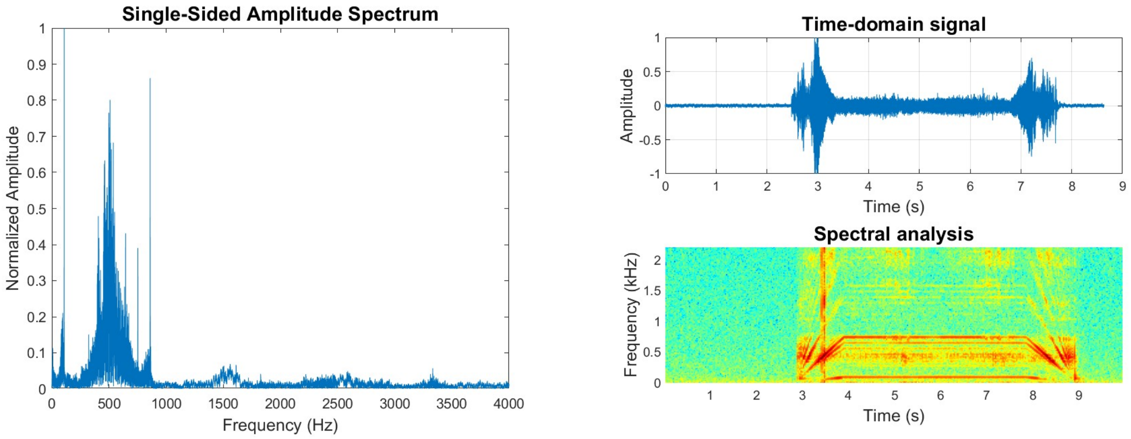
SH

Using the Matlab code in Appendix C, fig. 9.37 through 9.42 were generated from position 1 in fig. 9.35 as the set of measurements were described above. On the left are the single-sided amplitude spectrums of the vibration signal measured. The time-domain signal as it was measured is also plotted for reference on the right side. Below the time-domain signal is the associated spectral analysis.

The spectral analysis is Fourier-based. The signal measured is in the time-domain, and the frequency-domain description is sought after for the The fourth analysis, in section 10.4.1. Therefore a frequency analysis is utilized, i.e. a conversion from the time-domain to the frequency domain. The Fourier coefficients may be computed by the Fourier series analysis shown by (9.6). This is used with the help of Sigmund Gudvangen and his lecture notes [100].

$$X_k = \frac{1}{T}\int_{-\frac{T}{2}}^{\frac{T}{2}} x(t)e^{-j(\frac{2\pi}{T})kt}dt \tag{9.6}$$

The time domain waveform must be periodic, but can extend from $-\infty$ to ∞. Considering the j in $X(j\omega)$, it is usually suppressed when using the Fourier transform. It means that the frequency axis is the $j\omega$-axis in the s-plane, but σ in the complex frequency $s = \sigma + j\omega$ is then zero. The frequency ω goes from $-\infty$ to ∞, creating a two-sided spectrum as both positive and negative frequencies are included. The mathematical treatment is simplified with a two-sided spectrum, but it does not reflect the physical world. As the time-domain signal x(t) is real-valued, its frequency spectrum is Hermitian, that is that the imaginary part is anti-symmetric about $\omega = 0$, and the real part is symmetric about $\omega = 0$. As both parts of a Hermitian spectrum contain the same information, it is enough to only plot the positive frequency spectrum. The spectrums analyzed below are therefore single-sided.



(a) Amplitude Spectrum Pos. 1 0°-30°.

(b) Spectral Analysis Pos. 1 0°-30°.

Figure 9.37: Analyzed data Pos. 1 0°-30°.

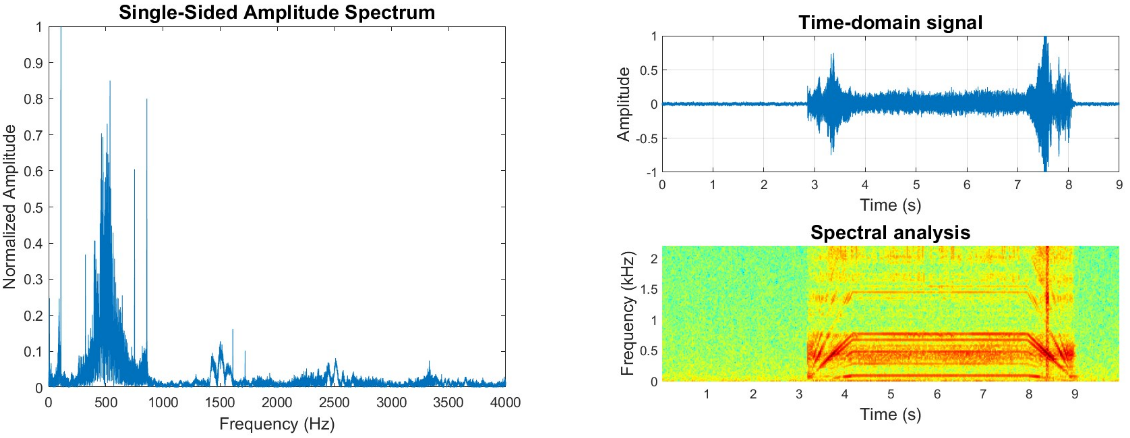
The finished analyzed result when the Effector moves from 0° to 30° is shown in fig. 9.37. Furthermore the single sided amplitude spectrum is shown in fig. 9.37a. The spectral analysis' x-axis represents time in seconds, while the y-axis represents the frequency in kilohertz. The color in a point of the spectrogram represents the amplitude of the frequency in that point. The darker the color, the higher the amplitude of the input signal. In other words, a spectrogram shows amplitude changes for every frequency component in the signal [101].

The resulting plots shows resonating frequencies located at around 500 Hz, 1500 Hz and 2500 Hz. The big spike located at around 50 Hz is generated by the electric motor. The dominating frequency is located at 500 Hz and is the frequency of interest. The other modal frequencies are so weak that they can be neglected as noise. The time-domain signal together with the spectral analysis is seen in fig. 9.37b In both the spectral analysis and time domain signal a big spike appears when the Effector starts and stops the inputted motion. The spectral analysis shows that the resonating frequency located at around 500 Hz, is present during the entire duration of the time period where the Effector is moving, the exception is in the acceleration phase and the deceleration phase of the movement.

Noticing the other frequency spikes present in the recorded vibrational measurement, they can be neglected and viewed as noise. The maximal amplitude signal represents the most significant frequency component in the note, i.e. the dominant frequency. These measurements and analysis is created for the Finite Element Analysis (FEA), the frequencies of interest are the ones that have the highest amplitude.

The amplitudes of the single-sided spectrum are normalized to account for the challenges of obtaining accurate amplitude measurements across different gain settings during the measurement process.

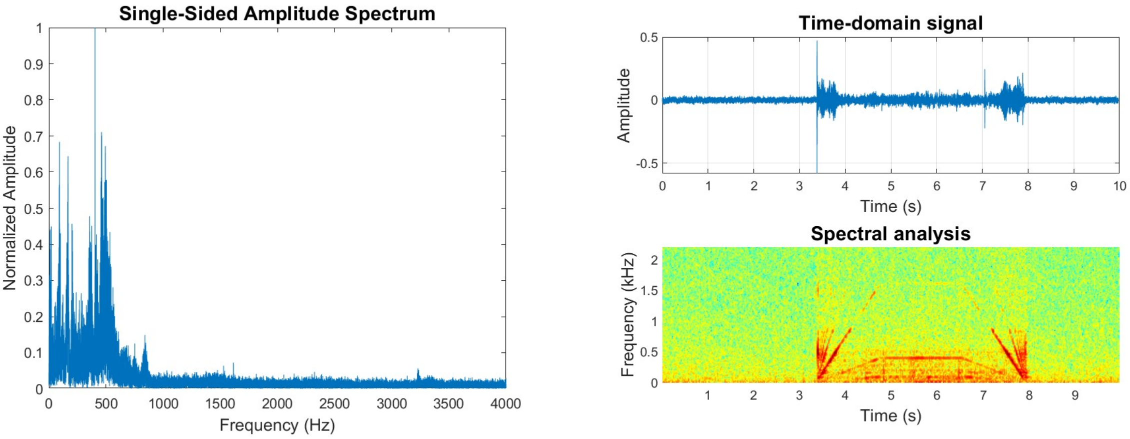
Describing the other measurements would generally result in the same description as the aforementioned. They are included as a point of interest and to get a full overview of the measurements and their significance to the overall System.



(a) Amplitude Spectrum Pos. 1 30°-0°.

(b) Spectral Analysis Pos. 1 30°-0°.

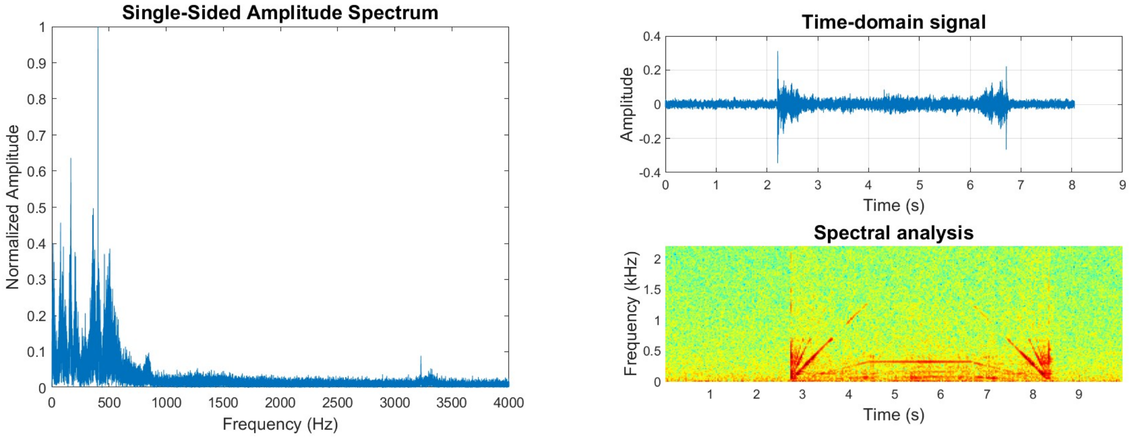
Figure 9.38: Analyzed data Pos. 1 30°-0°.



(a) Amplitude Spectrum Pos. 1 180° to -180°.

(b) Spectral Analysis Pos. 1 180° to -180°.

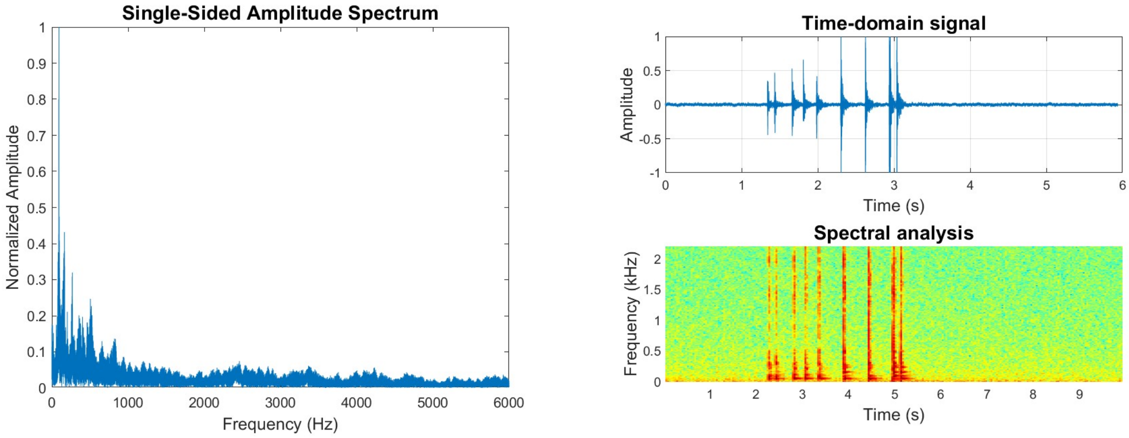
Figure 9.39: Analyzed data Pos. 1 180° to -180°.



(a) Amplitude Spectrum Pos. 1 -180° to 180°.

(b) Spectral Analysis Pos. 1 -180° to 180°.

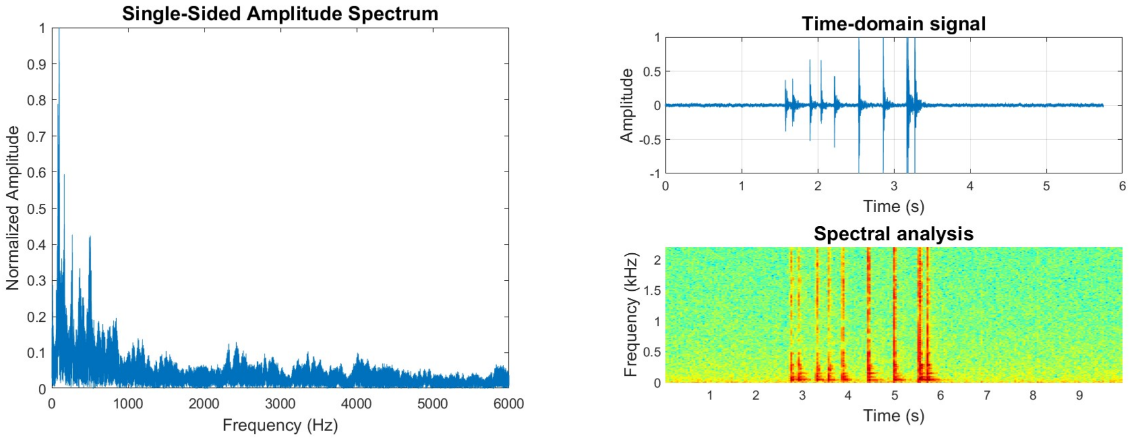
Figure 9.40: Analyzed data Pos. 1 -180° to 180°.



(a) Amplitude Spectrum Pos. 1 fire at 0°.

(b) Spectral Analysis Pos. 1 fire at 0°.

Figure 9.41: Analyzed data Pos. 1 fire at 0°.



(a) Amplitude Spectrum Pos. 1 fire at 30°.

(b) Spectral Analysis Pos. 1 fire at 30°.

Figure 9.42: Analyzed data Pos. 1 fire at 30°.

As a caveat it is interesting to mention the analysis of fig. 9.41 and 9.42. As previously mentioned in the section about the Projectile firing circuit, 9.3.6, the circuit did not function properly. This can be seen in both the time-domain signal and the spectral analysis of these measurements. The sequence firing will open and close the solenoids of each of the six canisters once, in sequence. That would lead to a total of 12 vibrations. There are 9 vibrations recorded, acknowledging the fact that three of the solenoids only open and do not close again. The analysis shows which canisters fail, as the solenoids open for a very short amount of time. In the sequence firing of the third, fourth and fifth solenoid valve it malfunctions. Even tough it was already known that some of the valves were defective the analyzed data shows that an accelerometer can be use full when troubleshooting for errors.

Chapter 10

Mechanical Design

10.1 Introduction

KC

In this chapter, the Effectors mechanical state was investigated and evaluated. Issues discovered were looked at through the looking glass and new design solutions was proposed, discussed and developed. This chapters main goal was to increase the Effectors accuracy, facilitating for the approval of the stakeholder requirement FR-04. This requirement states that the Systems Effector shall hit marked targets and is divided into three system requirements. These systems requirements, SR-04-01, SR-04-02 and SR-04-03 describes the dimensions of the targets, the distance between target and Effector and the required level of accuracy in both azimuth and elevation.

10.2 Mechanical state and issues

KC

After receiving the physical components from "Project Mini Launcher" [1], including the Effector and the projectiles, a check of the initial state was conducted. During this check, several issues was discovered. These discoveries are further discussed in this chapter:

10.2.1 Azimuth

The Effectors azimuth was not satisfactory precise. The motor was struggling to start the rotational movement, most likely caused by the static friction between surfaces in the rotational base, also known as the stick-slip phenomenon. There was also a problem when tightening the belt driving the azimuth. The forces working on the gear connected to the rotation base from the belt caused the whole base to shift towards the tension wheel. This resulted in the rotation base not keeping a vertical position, as represented by the red dotted line in fig. 10.1. The force from the belt is represented by the orange arrow and the original preferred position of the rotation base is represented by the blue line.

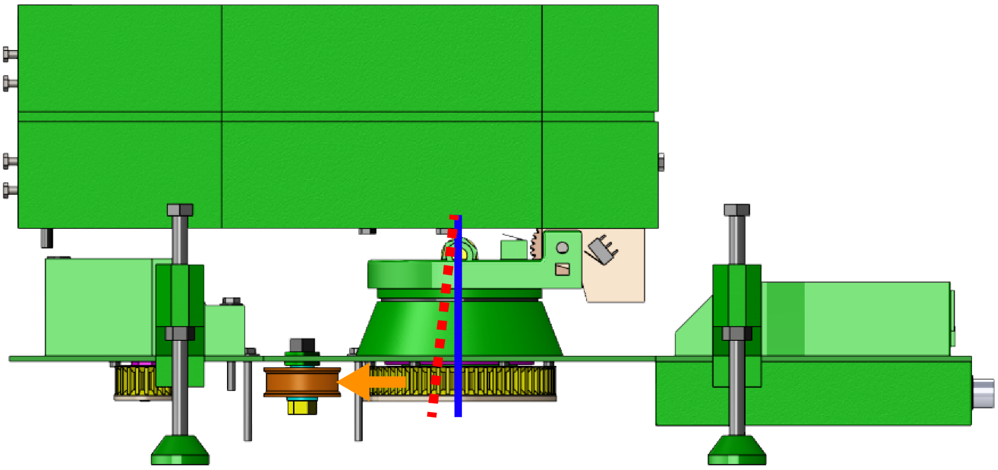


Figure 10.1: Bias in Effector rotation base.

This positioning was an obstacle for the rotation, since the originally free surfaces within the rotation base would touch, causing more friction between the parts. The reason for this shifting was that the rotation base lacked the structural strength needed to withstand the radial force from the belt. This occurred because the tolerances between the 3D-printed parts were not good enough and that the material used, PLA, is elastic.

Previously, there has been attempts, conducted by the previous bachelor group, to reduce friction by using Vaseline lubricants. This has not been very successful. After dismantling the rotation base, the conclusion was that lubricants did not work due to no space for it to fill.

10.2.2 Belt drive

The Effectors azimuth is run by a belt drive. As mentioned earlier, the rotational movement was not acceptably accurate. Several components from the belt drive can be contributing factors for this inaccuracy. Fig. 10.2 illustrates the placing of the belt. The center gear is connected to the rotation base, the bottom left gear is directly connected to the encoder and the bottom right is directly connected to the motor. The orange gear was a tension wheel.

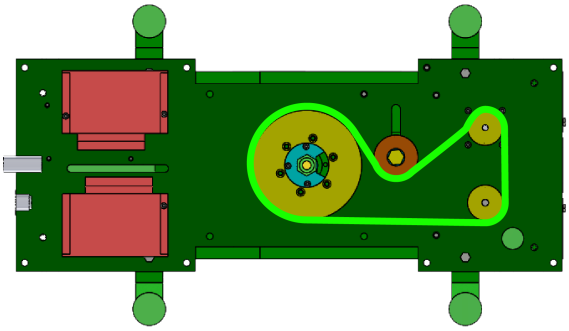


Figure 10.2: Illustration of belt drive.

When examining the belt drive, factors contributing to inaccuracy was found:

1. All gears used in the belt drive was made from 3D-print. This could be an issue due to several reasons. Since PLA is an elastic material, the gears could deform if exposed to loads, especially over time. This means that round gears can turn oval if presented to the radial forces from the belt. Also, 3D-print in itself is not very accurate. If components are printed with different settings, the tolerances will be different on each part. This results in the gear ratio possibly not being as first calculated.

2. The encoder is placed in direct contact with the belt, fully exposed to the radial forces from the belt when tensioned. According to the encoders data sheet, the encoder axle can only hold 4.9 N in radial direction [24]. This is less than the load used to tension the belt, meaning that the tension of the belt can overbear the encoder axle. This resulted in the encoder giving in to the load from the tension, shifting towards the tension wheel. When the encoder was not in its supposed vertical position, it did not function properly.

3. When changing the rotation direction, the belt must readjust causing backlash.

4. The tension wheel was, as all other gears in the belt drive, made from 3D-printed PLA. This was placed on a 3D-printed bushing on a bolt. The tolerances in these parts were not good enough, causing the tension wheel to shift in different directions when exposed to loads.

10.2.3 Elevation

The Effectors elevation was more accurate than the azimuth, but had other issues. These problems revolved around whether or not the Effector actually elevated. The elevation mechanism is shown in fig. 10.3 and consists of a static axle kept in place in the elevation base with setscrews. On this axle, there is a worm gear (gear 1) connected to a worm screw (gear 2) driven by the motor. When the motor drives the worm screw, the contact between gears will cause the Effector to elevate. One problem discovered here, was that if the setscrews keeping the axle in place was not tightened enough, the axle would rotate alongside the motor-driven worm screw and the Effector would not elevate. This was challenging since the elevation base was made from 3D-printed PLA, making it difficult to tighten the setscrews properly. Another issue hindering the smoothness of the elevation was that when elevating, the gear stationed on the axle would shift a little to the left. There were not made extra room in the structure for this to happen, causing the gear to touch, and then also cut into the surface of the structure.

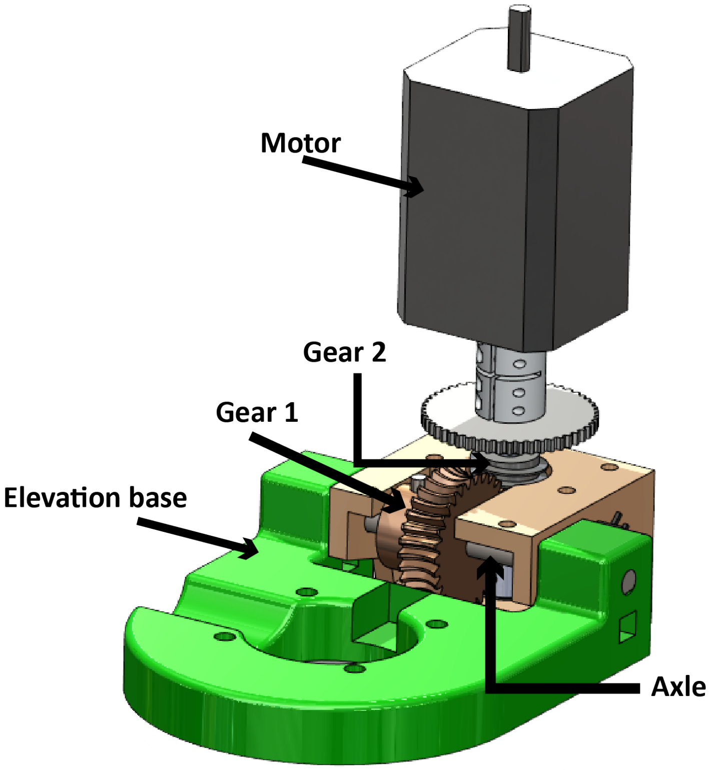


Figure 10.3: Elevation components.

10.2.4 Projectiles

When testing the firing-mechanism the projectiles were used. During this test, it was discovered that the quality of these projectiles were poor. The projectiles shape can be seen in fig. 10.4.

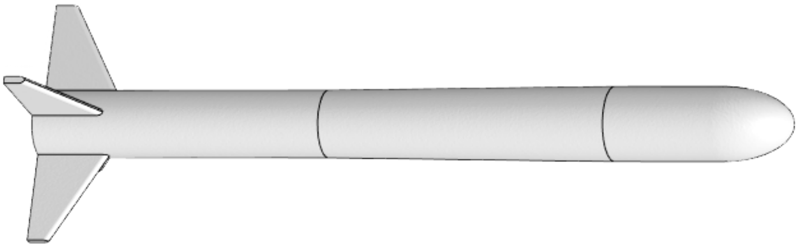


Figure 10.4: Old projectile

There are two problems with these projectiles: both the wings and the body fractures. The wings fracture because they are thin and fragile, not designed to withstand the impact of hitting a target. Here it was obvious that it was the material fracturing. The body on the other hand, was designed to withstand more impact force. When looking at the fracture surface when a projectile was split in two, it looked as though the material was not the problem here. Since this was a 3D-printed part that has been printed standing up, it was easy to see the different layers building the model. In the body's case, the adhesion between these layers does not seem to have been strong enough. This is a fault that has to be accounted for when using 3D-print. Even though the material used may be strong enough for its purpose, the adhesion may not be, causing the part to fracture.

These problems will each be dealt with separately and the process will be described in the following chapter.

10.3 Redesign

KC

After mapping the Effectors mechanical challenges, the process of redesign and design initiated. This included redesigning old components to function properly through reverse engineering, edit material choices and design new elements. When redesigning, the focus has been set at solving previously mentioned issues and on the approval of relevant requirements. This includes the stakeholder requirement FR-04 and its related system requirements SR-04-01, SR-04-02 and SR-04-03, as well as the system requirement SR-11.

10.3.1 Rotation base

KC

As described in chapter 10.2, the Effectors azimuth, or rotation, was not precise enough to meet our requirements in the state at which it was given to us. In order to fix this, the problems discovered regarding the rotation base must be resolved.

When looking at the rotation base, the main mechanical problems are the friction keeping the azimuth from moving smoothly and the lack of structural strength needed to withstand the radial force from the belt. To dissolve these issues, several concepts were thought of:

1. Keeping the rotation base as it is but adding space for lubricants.
2. Adding a bearing to the 3D-printed structure.
3. Changing the material from PLA to metal, preferably aluminum.
4. Adding a bearing and changing material from PLA to metal, preferably aluminum.

All these options are based on the existing design. This is because this design fits the rest of the Effector, and the parts are physically available. This gave the opportunity to proceed with reverse engineering. When using reversed engineering, elements from the existing design is kept but optimized to fit the new requirements. This means that the original design is used, only modified to fit the new needs.

The first alternative would solve some of the problem regarding friction, but nothing is done to improve the structural strength. This means that the issue regarding the lack of ability to withstand radial forces was not solved.

The second alternative would solve many of the friction-related issues, and the structural strength improves some. The problem with this solution, is that the tolerances needed for the bearing to keep in place and function optimally are very difficult to impose in the 3D-print. Since this should be dismantlable, using Loctite or other kinds of glue on the bearing is not an option.

The third alternative would, depending on how it is executed, solve some friction-related issues and add structural strength. Even tough this seemed promising, it would not solve the issues satisfactory.

The fourth alternative would solve the friction-related issues and add the needed structural strength. This is because the new material is less elastic than the old one, making it more rigid. Also, the bearing, depending on type, would carry a lot of the load.

From these four alternatives, the latter fit the goal the best. With access to the USN workshop there would be no problem manufacturing the required parts.

Design

When designing the new rotation base, the focus is set on multiple factors:

- Simplicity in design: How simple the design is.
- Simplicity in manufacturing: How simple it is to manufacture and assemble parts.
- Reverse engineering: How much of the old design can be reused.

In fig. 10.5, the old rotation base is illustrated. It consists of:

1. Flange that connects the bottom structure of the Effector to the top structure.
2. Bushings to ease the rotation.
3. Fasteners.
4. Base cover – the part visible when the Effector is fully mounted.
5. Base plate.
6. New bushing to add stability.
7. Rotation gear.

All parts, except fasteners and base plate are 3D-printed.

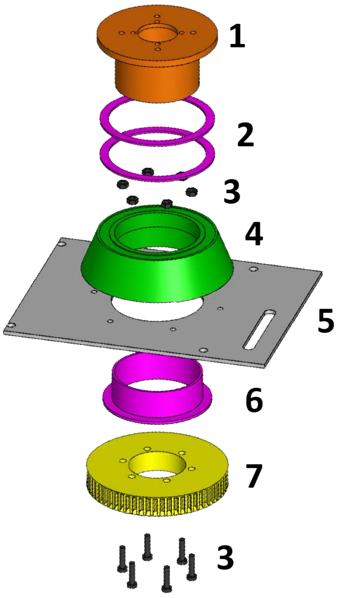


Figure 10.5: Original rotation base

Initially, the parts that has to be reused are the parts with external interfaces. This includes the orange bushing and the green base, since these are attached to external parts like the rotation gear and the base plate. There are several reasons to keep these parts:

- The flange connects the top – and bottom structure of the Effector, as well as the belt drive. Replacing this part will mean that several other parts will require redesigning.
- The base cover is designed to fit the features on the base plate, meaning that editing the base covers interface automatically creates a need to redesign the base plate. Also, this part is designed to resemble the real NASAMS, and this resemblance should be kept.

After landing on which parts to reuse and what concept to persuade, the design process was initiated. Since the decision of implementing a bearing, the need to research different types of bearings occurred.

Bearing selection

Bearings When choosing a bearing for the Effector, there are some important criteria to consider:

- Price
- Low friction coefficient
- Available in standard sizes
- Low maintenance/long grease-life

In this project, it is not important that the chosen bearing can withstand significant temperature changes. This is because the Effector will be kept at room temperature most of the time. It may be introduced to changes in temperature when transported, but this is not often enough for it to affect the bearing. It also may be stored in colder temperatures, but the chances of it being colder than 10 degrees are low.

Since the chosen bearing will be attached to the Effector helping with the azimuth, we do not need a bearing that can withstand high-speed operations. The Effector is moving at a certain speed, and it is not constantly moving. Even though, the bearing needs to have a low friction coefficient. This is because one of the current problems is that the motor are struggling to start the azimuth because of the friction between the 3D-printed surfaces.

When choosing bearings for a mechanical structure, it is important to find a bearing that will hold the loads they are exposed to. In our case, most bearings of the needed size will be fit for the purpose. This is because the Effector is not a big and heavy machinery. The top structure only has a mass of approx. 5 kilograms. Also, as mentioned before, the bearing chosen will not be affected by high-speed operations. This means that the total load on the bearing will not be very high.

Rolling element bearings vs. slide bearings When talking about bearings, there are two main categories: rolling element bearings and slide bearings. Both types have pros and cons. According to “Maskindeler 1" [102], the friction at start is a lot less using rolling element bearing, since the friction coefficient for rolling element bearing is $\mu = 0.02$ while the same coefficient for slide bearing is $\mu = 0.12 - 0.24$. This means that if the operations include frequent start and stop, the rolling element bearing is preferred. The rolling element bearing also has a low friction during operations, while this only applies to the slide bearing with fluid friction.

The slide bearing tolerates vibrations and impacts better than the rolling element bearing, but the rolling element bearing can withstand a greater force per area [102].

When it comes to maintenance, there are also a few differences. The rolling element bearing is easier to replace if broken than the slide bearing. If greased with fat, it also does not need as much supervision or maintenance. This means that even though the rolling element bearing is a little more expensive to buy than the slide bearing, the minimal maintenance cost makes it cheaper in the long run.

This information tells us that in this project, the rolling element bearing will fit the best. This is because it requires minimum maintenance, is cheap to use and has a generally lower friction coefficient [102].

Rolling element bearings After researching the different types of rolling element bearings, we have found two different that seems to fit the project:

- Single row ball bearing
- Roller bearing

The reason we chose to look further into these two types, is that both are common bearings in affordable price ranges that show high functions. Comparing the two types, we found that: The difference between the two bearings, is the shape of the inner elements. The ball bearing consists of spherical objects and the roller bearing consists of cylindrical objects. The different shape of these objects is giving the bearings different area of contact, which effects the friction. The ball bearing is the best suited for operations where the ease of operation is important. This is because it experiences less friction than the roller bearing, making the rotation of the spherical objects run smooth. Another positive thing about the ball bearing is that it can handle both radial and axial loads, unlike the roller bearing. This being said, the roller bearing is designed to bear larger loads than the ball bearing. The roller bearing comes in different variants, and the one we were most interested in were the needle bearing. This type is a roller bearing where the cylindrical objects have a much smaller diameter, resembling needles. Alongside the pros and cons from the regular roller bearing, the needle bearing has the advantage of having a smaller thickness, making it suitable for structures where space is critical. It also has the disadvantage of more friction since the needles rotates with a greater velocity. Both ball bearings and different variants of the roller bearing is commercially available for a relatively low price [103].

Other types, such as the tapered roller bearing and the thrust ball bearing, was also researched. The tapered roller bearing is more complex, but it can withstand greater axial and radial load than the ball bearing and roller bearing. Since the chosen bearing will not be exposed to extreme loads, this is not an necessary property. The complexity of this bearing entails that it is not researched further. The thrust ball bearing can only withstand axial loads, and since there is no guarantee that there will be no such loads working on the system, this is not a good fit [102].

Conclusion bearing selection After learning all this, the conclusion is that for this project, the ball bearing fits best. This is because we prioritize low friction over high loading capacity. We need the Effector to rotate smoothly even though it's not rotating at high speed, and it will not be exposed to heavy loads.

Corrosion

In the chosen concept, the rotation base is produced in aluminum and the bearing house is made of bearing steel. Using active metals and combining two metals with different electrochemical potential raises the question if corrosion will be a problem. Corrosion is a form of electrochemical or chemical decomposition of a material. It can occur in different forms, such as oxidation, galvanic corrosion, pitting corrosion, intergranular corrosion and more. The main reason for corrosion to occur is that metals found in nature are not in their pure form. After being extracted from the ore, the material is purified. A purified metal is not in a natural state and will react to return to this. Defects and inhomogeneities in the materials microstructure can also contribute to development of corrosion [104].

Corrosion is typically accelerated if the material is exposed to environments including elevated humidity, low pH-value or increased temperature [104]. This will not be an issue for the rotation base since the Effector will be stored inside at room temperature.

Galvanic corrosion occurs when two materials with different potential are placed together in a humid environment, also called an electrolyte. An electrolyte is a liquid containing minerals with the ability to conduct current. As mentioned, the rotation base is produced in aluminum and the bearing housing is made from steel. These materials have different potential and are placed on different levels in the galvanic table. In the galvanic table, materials are sorted by their potential: noble metals are less reactive and corrodes at a lower rate than non-noble metals. If two or more metals with different potential are placed together, the metal with the highest potential will corrode first. From the galvanic table, it is found that aluminum has a higher potential than steel and will therefore corrode first if placed in an electrolyte. This will not be a problem, since the Effector will not be operated in an electrolytic environment [104].

In conclusion, corrosion is not an issue for the rotation base. This is because the Effector will be stored and operated at controlled environments: dry and indoor.

Bearing dimensions and placement

After this research, I-Senteret was contacted to provide the needed bearing. They offered different sizing suggestions:

1. Ball bearing 60x85x13
2. Ball bearing 60x78x10
3. Ball bearing 60x95x18
4. Axial bearing 60x85x17

The first alternative would fit in the original geometry with only a few modifications. The disadvantage with this is that the price is a little steep: 1000 NOK.

The second alternative would also fit in the existing geometry, but since it is smaller than alternative 1, it will absorb less forces than the bigger one. This alternative also has the same disadvantage as the first alternative: the price is the same.

The third alternative would not fit in the original geometry without major alterations. This has a nicer price, but the extra work regarding more redesign does not make it worth it.

The fourth alternative can absorb more axial forces than the other alternatives, but it can not absorb the radial forces. As described in the bearing selection, the axial bearing is not a good fit for this system. Apart from this, the bearing would fit in the original geometry and the price is at the same level as alternative three.

From these alternatives, alternative 1 is the best fit. This conclusion is made based on the following factors:

- It is a ball bearing, meaning it is able to withstand both radial and axial forces.
- It is simple to fit in the existing geometry without excessive alterations.
- It is the largest ball bearing not in need of excessive alterations to the existing parts. Bigger ball bearing is better, since this is equivalent to the balls inside the bearing being bigger. This causes an increase in contact area, giving the friction a larger area to work on. This results in less friction per area and more evenly distributed forces.

After choosing the bearing, it is important to agree on a placing. Sketches of the two best solutions were made. Fig. 10.6a and 10.6b shows the chosen bearing placed either on top of the base cover, or at the bottom. If placed on bottom, the only thing necessary to keep it in place is a thin 3D-printed support. If placed on top, a retaining ring will be needed to keep it in place.

(a) Bearing placed on top.

(b) Bearing placed on bottom.

Figure 10.6: Different placings of bearing in rotation base.

Since the flange is directly connected to the rotation gear driven by the belt, placing the bearing at the bottom will be the best option. This is because at this position it will be closest to the radial forces from the belt. By minimizing the distance from the bearing to the active force, the bearing will relieve the structure from most of the forces, increasing the structural strength.

Bearing life expectancy After choosing the ball bearing, the desire to investigate the bearings life expectancy occurred. The bearing chosen is an NTN 6912LLU/5K ball bearing with ID 60 mm, OD 85 mm and width 13 mm, as mentioned earlier. To calculate the life expectancy, some information about the bearing is needed:

- $C_{10} = 18,2$ kN, the basic dynamic load rating [105].
- $C_0 = 14,3$ kN, the basic static load rating [105].
- $L_R = 1 \cdot 10^6$ rev, SKF is rating their bearings for 1 million revolutions, the calculations will be based on this since there is no easy access information explaining what NTN uses [106]
- $F_a = 200$ N, the axial load the bearing is exposed to.
- $F_r = 200$ N, the radial load the bearing is exposed to.

The last two items on the list represents the axial and radial loads on the bearing from the Effector. The axial force influencing the bearing comes from the weight of the top structure. Since the top structure has a mass of 4-5 kg, the maximum axial force will be:

$$F_y = m \cdot g = 5\,\mathrm{kg} \cdot 9,81\,\mathrm{m/s^2} = 49,05\,\mathrm{N}. \tag{10.1}$$

There are no other axial forces influencing the bearing, meaning that $F_y = F_a$. Choosing this force to be much greater than it is in reality is done to ensure that the life expectancy of the bearing is well within what is needed. The safety factor used is approx. $n = 4$. The radial forces consist of the force from the belt when satisfactory tensioned. The force from the belt is unknown at this point of the process. Because of this, it is decided to use the same value for the radial forces as for the axial forces, including a safety factor:

$$F_r = F_a = 49.05\,\mathrm{N} \cdot 4 \approx 200\,\mathrm{N}. \tag{10.2}$$

The resultant of these loads can be expressed as:

$$F_e = X_i V F_r + Y_i F_a. \tag{10.3}$$

It is known that the rotation factor $V = 1$. This is because in this system, it is the inner ring of the bearing that is rotating. The values of X_i and Y_i has to be found from a table and are dependent on the bearings construction and geometry. Using table 11-1 in Shigleys [106]:

$$\frac{F_a}{C_0} = \frac{200\,\mathrm{N}}{14300\,\mathrm{N}} = 0.014. \tag{10.4}$$

From the same table, this value gives $e = 0.19$. e is an abscissa defined by the intersection of $\frac{F_e}{VF_r}$ and $\frac{F_a}{VF_r}$ when plotted. From this, the value of X_i and Y_i can be determined. Since:

$$\frac{F_a}{VF_r} = \frac{200\,\mathrm{N}}{1 \cdot 200\,\mathrm{N}} = 1 \rightarrow 1 > e \rightarrow i = 2, \tag{10.5}$$

where $i = 2$ means that $X_i = X_2$ and $Y_i = Y_2$. The value of i can be 1 or 2, depending on whether $\frac{F_a}{VF_R}$ is less than or greater than the value of e. Going back to table 11-1 in Shigleys [106], we see that $X_2 = 0.56$ and $Y_2 = 2.3$. From this, F_e is found using (10.3):

$$F_e = X_i V F_r + Y_i F_a = 0.56 \cdot 1 \cdot 200\,\mathrm{N} + 2.3 \cdot 200\,\mathrm{N} = 572\,\mathrm{N} \tag{10.6}$$

It is known that [106]:

$$C_{10} = F_e \cdot \left(\frac{L_{10}}{L_R}\right)^{\frac{1}{a}} \tag{10.7}$$

where L_{10} is the bearings life in revolutions under the desired loads and L_R is the theoretical bearing life in revolutions. In this case, $a = 3$ since the bearing is a ball bearing. Since the interesting element in this equation is the L_{10},(10.7) needs to be solved for L_{10}:

$$L_{10} = L_R \cdot \frac{C_{10}}{F_e}^{a} \rightarrow L_{10} = 1 \cdot 10^6\,\mathrm{rev} \cdot \frac{18200\,\mathrm{N}}{572\,\mathrm{N}}^{3} = 3.22 \cdot 10^6\,\mathrm{rev}. \tag{10.8}$$

To transform this value into hours:

$$L_{10} = 60 \cdot L_{10} \cdot n_D \rightarrow L_{10} = \frac{L_{10}}{60 \cdot n_D}, \tag{10.9}$$

where n_D is the rotational speed of the Effector, given in rev/min. This has been timed to 12 rev/min.

Using 10.9, and the result is:

$$L_{10} = \frac{3.22 \cdot 10^6\ \mathrm{rev}}{60 \cdot 12\ \mathrm{rev/min}} = 44739753\,\mathrm{h} \approx 44\ \text{million hours} \tag{10.10}$$

The life expectancy of this ball bearing under these conditions is approx. 44 million hours. This is much longer than needed for this system. In table 11-4 in Shigleys [106] we can see that the recommended bearing-life for instruments and apparatus of infrequent use is approx. 500 hours. This means that the bearing should function for a long time and that other failing modes will impact the system before reaching the end of the bearing's life [106].

Redesign

Fig. 10.7 shows the flange before and after redesign. The implemented changes are:

- Extension of the center hole on top of flange. This was done to give air and cables entering through this hole more room.
- Adding a rim fit to rest on the inner ring of the ball bearing.
- Changing the diameter of the bottom part to fit the bearing using press fitting.
- Moving clearance holes used to attach the air.

(a) Old flenge-design.

(b) New flange-design

Figure 10.7: Old and new flange-design compared.

Fig. 10.8 shows the base cover before and after redesign. The implemented changes are:

- Bushing-space removed.
- Edited inside rim to fit for rest on outer bearing ring.
- Inside diameter changed to fit bearing with press fitting.

(a) Old base cover.

(b) New base cover.

Figure 10.8: Old and new base cover design compared.

Conclusion

Editing the material for these parts from PLA to aluminum will increase the mass. This issue has been neglected, since the only part whose weight will affect the rotation is the flange. The original flange has a mass of 92.8 grams, while the new part will have an estimated mass of 234.3 grams. Even though this constitutes an increase of approx. 250%, the reduction of friction will still make the rotation movement easier for the motor to perform.

In conclusion, the new rotation base design is shown in an exploded view in fig. 10.9, showing the needed parts: the flange, the base cover, bearing and support.

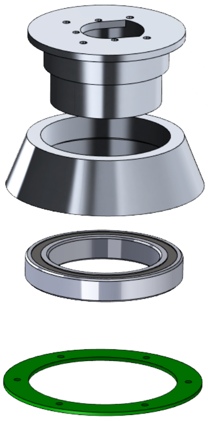


Figure 10.9: Exploded view showing the new rotation-base design.

Production

MH | KC

To machine the parts designed by Karine, the options was to either turn them using a lathe or cut them out with a milling machine. Seeing as the parts are round, it was best suited to turn the parts in a lathe which was done using the schools Pinacho lathe. To choose the raw material it is best to get as close as possible to the biggest dimension on the part, this is to minimize waste and machine time. Cutting away excess material takes both time and creates a lot of chips. Another thing to keep in mind is that the raw material needs to be long enough to be able to fix the part into the chuck on the lathe, as well as having room to machine the desired part. This made it so that the raw material for the rotation base was a round stock of aluminum with a diameter of 110 mm and length of 150 mm. The rotation cylinder was made from a round stock with a diameter of 100 mm and length of approx. 170 mm, the material was donated to the project from KDA Aerostructures (DMA).

A lathe is a machine that can perform a number of different machining operations, such as turning, facing, drilling and threading. The most basic operation is turning, meaning that the tool is static while the work piece rotates. It is an excellent machine for production of circular parts. The lathe is also commonly used to produce parts with complex design, close tolerances, good surface finish and external and internal threads. Since this design has close tolerances and desires a good surface finish, this machine is a good fit [107].

When performing a turning operation, there are different cutting-data to consider. This can be the work pieces' rotational speed, the depth of cut or the feed. These data depend on the material of work piece and cutting tool and the desired tolerances and surface finish. For instance, too rapid feed can result in a rough surface. As recommendation for general-purpose starting conditions when the work piece material is an aluminum alloy and the cutting tool is either uncoated or TiN-coated carbide, the feed should be approx. 0.45 mm/rev. The corresponding cutting speed should be between 490 and 550 m/min [107].

The material chosen for these parts, aluminum, has a good machinability in general. One problem for the softer aluminums is that they sometimes form a Built-Up Edge (BUE) chip. This is a type of chip that "consists of layers of material from the work piece that gradually are deposited on to the tool chip" [107]. The BUE chip becomes unstable and breaks off when growing big enough, resulting in a rough surface. Because of this, high cutting speed, rake angles and relief angles are recommended. When machining aluminum it can be hard to achieve fine tolerances because of the materials relatively low E-module and high thermal expansion coefficient [107].

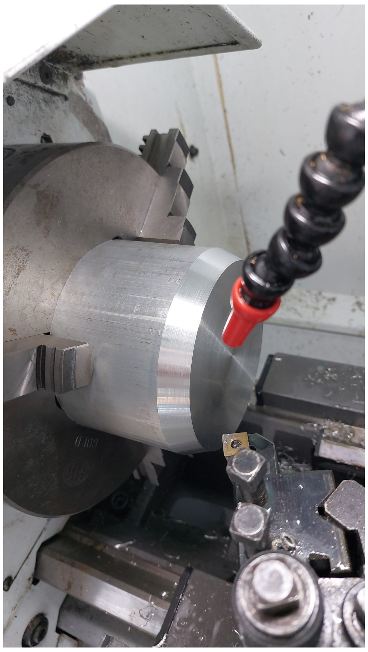


Figure 10.10: Begining of angle cut.

By starting with the rotation base a little room for error was given when getting acquainted with lathe, seeing as no group member has used the machine before. Facing is done as the first cut, and then the angled part of the rotation base was cut, the beginning of the angle cut is seen in fig. 10.10. The finished angle size is shown in fig. 10.11. A general rule of thumb when turning is to remove as much of the material as possible before doing the cutting the bits of the part with the tightest tolerances. This is to ensure that there is enough material when cutting the part to the final tolerances. It is also important to make as many cuts as possible from one side of the part, because when rotating the part and fixing it in the chuck there is room for error. This is since it is difficult to fix the part with the same zero point as the other side. To minimize the throw a dial gauge can be used to measure the throw on the part when fixing it in the chuck. Another important factor is that when removing material friction is produced, which causes heat, and this can make the part warp. When doing so the last cut that was taken on the rotation base was on the inside diameter on the bottom. This had a measurement of ø 85 with a tolerance of + 0 to + 0.03.

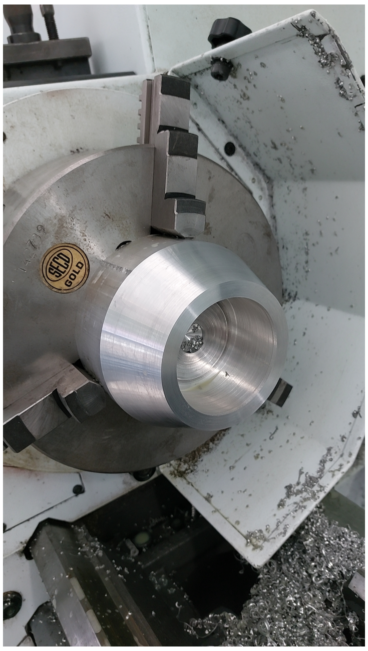


Figure 10.11: Rotation base angle cut

Cutting fluid was used during the turning, this is to minimize the heat generated when machining, as well as reducing the wear on the tools used. For the parts turned for the Mini Guardian project, mainly three different tools were used for turning, along with various sizes of drill bits. The drill bits are used to start the hole for the inside diameter, firstly a small drill bit is used to create a starting center hole. And then the size of the drill bit is increased until the tool for turning inside diameters fits inside the hole and can take the rest of the material of. When creating the hole, it is important to drill deep enough into the part. When the inside diameter of the hole and angle surface was finished and turned to the right measurements, the part was flipped in the lathe. A dial gauge was used to measure the throw when the part was fastened in the chuck, this is shown in fig. 10.12, the dial gauge is capable of measuring 0.01mm. This is in order to minimize the throw when flipping the part, making the new cuts correct in reference to the other side. When fastening the part it is lightly fastened and then the chuck is spun around by hand in order to see the throw. If the measurement is big a rubber hammer is used to tap the part in the correct direction, then it is simply a matter of trial and error to get the part as close as possible to 0mm throw.

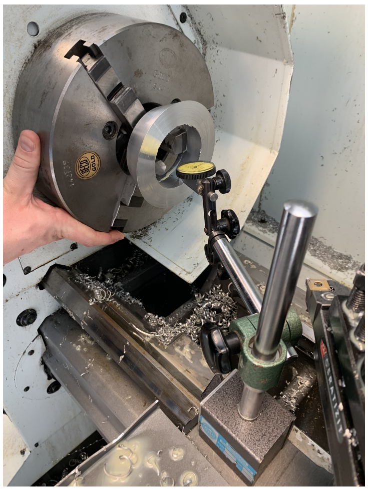


Figure 10.12: Measuring throw.

To get good surface finishes the lathe can be put into automatic feeding by pulling a lever, this makes it so that the given axis moves in the desired direction with a fixed speed according to the rpm speed of the spindle. Different feeds are used for different speeds and diameters of the parts turned. By using aluminum this range is quite big as aluminum is forgiving when machining. A general rule of thumb when removing material is to take away up to 2 mm per cut for coarse cuts, and for fine cuts the depth is 0.2 mm or less. To measure the if the correct tolerances are accomplished a internal micrometer was used, this is shown in fig. 10.13.

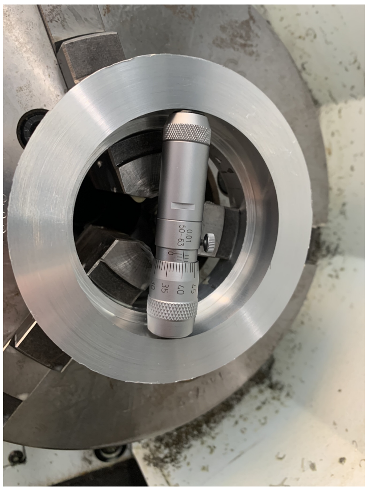


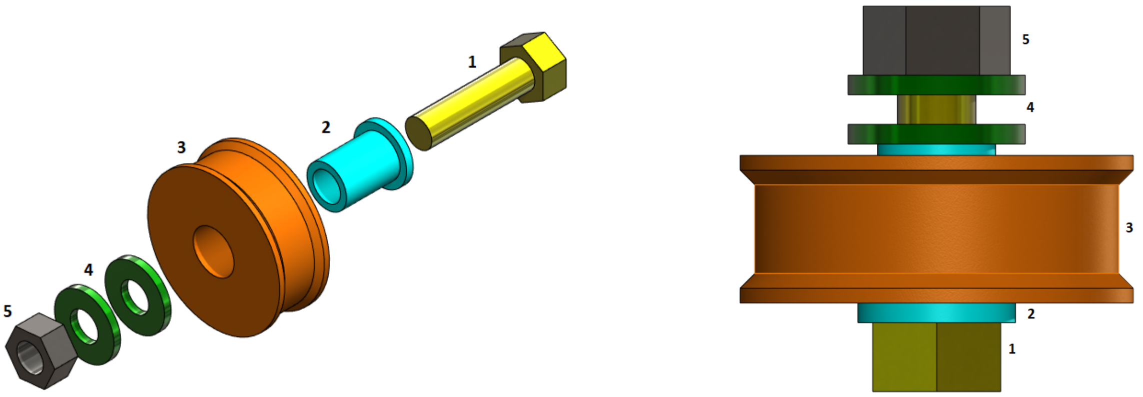
Figure 10.13: Measuring bearing clearance

To drill and tap the holes on the bottom and top side of the parts a drill press was used, the holes were marked with a printed paper template. The holes were tapped by hand, and the drill press was used as a starting guide to make sure the tap went in to the hole straight.

10.3.2 Tension Wheel

KC

After doing a physical check-up of the Effector, there were found several things that needs to be rectified for the Effector to complete the FR-04 stakeholder requirement. One of these things are the tension wheel used to tighten the belt driving the azimuth. When examined, it was found that this wheel was not steady and would move in more than one direction when exposed to tension. This is because there is some slack between the wheel itself, the bushing, and the bolt. Fig. 10.14a is showing the original tension wheel with its functioning components in an exploded view. Here, 1 is the bolt, 2 is the bushing, 3 is the tension wheel itself, 4 is two washers and 5 is a nut. Fig. 10.14b shows the tension wheel and its components assembled.



(a) Exploded view of original tension wheel

(b) Assembly of original tension wheel

Figure 10.14: Original tension wheel illustrated.

The problem here, is that the fitting between the bolt, the bushing and the tension wheel is not tight enough, causing the tension wheel to move when exposed to tension. This is unfortunate for the belt drive. To fix this problem, it is required to redesign this part.

When redesigning this part, there are two main goals: to reduce the friction between the dynamic wheel and the static bolt and reduce the slack between the components. To achieve this, the material will be changed from 3D-printed plastic (PLA) to aluminum and a ball bearing will be fitted between the bushing and the tension wheel.

Since this is mainly reversed engineering, reusing parts is preferred. From the old assembly the bolt, the washers, the nut, and the bushing can be reused. This means that the only thing in need of redesign is the tension wheel.

The bushing has an outside diameter of 12 millimeters, meaning that the ball bearing needed must have an inner diameter of the same dimension. The chosen ball bearing has ID 12 mm, OD 21 mm and width 5 mm. When redesigning the tension wheel, it is important for the stability to place the bearing in the middle. Since the original tension wheel is 15 mm in height, this bearing must be placed on a supporting edge 5 mm from the rim. Fig. 10.15 shows an exploded view describing how the parts are assembled.

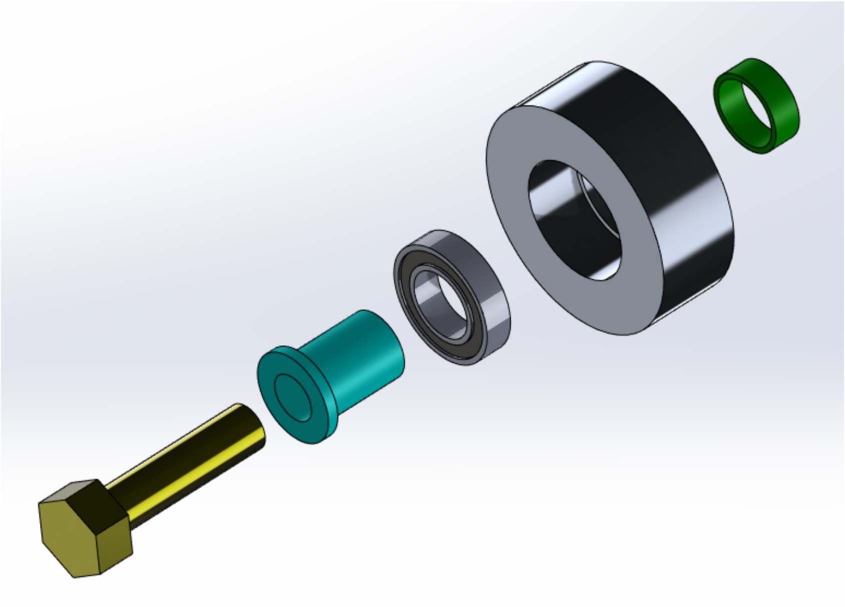


Figure 10.15: Exploded view of redesigned tension wheel.

When redesigning the tension wheel, it is preferred to use as much as the old model as possible. Because of this, the old model was used as a base, only adding or removing some features. The existing dimensions are kept the same, removing the belt tracks along the outer diameter and adding room for the bearing. 2D drawings of this part is found among the belonging attachments. Since the ball bearing has to be fitted tightly, the only critical dimension on this part is the inner diameter where the bearing is supposed to be placed. This is supposed to be fitted with a press fit, which is why this dimension is imposed a strict H7 tolerance.

Production:

The only part that must be produced to upgrade the tension wheel assembly, is the tension wheel. This part was manufactured at USNs workshop using their lathe. The material used is a 6082 aluminum alloy solid rod with a diameter of 40 mm. Since the outer diameter of the part is also 40 mm, this did not need manufacturing. The lathe was used to first bore the center hole with different size bores: starting at 7 mm and moving our way up to 19 mm using four bores. Then the lathe tool was changed and the room for the bearing was made. Despite carefulness, the hole got a little bigger than the H7 tolerance allows. Because of this, an investigation of methods to fit the bearing without making a new part was preceded.

When finding methods to fit the bearing, it was decided to investigate three possibilities: setscrew, retaining ring and Loctite. When choosing a method, these are the criteria we emphasize:

- Function
- Price
- Time-consume
- Simplicity in manufacturing

- Simplicity in dismantling

A setscrew requires a hole drilled and threaded through the tension wheel, into the bearing. This will ad pressure to the bearing, keeping it in place both axial and radial. One downside with this solution is that the pressure from the setscrew will cause the bearing to not be centered in the tension wheel. The manufacturing is quite simple, but it requires some time. This method is fully detachable. Regarding cost, setscrews are sold in sets, but we have some available.

The retaining ring requires a placing-track to stay in place. This has to be manufactured in the lathe and requires some time and precision. The downside here, is that the retaining ring only will keep the bearing in place in axial direction, not radial. This method is also fully detachable. As for the setscrew, retaining rings are mostly sold in sets. These sets have a prize of approx. 130 NOK.

Loctite 648 is a kind of glue specially developed for bearings and cylindrical surfaces. This will lock the bearing in both radial and axial direction and is very easy to use. It has a curing time for work of five minutes, and is completely cured after 12 hours. It can be removed by using special substances from the same brand, or by heating it up above the working temperature (180 degrees Celsius). This means that is not easy to dismantle, but it is possible. One bottle of this glue is prized at 179 NOK.

Since all three options has a relatively low cost, this criterion does not apply anymore.

Based on the information above, Loctite is the best fit for this part. This is because it is strong, centers the bearing and locks it in both directions. The fact that it is hard to dismantle is not very important for this specific part and can therefore be neglected.

Conclusion:

After production of the new and improved tension wheel, the assembling and testing begin. Since the new tension wheel is based a lot on the previous design, the assembling of the wheel on the Effector is the same. It is fastened by the bolt and a nut, using washers to create space between the base plate and the nut and the base plate and the bushing. After assembling, testing proceed. This is done by mounting the belt the same way it is mounted when the Effector is operating and tightening the belt using the new tension wheel. When reached the satisfactory belt tension, the system is rotated to visually inspect if the belt is running smoothly and if the tension wheel is capable of resisting the forces coming from the belt. This test gave satisfactory results, showing that the belt runs without major disturbance and that the tension wheel is holding its position. This means that is is capable of resisting the said forces and our goal is reached.

10.3.3 Elevation base

KC

When testing the accuracy of the Effectors elevation, it was discovered that even though the elevation itself was satisfactory accurate within the required deviation of ±0.5° described in the system requirement SR-04-03, the Effector would sometimes struggle to elevate at all. This problem is important to solve, due to the stakeholder requirement FR-04, saying that the Effector shall be able to hit marked targets. If the Effector is not able to elevate, the range within which it can hit targets will be drastically reduced.

After a visual inspection, several reasons for this problem were found. These problems are listed in chapter 10.2. The main issues are how to fasten the axle the Effector elevates about and removal of material hindering the elevation. The main part to be redesigned to solve the elevation issues, is the elevation base. In fig. 10.16, the original elevation base is displayed.

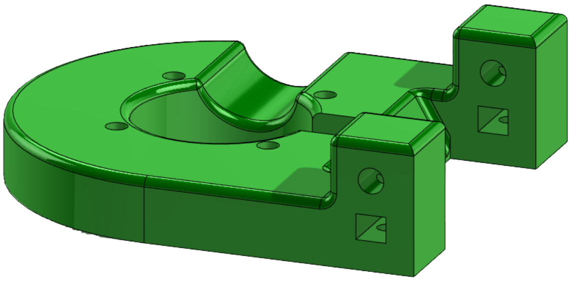


Figure 10.16: Original elevation base

To solve these problems, multiple design alternatives have been developed:

1. The elevation base can be redesigned and produced in aluminum.
2. Keep the elevation base in the original 3D-printed plastic material but replace the setscrews with a custom washer:
   - A: Welded to the axle and using pins to keep still.
   - B: Screwed to the axle from the ends and using pins to keep still.
   - C: Screwed to the axle from the ends and placed in a custom slot to keep still.
3. Keep the elevation base in the original 3D-printed plastic material but replace the setscrews with splints from the front.

Alternative 1:

This alternative will increase the parts strength and ability to withstand greater loads. It will also increase the mass of the Effectors top structure. The original elevation base, without any redesign, has an original mass of 130 grams. Changing the material on this part to 6082-T4 aluminum, the mass will increase 117%, to 282 grams. These values are according to SolidWorks mass properties. This increase in mass can be reduced to 63%, or 212 grams by redesigning the part and removing excess material, as seen in fig. 10.17. With this solution, fastening the setscrews will not be a problem. It will be possible to thread the holes in the elevation base housing the setscrews, making it easier to fasten them the correct way adding the correct amount of pressure to the axle. This will keep the axle from rotating and the Effector will elevate as it should.

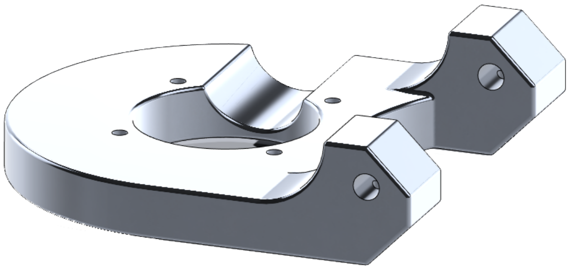
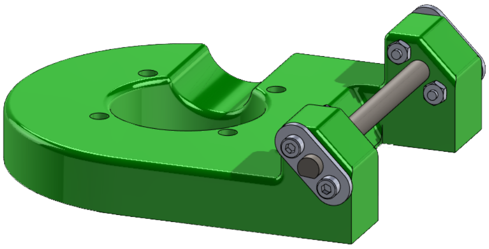


Figure 10.17: Illustration of redesigned elevation base to be produced in aluminum.

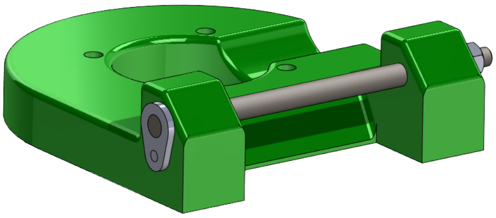
Alternative 2:

The second alternative can be performed in two different ways:

- **A:** Designing a custom washer to be spot-welded on the end of the axle. The axle must be modified to be long enough for this. The custom washer will have enough space for pins to be welded on, securing the axle and its position. The first version of this design used bolts and nuts to fasten the custom washer. This will not be possible without major redesigning of several parts, due to lack of space. Because of this, the bolts were replaced with pins welded to the custom washer. The different versions are seen in fig. 10.18. Since the custom washer can only be welded to one side for it to be possible to disassemble, the other side of the axle will be threaded and secured with a nut. This is seen in fig. 10.18b.



(a) The first idea for design alternative 2A.



(b) The second idea for design alternative 2A.

Figure 10.18: The first and second design idea for alternative 2A.

- **B:** The same as alternative 2A., but instead of welding the custom washer to the axle, it will be fastened by a bolt. This requires a threaded hole at each end of the axle. This is easier to dismantle than alternative 2A. Using this alternative, there will be possible to have pins at both ends, giving extra security for the axle. This is illustrated in fig. 10.19.

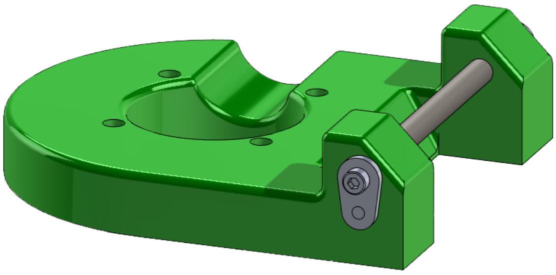


Figure 10.19: Illustration of design alternative 2B.

- **C:** This alternative is similar to alternative 2B, but instead of welding pins to the custom washer for support of the axle, the washer will be fitted in a submerged slot. This does not require welding. The axle needs to be shortened a little bit and the redesign of the elevation base must include the slots. This alternative is seen in fig. 10.20.

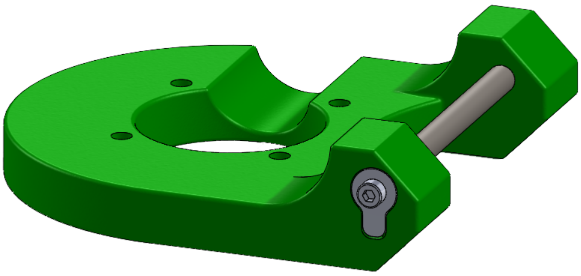


Figure 10.20: Illustration of design alternative 2C.

When considering welding as a manufacturing method, knowledge about what happens to the material when welded is important. The welding process can be divided into three classes: fusion welding, solid state welding and brazing and soldering [107].

Fusion welding uses heat to melt the materials together, before cooling down and resulting in joint materials. In Solid-state welding, the materials are not melted. The materials are joined in different matters, by using for instance friction or explosions. Brazing and soldering uses filler metals and lower temperatures [107].

Because of equipment-access, the welding category that is being focused on in this project is the fusion welding:
When using fusion welding, it is important to be aware that the material will melt and the consequences that follow. During fusion welding, the material divides into three zones: the base metal, the Heat-Affected Zone (HAZ) and the weld metal. The weld metal is the zone where the temperature is above the liquidus limit, and the metal is in liquid form. The HAZ is a zone where the temperature is high, but still not as high as solidus. This means that the metal is still in a solid state, but the microstructure is being affected by the heat. This zone is located next to the weld metal. The base metal is the material not reaching a temperature high enough to affect the microstructure and is therefore the

same as the original material [107].

As mentioned, the HAZ has a different microstructure than the original material due to the high temperature it has been exposed to. This microstructure depends on several factors, such as the temperature reached, and the time used to reach it and/or cool down. The HAZ size and characteristics are determined by the metallurgical factors and the material's physical properties. The way the strength and hardness of the original material was achieved impacts the strength and hardness of the HAZ. For example, if the original material has been cold worked, the heat applied when welding will cause the grains in the materials microstructure to recrystallize. After cold working such as cold rolling, the grains will be elongated in the direction of the rolling. The recrystallization will result in two things: the material further away from the weld metal will recrystallize into fine grains, which is preferred. In the material closest to the weld metal on the other hand, the heat will cause the recrystallized grains to grow in size, which is not preferable. Larger grains cause a weaker and softer material, leaving this to be the weakest point in the joint [107].

The weld metal will, after being exposed to high temperature and, if needed, some kind of filler metal, cool down. When cooling down, the solidification begins: the solidification process begins with the formation of dendritic grains growing parallel to the greatest temperature gradient, meaning towards the heat source. The welding process chosen, the material and alloy and the type of filler metal affects the size and structure of the grains. Even though the cooling rate seems to be high, the great temperature gradient results in rough grains giving the material lower strength, toughness and ductility. This can be improved by adding the correct filler metal or increasing the cooling rate [107].

The changes in microstructure and thermal fluctuation can cause faults in the welded areas. These faults can also be a result of poor education of the operator. Typical faults are porosity, incomplete penetration, cracks and residual stresses [107].

Residual stresses can lead to defects in the weld. Typical defects are distortion, warping and buckling in the material, as well as reduced fatigue life. For this project, these defects are dangerous. Since the parts requiring welding have small dimensions, a case of distortion, warping or buckling is not accepted. Because of this, and the fact that no group member is a trained welding operator, it is preferred to avoid welding [107].

With alternatives B and C, there is minimum to no need for welding, it is easier to assemble and disassemble if needed and the axle will be supported at both ends. This will distribute the loads more evenly. The material used in all three alternatives is steel for both the axle and the washer.

Alternative 3:

The third alternative will replace the setscrews with pins inserted from the front. This requires a drilled hole though the plastic elevation base and through the axle. The pin will be thread through this hole. Machining this will be hard, since it is extremely important that the holes fit each other. Also, the axle is round and only 6 mm in diameter, making it hard to drill in it. Drilling a hole normal to an axle will weaken the axle, this has to be kept in mind if this alternative is chosen. Alternative 3 is shown in fig. 10.21.

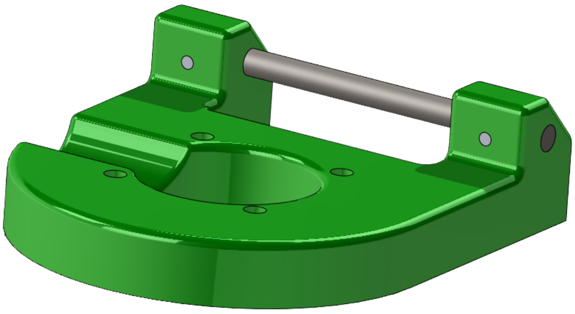


Figure 10.21: Illustration of design alternative 3.

Criteria for decision

When deciding on a design to proceed with, there are several things to account for. The most important aspects for this project are:

1. **Price:** Since there is a budget limit, the price is important. This includes the cost of material, manufacturing and elements that need to be bought.
2. **Time:** Since this project is on a strict time schedule, it is important to save as much time as possible. This includes delivery time for purchased parts and the time it takes to produce both the product and the drawings etcetera.
3. **Simplicity:** How simple the part is to manufacture and how difficult it is to get the required equipment and raw material needed.
4. **Functionality:** The ability to reach physical requirements. In this case, the physical requirement is to keep the axle from rotating.

The different design alternatives are compared on these four criteria, as seen in tables 10.1, 10.2, 10.3 and 10.4 .

Table 10.1: Comparing alternatives regarding price.

Price	
Alternative 1	This part will be sent to an external workshop for production in a Computer Numerical Control (CNC) machine. The workshop has agreed to help us with this at a low cost. There is no need for other elements or tools, keeping the price low.
Alternative 2A, B and C	Since material and tools are available at the USN workshop, these parts will also have a low cost. Required bolts and nuts must be purchased, but these parts are not very expensive.
Alternative 3	Since material and tools are available at the USN workshop, these parts will also have a low cost.
Conclusion	No alternative stands out as the lowest cost. All alternatives are in the lower price range.

Table 10.2: Comparing alternatives regarding time.

Time	
Alternative 1	Since this part will be sent to an external workshop for production, it will be less time-consuming for the group. When the part is sent, the group can focus on other tasks until it is done, giving the opportunity to be productive while the part is being manufactured. One possible issue to be aware of, is that this makes the project dependent on an external part.
Alternative 2A	This part has to be produced in the USN workshop using multiple tools and machines, including welding. Because of this, it will be a time-consuming production. The risk of the material deforming when welding is also present. If this happens the production must start over.
Alternative 2B	This part also has to be produced in the USN workshop, but with less welding. Only one weld reduces the risk of the material deforming, but it does not eliminate the risk.
Alternative 2C	This part has to be produced in the USN workshop, but it does not require welding. Excluding welding makes this production less time-consuming than alternatives including it.
Alternative 3	This part also has to be produced in the USN workshop, but with simpler tools than alternative 2A and B. Despite this, the importance of being thorough will require some time.
Conclusion	Alternative 1 is the least time-consuming alternative for the group, even though it will make the project dependent on the external part. If the external part is not able to deliver, alternative 2C is the least time-consuming self-produced option.

Table 10.3: Comparing alternatives regarding simplicity.

Simplicity	
Alternative 1	When sending this part to an external workshop, it will be produced in a CNC-machine. This makes the production very simple for the group, not needing to do much other than delivering the model and drawings to the workshop.
Alternative 2A	This alternative requires welding at multiple points. Even though it is only spot-welding, it is an advanced manufacturing method requiring practice. Both the axle and the custom washer has to be produced as well. The axle must be extended and threaded in one end. The washer, axle and pin must be welded together.
Alternative 2B	This part requires welding, but not to the extent alternative 2A does. This makes it simpler to produce than that alternative. Both the axle and the custom washer has to be produced here as well. The axle must be drilled in each end and these holes must me threaded. The washer and the pin must be welded together.
Alternative 2C	This alternative does not require welding, leaving the only parts to be produced to be the washer and the axle. The axle needs to be shorter than it is today, and it must be drilled in each end and these holes must me threaded.
Alternative 3	Even though the design is easier, the importance of the drilled holes tolerances makes this part hard to manufacture. The axle has a diameter of 6 mm, making it difficult to drill a correct hole normal to it.
Conclusion	Alternative 1 is the simplest alternative for the group, since it does not require hands on manufacturing from any group member. If the group has to produce this part, alternative 2C is the simplest alternative due to no welding.

Table 10.4: Comparing alternatives regarding functionality.

Functionality	
Alternative 1	This alternative will make sure that it is possible to tighten the setscrews enough to keep the axle from rotating. The part will increase in mass, making it heavier to rotate.
Alternative 2A	The pin supporting the axle should be enough. Because of the welding, there will only be pin on one side. This will distribute the loads unevenly, which is not desired. If not welded correctly, the pins may break of.
Alternative 2B	When inserting pins on both sides, the loads will be distributed more evenly, making this alternative more secure than alternative 2A. If not welded correctly, the pins may break of.
Alternative 2C	The only thing keeping the axle in place in this alternative is the 3D-printed edge in the submerged slot. If printed with the correct settings this should be enough. The dimension on this edge is dependent on the thickness of the washer. If the thickness increases, so does the strength.
Alternative 3	Pins normal to the axle will keep it from rotating, but the narrow dimensions can be an issue.
Conclusion	All alternatives will keep the axle from rotating, but alternative 1 is the most secure option.

Based on these four important aspects, it is clear that as long as there is an option to send the part to an external workshop, this is the alternative that is best for this ongoing project. This is because it liberates time from production to focus on other aspects of the project without being a pricey option. It is also simple to produce for the group, since the only production done by any group member is the design of the 3D model and the making of the production drawings. This alternative is also the one best suited for the required task since the functionality of the other alternatives depends on the quality of the manufacturing. If this opportunity is no longer available or the external workshop cannot finish it within the needed time frame, alternative 2C is the next best choice. This is because it is the easiest and least time-consuming to produce by the group members in the USN workshop.

The chosen design:

Both alternative 1 and alternative 2C seems to be a good fit for the project. When comparing the five alternatives, alternative 1 turned out on top. Since this alternative makes the project dependent on a third part, alternative 2C is chosen in addition, in case the external workshop is not able to complete production in time. Alternative 2C is the second-best option and will be produced in the USN workshop as a backup solution. Since both alternative 1 and 2C are the preferred options, different FEA were run on these parts to make sure they can be exposed to the existing loads without failure. The results from these analyses is that both part will survive the loads, one with a better margin than the other, and fatigue will not be an issue. How these analyses were run can be read in section Elevation base.

Production

As mentioned earlier, the elevation base from alternative 1 will be produced at an external workshop using a CNC-machine. For the CNC-operator to be able to produce the desired parts, a 3D model and a 2D production drawing are required. After redesigning the elevation base using SolidWorks, the 3D model was saved as a .STEP file ("Standard for the Exchange of Product Data" also known as ISO 10303). This type of file is an ISO standard exchange format and can be read by several different programs and software [108]. The 2D production drawing was also produced and can be seen in attachment xx. The expected production time for this part is one full workday, and the CNC-operator requested one week to complete the job. The extra time is because the operator is doing this job as a favor and has other work to do also. From the drawings were delivered to the workshop it took approx. one and a half week to receive the finished part. Fig. XX Shows the new aluminum part compared to the old 3D-printed part. Since the external third part workshop managed to deliver this part well within the projects time frame, the need to produce alternative 2C is no longer present. This means that the design is finished and the FEA has been run, but the physical production will not take place.

10.3.4 Projectiles

KC

One of the elements in need of improvements from last year was the projectiles used when firing the Effector. The projectiles were developed using OpenRocket [4] and manufactured by 3D-printing using PLA. The projectiles were both designed and printed as one part. This means that the fins were attached to the body from design, not mounted post production. The print direction was vertically standing up, which is the weakest print-orientation. As seen in appendix Y, the yield strength of tough PLA at this orientation has an average of 32.67 MPa, compared to the horizontally flat orientation with an average of 52.36 MPa. Since the biggest issue with the projectile was that they broke when hitting targets, the print orientation was reviewed.

As mentioned earlier, the projectile body usually broke between the layers of printed PLA, not necessary in the material itself. Because of this, alternative ways to produce these projectiles were therefore investigated. Since new production methods were discussed, the desire to redesign appeared.

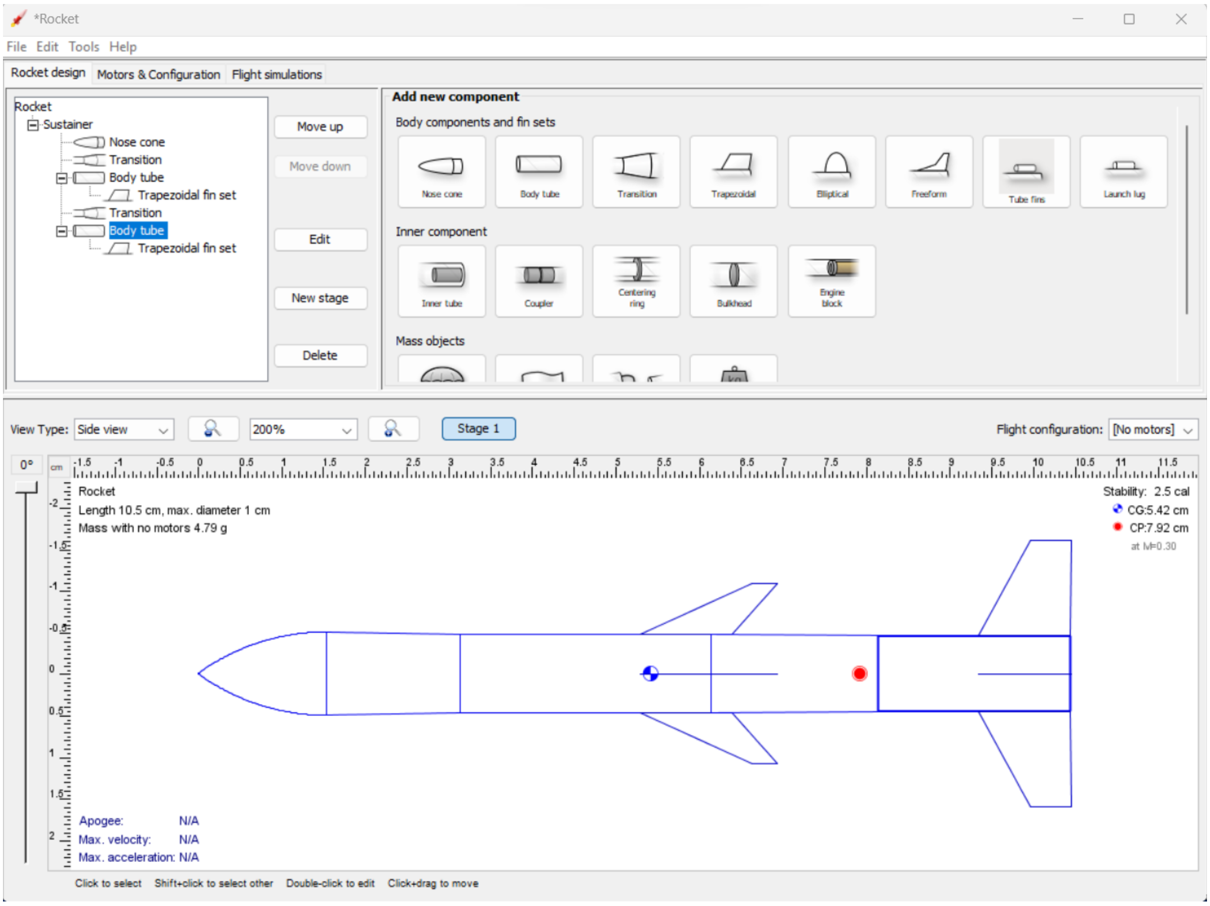
Redesign

The design of the projectiles were initially OK. The reason redesign became a subject, was because the need of a pointier nose cone appeared. Also, resembling the Advanced Medium-Range Air-to-Air Missile (AMRAAM) used by the real life NASAMS was desired. The AMRAAM has two sets of four fins, resulting in a total of eight fins.

The initial projectile were looked at through reverse engineering, and several aspects of the design were reused. This includes the dimensions. The projectile has an OD of 10 mm and an ID of 6.5 mm. The length is set to 10.5 mm. These values has been used when designing the new projectiles, because they fit in the projectile chambers.

When designing the new version of the projectile, OpenRocket [4] was used. This is, according to their website "a free, fully featured model rocket simulator that allows you to design and simulate your rockets before you build and flying them" [4]. This program can be seen in fig. 10.22. In this figure, the Center of Gravity (CG) and the Center of

Pressure (CP) for the designed projectile is also shown: CG is illustrated by the blue dot, CP by the red. CP is "the point through which the resultant force acts" [109].



View Type: Side view

200%

Flight configuration: [No motors]

Rocket
Length 10.5 cm, max. diameter 1 cm
Mass with no motors 4.79 g

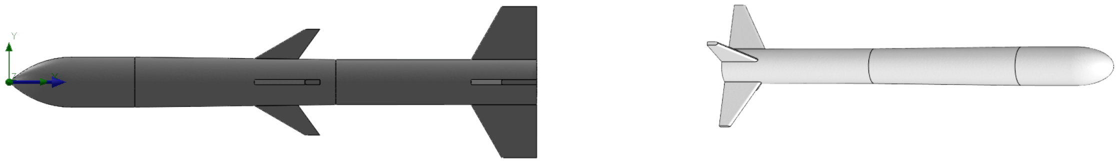
Stability: 2.5 cal
CG:5.42 cm
CP:7.92 cm
at M=0.30

Apogee: N/A
Max. velocity: N/A
Max. acceleration: N/A

Figure 10.22: OpenRocket [4] feed.

The relationship between CG and CP is important. When CP is closer to the projectiles front than CG, the projectile is unstable. This means that disturbance to the projectile in flight would cause it to deviate from its original position. When CP is located further back than CG, the projectile is stable. This is because a disturbance to the projectile in flight would generate a restoring force, bringing the projectile back to its original position [109]. According to OpenRocket, CG of this projectile is located 54.2 mm from the projectiles front, while the CP is located 79.2 mm from the same reference. This makes sure the projectile is theoretically stable.

The new projectile design was, as mentioned, done using OpenRocket [4]. To be able to produce the actual projectile, the model had to be converted to SolidWorks. Since there is no direct way to do this, the SolidWorks model had to be drawn from scratch using the dimensions from OpenRocket. The final model is shown in fig. 10.23a.



(a) The new projectile displayed in SolidWorks.

(b) The old projectile displayed in SolidWorks.

Figure 10.23: New and old projectile illustrated.

As seen in fig. 10.23, the new and old projectiles has some differences. Even though the basic dimensions are the same, there are three distinctively different elements:

- The new projectiles nose cone is significantly sharper than the old one. This has been altered because the target to be hit by the projectiles are defined as balloons. To pop the hostile targets, the projectile needs to be sharp.
- The new projectile has an extra set of fins located approx. at the middle of its body. This is to resemble the AMRAAM.
- The new projectiles back fins are larger than the old ones. This is done to increase stability and to resemble the AMRAAM
- Even though it is not seen in the illustration, one major difference is that the old projectiles were produced using an FDM-printer and PLA material. The new projectile is produced in Nylon 12 using an SLS-printer. This solves the problem regarding poor adhesion between the layers of print.

10.3.5 Sensor placement and enclosure

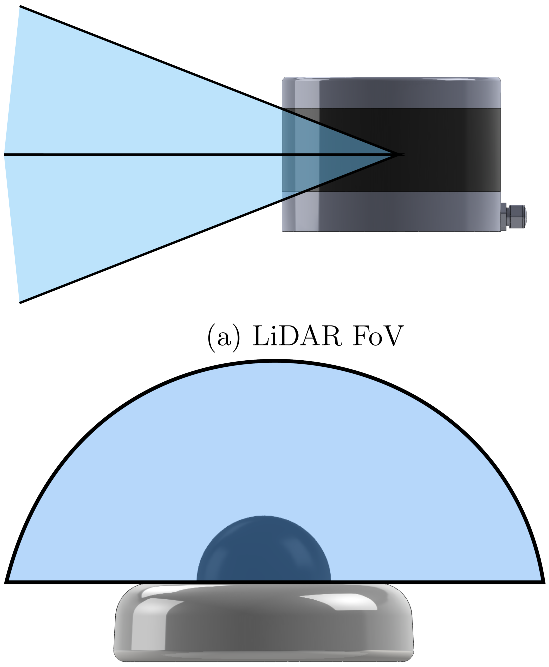
KC

To ensure the approval of the system requirement SR-11, the process of designing an Enclosure for the Sensors is initiated. This requirement states that “the Sensor should have an enclosure”. This is necessary both to protect the sensor, but also to ensure the Sensors’ placing in relation to the Effector.

When initiating this process, some criteria must be respected:

1. The Enclosure must ensure that the Sensors will be placed at the exact same distance from the Effector at all times.
2. The LiDARs field of view and the cameras and Effectors range must intersect.
3. Minimize the percentage of blind spots both for the Sensors and the Effector.

The Sensors used is, as earlier mentioned, a Velodyne VLP-16 LiDAR and an AXIS M5074 PTZ camera. The LiDAR has a FoV of 15° below and above the horizontal axis, giving a total FoV of 30° in any direction. The cameras’ only limit is no range below the horizontal axis. Fig. 10.24 illustrates both the LiDAR and the cameras range.



(a) LiDAR FoV

(b) Camera range

Figure 10.24: Illustrating the LiDAR FoV and camera range.

The initial thought involves the Effector placed next to, but in the same plane as the Sensors, while placing the Sensors on top of each other. The camera is placed on the ground and the LiDAR is placed at a suitable height. This placing would make sure all ranges intersect and keep the blind spots to a minimum but requires the Effector to know where it is relative to the Sensors. This placing gives the Effector and Sensors different coordinates in the X and Z axis, only corresponding in the Y axis.

To simplify this, an idea of where the Effector and Sensors placing corresponds in two axis is investigated. If the Effector and the Sensors are placed directly above each other, they are placed on the Z axis, at different Z coordinates but the same XY coordinates. When pursuing this idea, it is important to investigate the intersection of the different ranges and at which height to place the Sensors.

Placement of camera and LiDAR

Since the camera range does not include anything below the horizontal axis, as shown in fig. 10.24b, this ether has to hang upside down from a significant height, or be placed as close to the ground as possible. Since placing the camera upside down restricts the line of sight above ground, this is not preferred. The decision to place it as close to the ground as possible was therefore made. This implies that the LiDAR has to be placed above the camera. Due to the Effector being placed directly underneath the camera, the camera must be placed as far above ground as the Effector is tall when elevated. From measuring the physical Effector, it is concluded that this height must be approx 0.5 m above ground. This means that the camera can not detect objects located closer than 0.5 m to the ground. This implies that the Effector must be elevated when firing at identified objects.

The LiDAR has a more restricted FoV than the camera and needs therefore more accurate investigation to ensure the best possible placing. According to system requirement SR-04-01, The Effector shall hit marked targets with a diameter of 30 ± 5 cm at a range of 2.5 ± 0.5 m. It is known that the LiDAR has a FoV of 30°. Using trigonometry, it is possible to determine at which height the LiDAR should be placed to be able to detect objects on the ground at a distance of x = 3.0 m:

$$\frac{y}{\sin a} = \frac{x}{\sin b}. \tag{10.11}$$

As seen in fig. 10.25, a represents the angle known from the LiDAR: $a = 15°$. b represents the unknown angle found by summing up the degrees within the triangle: $b = 180° - 90° - 15° = 75°$. x is already known to be 3.0 m.

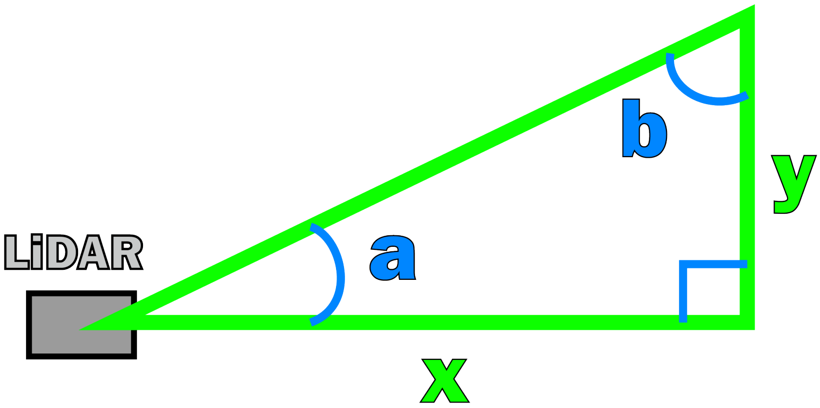


Figure 10.25: Illustration of values used in (10.11)

Using eq. 10.11, the height y can be found.

$$\frac{3\,\mathrm{m}}{\sin 75} = \frac{y}{\sin 15} \rightarrow \frac{3\,\mathrm{m}}{\sin 75} \cdot \sin 90 = 0.8\,\mathrm{m}. \tag{10.12}$$

From eq. 10.12, the LiDAR will detect an object on the ground at a distance of 3.0 m if placed 0.8 m above ground. this gives the LiDAR a total vertical range of 1.6 m above ground. If placing the LiDAR at a higher altitude, the LiDAR also detects objects at a higher altitude, but is unable to detect at ground level at a distance of 3.0 m. Since the cameras range is restricted to 0.5 m above ground, there is no need for the LiDAR to detect objects here. If the LiDAR is placed approx. 1.0 m above ground, it is able to detect objects from 0.2 m above ground to 1.8 m above ground.

When investigating the intersection between the ranges of the Effector, the LiDAR and the camera, SolidWorks is a suited tool. Fig. 10.26 illustrates the intersection.

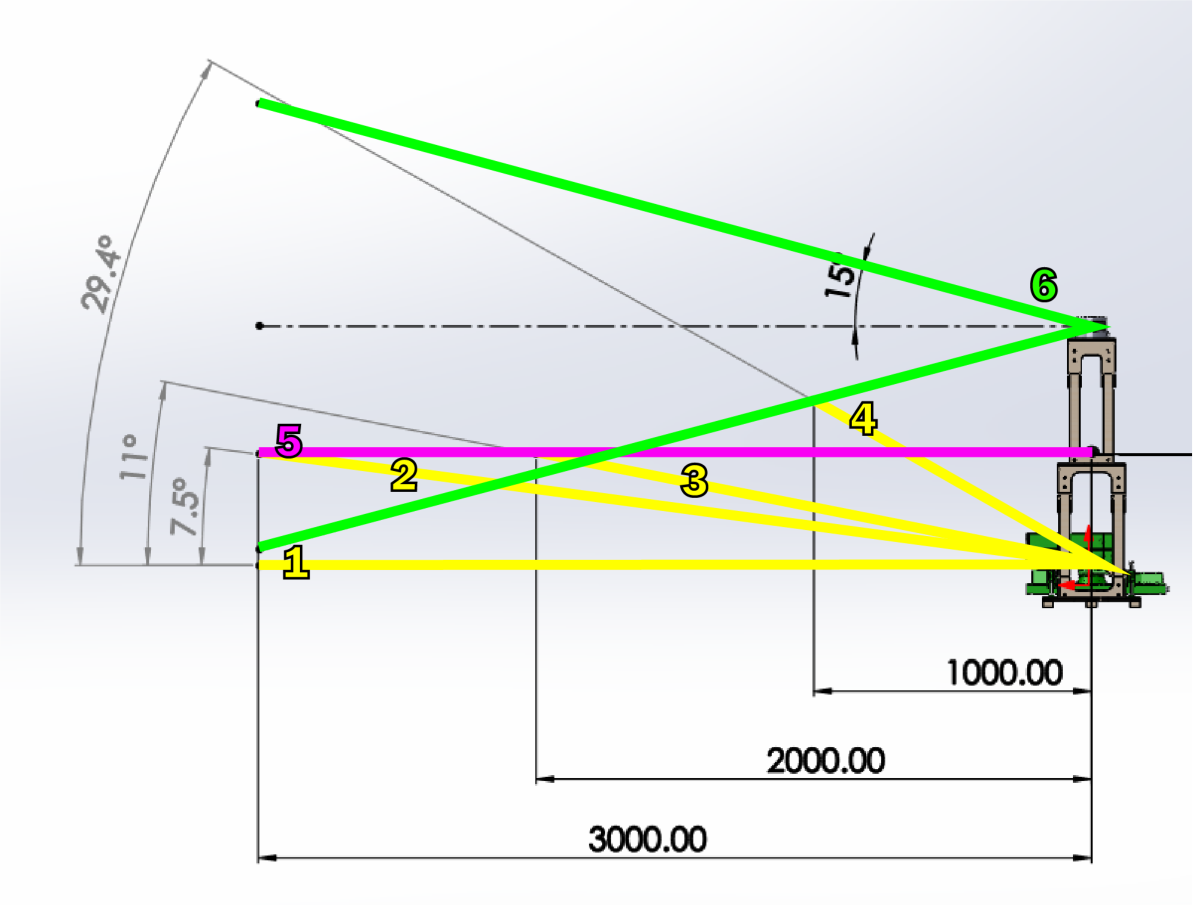


Figure 10.26: Intersection between the Effectors, LiDARs and cameras range.

In fig. 10.26 the green lines (6) represent the LiDARs FoV, the pink line (5) represents the cameras range and the yellow lines represent the Effectors degree of elevation. Line number 1 shows the Effectors range at 0°. It is clear from the illustration that the three elements never intersect at this state. If the Effector is elevated approx. 7.5°, the elements range intersects at approx. 3.0 m. For the three elements to intersect at 2.0 m, the Effector must be elevated to at least 11° as illustrated by line 3. Line 4 illustrates that the Effector must be elevated at approx 29.4° for the ranges to intersect at a distance of 1.0 m. This shows that the closer an object is to the Enclosure, the steeper the elevation angle of the Effector must be. It also shows that placing the camera approx. 0.5 m and the LiDAR approx. 1.0 m above the Effector is an acceptable placement.

Design

After establishing the placing of the Sensors, the design phase initiated. The most important factors to keep in mind when designing the Enclosure is:

1. The Enclosure must be steady enough to handle the vibration from the LiDAR, camera and Effector, even at a height of 1.0 m.

2. The Effector is placed at the bottom. The Enclosure must have enough room for the Effector to be able to rotate 360° and elevate to its limit.

3. The minimization of blind spots is important, but not inevitable. The structure can have a maximum of four pillars, one in each corner of the Effector.

4. The Enclosure must be simple to assemble and disassemble and be portable.

The chosen material is wood pillars and plywood. This are materials that are easy to work with and adjust as desired. They also have a satisfactory strength. The cross section of the pillars used are 36x23 mm and the plywood has a thickness of 6 mm. Every part made of plywood is cut out in an Epilog laser cutter.

The structure starts with a bottom plate designed with cut-outs to the Effectors support legs. This is so that the Effector is stationary at the same spot always. at each Effector corner, pillars are placed, forming the base for the structures second floor. Base plates and structural support are made from plywood. At the second floor, the camera is placed directly above the Effectors rotation base. The camera is also surrounded by four pillars creating the base for the third and last floor. This is where the LiDAR is placed. Both the camera and the LiDAR has their own holders to make sure they are always mounted at the same place. The structure is constructed to be disassembled into five modules: The three base plates are separate modules, as well as the two different side structures. Most of the modules are also possible to disassemble. This makes the Enclosure portable. Fig. 10.27 shows the final model of the Enclosure with the Effector and the Sensor mounted. Drawings showing the different modules and the total structure can be seen in Appendix E thru J.

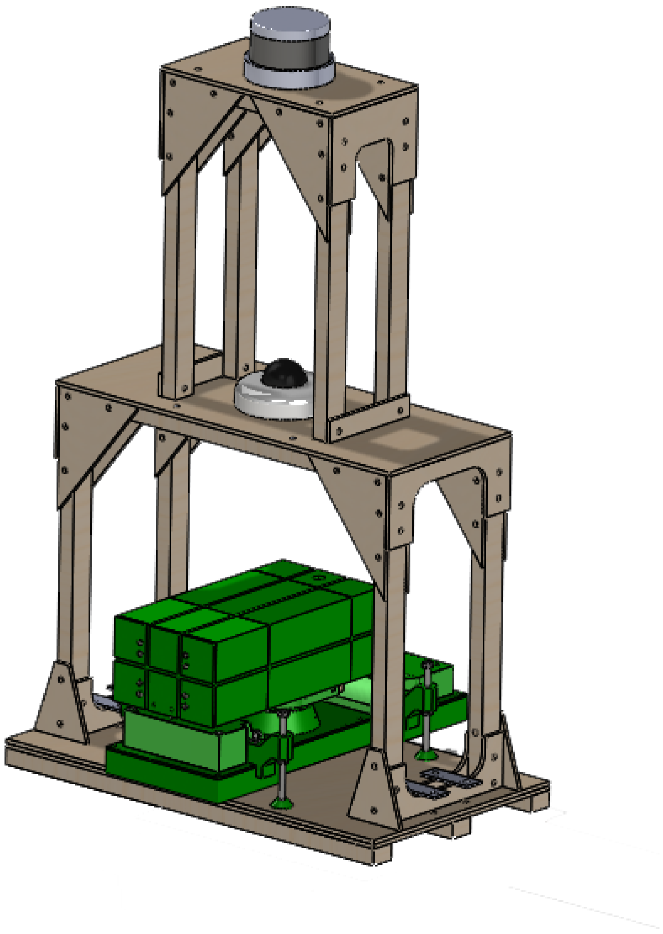


Figure 10.27: Illustration of the finished Enclosure mounted with Effector and Sensors.

With this design, the Enclosures criteria has been fulfilled. The structure is steady enough to handle vibrations from the different components. In addition to the steady structure, the LiDAR is placed in its own holder, standing on top of approx. 4 mm of vibration absorbing material. The Enclosure is also big enough to house the Effector, giving it the freedom to rotate and elevate as much as needed. The number of blind spots has been kept at a minimum, even though the result was a bot more than what was initially wanted. Since this is a proof of concept, this is not a problem and the system not noticeably disturbed by the blind spots. It is also made sure that the Effector has at least a 90° range at the front and back. Due to the modules, the Enclosure is also both possible to disassemble and portable.

10.4 Finite Element Analysis (FEA)

KC

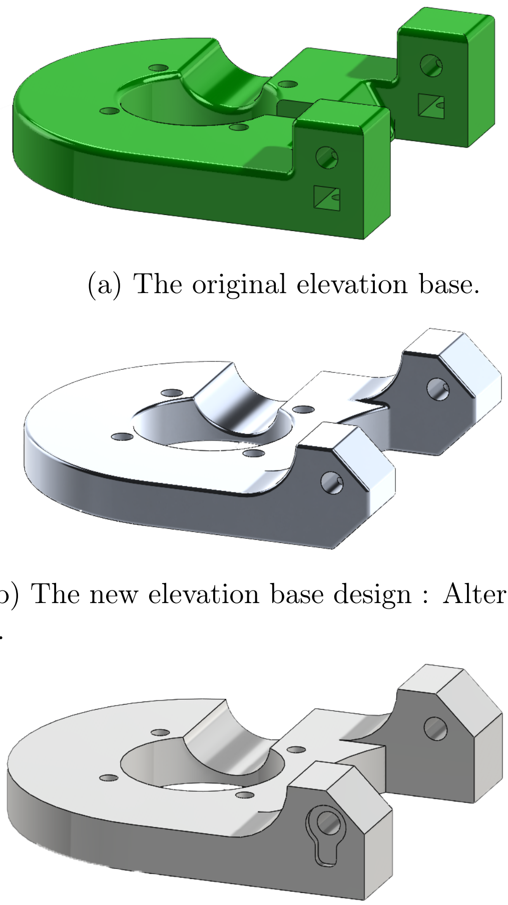
Finite Element Analysis (FEA) is a digital simulation run on 3D-models to investigate how the proposed design will act under the influence of the current conditions. In the simulation, fixtures and loads representing the physical environment are introduced to the 3D-model of the desired part. The result of these analyses gives a good pointer to whether or not the design will function properly under the expected conditions when produced.

10.4.1 Elevation base

KC

After deciding on a new design, different Finite Element Analysis (FEA) were run to make sure the design will hold the loads it will be exposed to. The FEA will show whether or not the designs chosen will withstand the expected loads. The designs chosen are alternative 1 and 2C. In alternative 1 the elevation base is produced in aluminum and redesigned to reduce the mass. In alternative 2C the elevation base is produced by 3D-printing and redesigned to keep the axle from rotating in a different secure way. The design processes and the choices are explained in chapter 10.3.3.

First, the old and the new designs will be compared. As seen in fig. 10.28, the three designs have many similarities.



(a) The original elevation base.

(b) The new elevation base design : Alternative 1.

(c) The new elevation base design : Alternative 2C.

Figure 10.28: The original and the new design of the elevation base.

The FEA to be run on these parts is one static analysis for when the Effector is in starting position (elevated 0°), one static analysis for when the Effector is fully elevated to 40°, one fatigue analysis representing the Effector moving from 0 to 40 to 0° multiple times and one frequency analysis investigating the models natural frequency. The fatigue and frequency analysis will only be run for the aluminum part.

Before starting the simulation, it is wise to reflect over the possible outcomes. When looking at the old model, a few areas of interest are seen. For instance, the radius between the body and the ear is small. This can cause a tension build-up in the transition between the two elements. Also, the body is massive with a thickness of 15 mm. The space cut out for the pressure-air pipe locally reduces this thickness, possibly weakening the part and creating another tension build-up in the clearance hole located between the reduction and the area affected by the loads. This part was originally designed to be 3D-printed.

The new design, the radius between the body and the ear is significantly greater. This is done to avoid the possible tension build-up. The thickness of the body is reduced by three millimeters. This means that the material left after making space for the pressure-air pipe will be even thinner than the original. This means that the tension build-up in the clearance hole may be greater than in the original part. The FEA's are run to make sure this will not be a problem. As a rule, an FEA is not approved unless the maximum tension in the part is less than the materials yield strength when including a safety factor of at least 1.2.

The first analysis: New and old design at 0°

The first FEA run, is the static analysis representing the Effector at starting position, meaning not elevated. This analysis is run on both the new design alternatives and the old version of the part. When running the analysis, the material aluminum 6082-T4 will be used for alternative 1 and the old design. This is to compare the designs using the material the new elevation base will be produced in. Since this is the preferred alternative, this is the one being compared to the old version. The material used in the analysis of alternative 2C is a polymer.

When figuring out which polymer to use for alternative 2C, the first step was to find out which materials are at hand. In this case, the choice stands between three options: ABS, Tough PLA and Nylon 12. To figure out which one of these are best suited for this part, a tension test was conducted to find the materials yield strength. How the test was executed and the results can be read in appendix Y. From these test results, it is concluded that the Nylon 12 material will be the best fit for this part, since this is the material with the highest and most evenly distributed yield strength.

Before running the analysis, it is vital to identify all loads affecting the part. In this case, there are three loads:

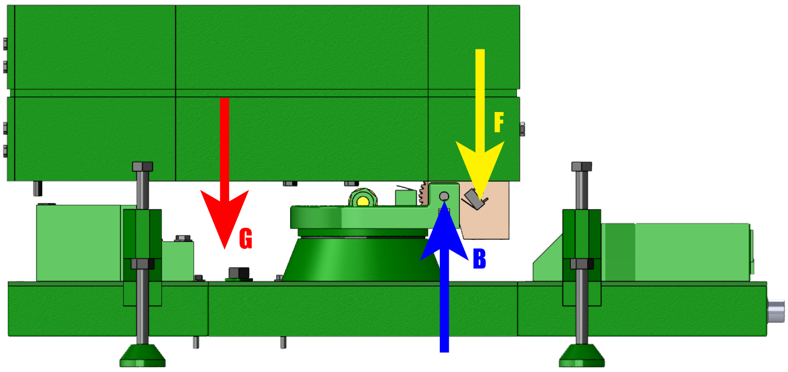


Figure 10.29: The forces affecting the elevation base when not elevated.

As seen in fig. 10.29, the elevation base is affected by the top structures' gravitation force G, the force from the worm gear F and the force from the "bukk" B, resting on the axle. The force G was found from $G = m \cdot g$, where m is the mass of the top structure. This is set to 5105.3 grams, found in SolidWorks "mass properties" evaluation tool. The value g is the gravitational acceleration defined as 9.81 m/s^2. By using this, G is found:

$$G = mg = 5.1053\,\text{kg} \cdot 9.81\,\text{m/s}^2 = 50.08\,\text{N}.$$

The distance between the attack point of G, the Center of Gravity (CG), and the axle the Effector elevates around is $x_1 = 99.171$ mm. The distance between F and the axle the Effector elevates around is $x_2 = 15.5$ mm. These values are from SolidWorks measuring tool.

Since the summation of the torques revolving point B equals to zero, it is possible to determine the force F:

$$\sum M_B = 0 \rightarrow Gx_1 - Fx_2 = 0. \tag{10.13}$$

Solving for F in (10.13):

$$F = \frac{Gx_1}{x_2} = \frac{50.08\,\mathrm{N} \cdot 0.099171\,\mathrm{m}}{0.0155\,\mathrm{m}} = 320.4\,\mathrm{N}. \tag{10.14}$$

Knowing this, the summation of forces in the vertical direction can be used to determine the force B:

$$\sum F_y = 0 \rightarrow B - G - F = 0. \tag{10.15}$$

Solving for B in (10.15):

$$B = G + F = 50.08\,\mathrm{N} + 320.4\,\mathrm{N} = 370.5\,\mathrm{N}. \tag{10.16}$$

These three forces affect the elevation base by being transferred to it through the axle. This means that forces G and F will create a torque revolving the axle and force B will work directly on the axle. These loads will be used during the first FEA of these three parts.

When running the analysis, the part needs to be fixed somewhere to represent the way it is supposed to be mounted. In this case, all four clearance holes used to connect the elevation base and the rotation cylinder will be fixed. This is because in reality, these parts will be connected by screws and be unable to move relatively to each other.

Fig. 10.30 illustrates the placing of the fixture and the loads when running the FEA. The green arrows in the clearance holes represent the fixed geometry and the purple arrows in the axle-space represent the different loads. The purple arrows pointing upwards represent B, the purple arrows rotating clockwise represents G and the purple arrows rotating counterclockwise represents F. Since the summation of the torques revolving around one point is zero, the torque from G and F will cancel each other and equal zero. However, to show what they represent it is chosen to keep them in the analysis.

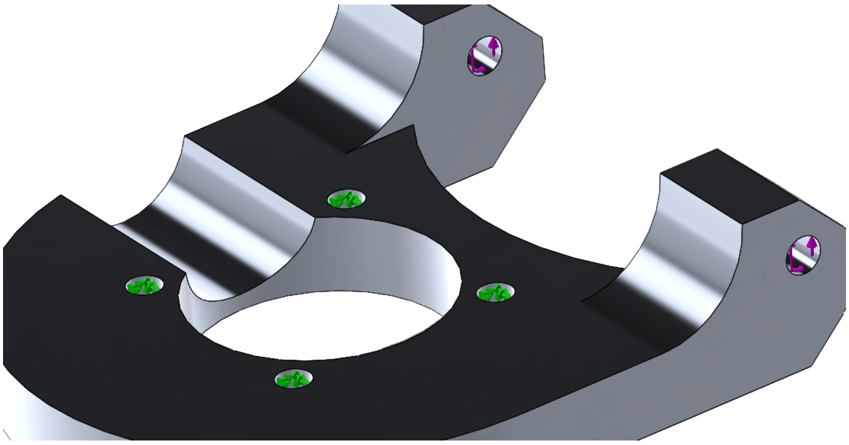


Figure 10.30: The loads placed on the elevation base.

The safety factor chosen when running analysis on alternative 1 is $n = 2.5$. This means that M_G, M_F and B will equal:

$$M_G = Gx_1n = 50.08\,\mathrm{N} \cdot 0.099171\,\mathrm{m} \cdot 2.5 = 12.42\,\mathrm{Nm} \tag{10.17}$$

$$M_F = -Fx_2n = -320.4\,\mathrm{N} \cdot 0.0155\,\mathrm{m} \cdot 2.5 = -12.42\,\mathrm{Nm} \tag{10.18}$$

$$B_n = Bn = 370.5\,\mathrm{N} \cdot 2.5 = 926.25\,\mathrm{Nm}. \tag{10.19}$$

When running the analysis on alternative 2C, the safety factor used are different. This is set to be $n = 1.4$. In this case, M_G, M_F and B will equal:

$$M_G = Gx_1n = 50.08\,\mathrm{N} \cdot 0.099171\,\mathrm{m} \cdot 1.4 = 6.95\,\mathrm{Nm} \tag{10.20}$$

$$M_F = -Fx_2n = -320.4\,\mathrm{N} \cdot 0.0155\,\mathrm{m} \cdot 1.4 = -6.95\,\mathrm{Nm} \tag{10.21}$$

$$B_n = 370.5\,\mathrm{N} \cdot 1.4 = 518.7\,\mathrm{N}. \tag{10.22}$$

These are the total loads. As seen in fig. 10.30, they are distributed equally between the two areas at which the axle is in contact with the elevation base.

After determining the forces, the model has to be meshed. This means that SolidWorks divides the part into small elements used to calculate tension. The smaller the elements, the more precise calculations. For this model, a fine high-quality mesh is chosen, creating second degree elements. The mesh used is a standard mesh with a global size of 2 mm and a tolerance of 0.1 mm. As seen in fig. 10.31, the mesh is uniform, not focusing on any specific areas. To check if the mesh is fine enough, the aspect ratio can be inspected. According to SolidWorks "The aspect ratio of an element is the ratio between the longest edge and the shortest normal dropped from a vertex to the opposite face, normalized with respect to a perfect tetrahedral" [110]. The perfect aspect ratio is 1.0. If over 90% of the elements in the mesh have an aspect ratio below 5, the mesh is considered high-quality. For this particular mesh, 99.8% of the elements have an aspect ratio below 3, making it a high-quality mesh [110].

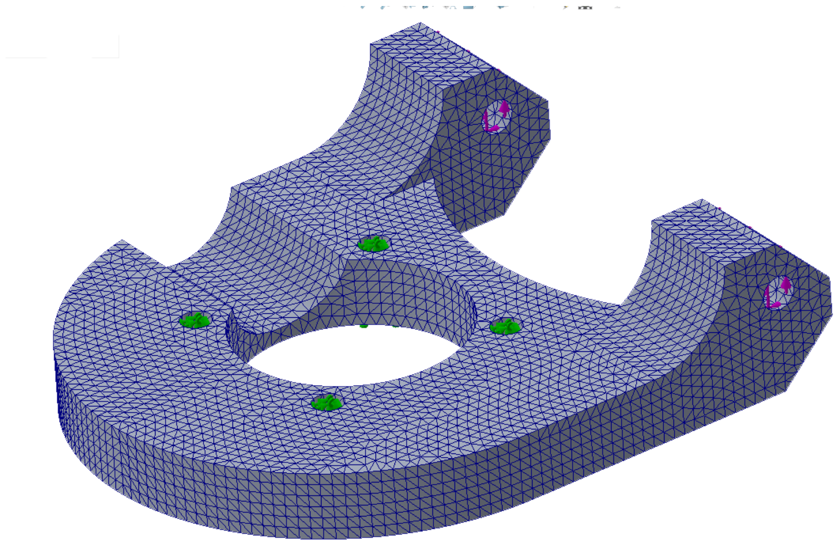


Figure 10.31: The elevation base after being meshed in SolidWorks.

When the part is fully meshed, the analysis is run. Fig. 10.32 shows the result from alternative 1.

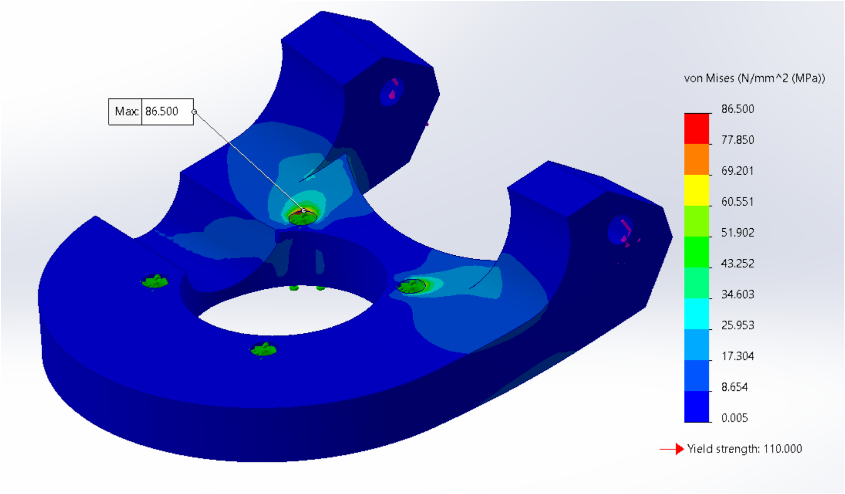


Figure 10.32: The aluminum elevation base after the analysis is run.

The material used, aluminum 6082-T4, has a yield strength of 110 MPa. The maximum tension to occur in the part is 86.5 MPa, including a safety factor of 2.5. In conclusion, this part will be able to withstand the loads it will be exposed to without any trouble.

Then, the analysis of alternative 2C is run. In this analysis, the lowest average yield strength of Nylon 12 from the tension test found in appendix Y were used. The result of the FEA is seen in fig. 10.33.

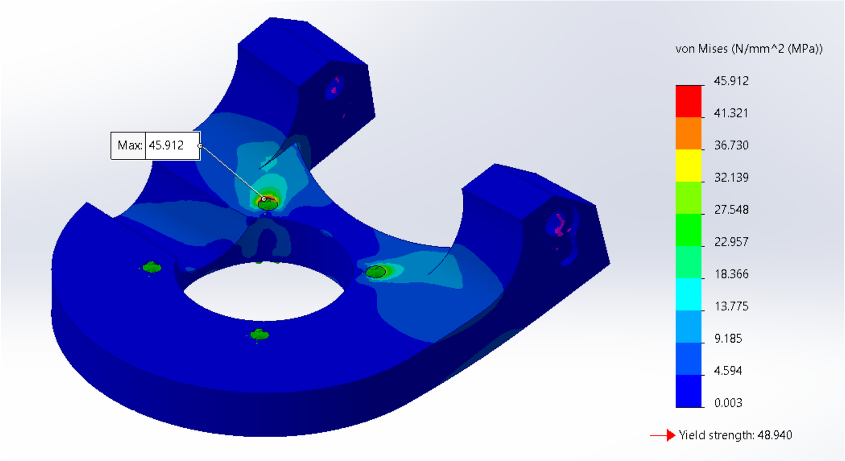


Figure 10.33: The Nylon 12 elevation base after the analysis is run.

This result shows that the maximum tension occurring in the part will be 45.9 MPa, while the materials yield strength is 48.9 MPa. This means that the design will withstand the loads, including the safety factor of 1.4.

When running the analysis of the old design, the exact same settings were used: the same loads and the same mesh. The material chosen was aluminum even though the original part was produced using 3D-print. This was to ensure a true comparison between this part and alternative 1. Fig. 10.34 shows the result of the analysis of the old design.

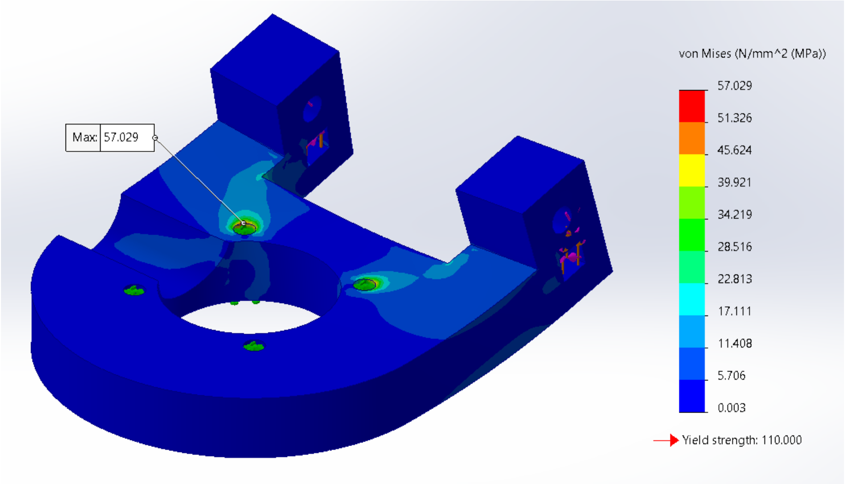


Figure 10.34: The original elevation base after the analysis is run.

The maximum load occurring in this part is 57.03 MPa. This is approx. 34% less than for alternative 1. This is because when designing the new part, the desire to reduce the mass was greater than the desire to keep the maximum tension as low as possible. Also keep in mind that the original part was designed for 3D-printing. The total mass reduction, based on the information found when using SolidWorks' mass properties, is 28%. An analysis was run using a test-model with the new design but original body thickness. The maximum tension in this part was 55.4 MPa, which is slightly less than the original design. This proves that the difference between the strength of the two designs is not very significant and that the redesign is a success.

The first FEA shows that both alternative 1 and 2C will withstand the loads they will be exposed to when the Effector is in starting position.

The second analysis: New and old design at 40°

The Effector can be elevated to a maximum of 40°. This will affect the loads working on the elevation base. The gravity of the top structure G and the force B from the "bukk" will still be vertical, but the force F between the gears gets affected by the angle α. This gives two components of the F force: the x and y component. Fig. 10.35 illustrates the forces working on the Effectors elevation base when fully elevated.

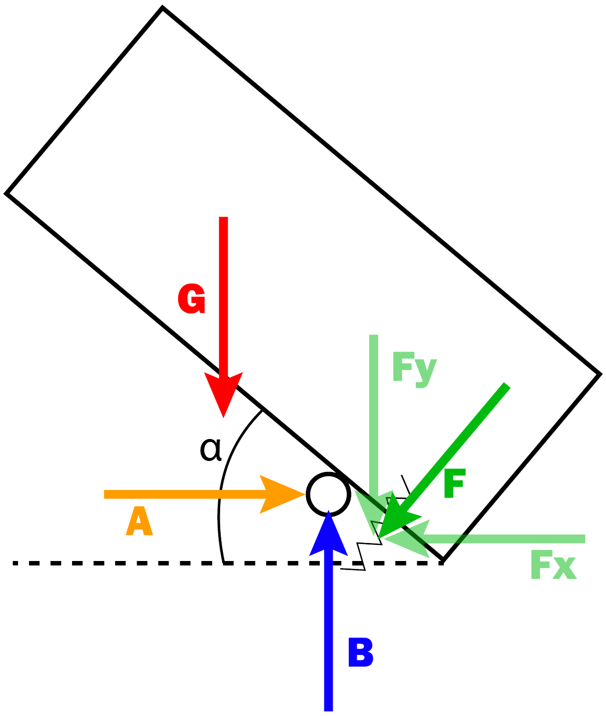
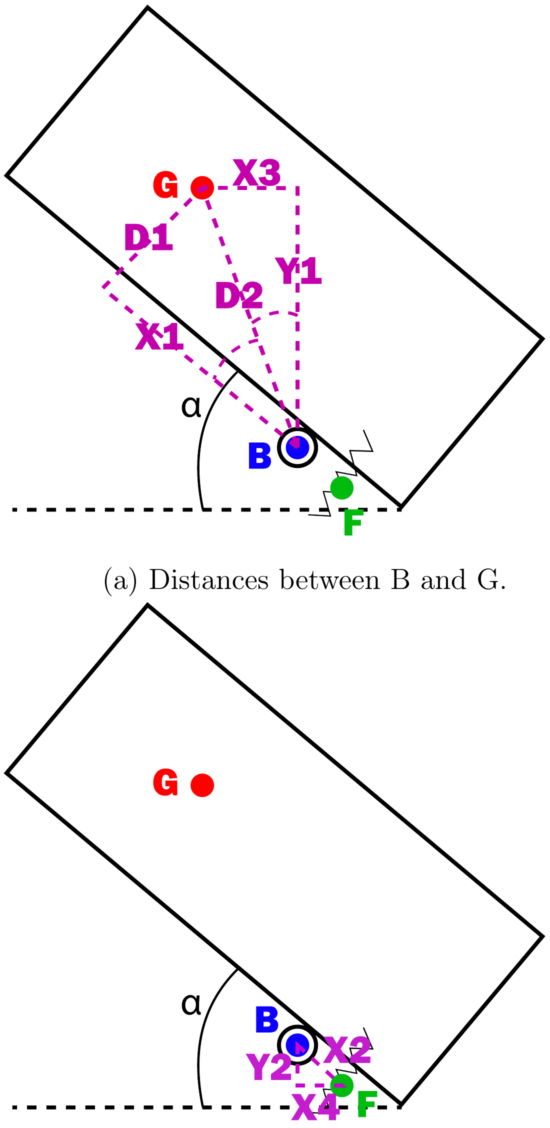


Figure 10.35: The forces affecting the elevation base when the Effector is fully elevated.

To identify these loads, as much information as possible is needed.
From the first FEA, G is known to be 50.08 N. The simplest way to find more of the forces is to summarize the torque in point B. In order to do this, some distances are required. These distances are illustrated in fig. 10.36.



(a) Distances between B and G.

(b) The distances between B and F.

Figure 10.36: Distances between B, G and F illustrated.

In fig. 10.36a, D_1, D_2 and X_1 is known from SolidWorks measuring tool: $D_1 = 65.00\,\text{mm}$, $D_2 = 118.69\,\text{mm}$ and $x_1 = 99.17\,\text{mm}$. X_3 is the needed distance to calculate the torque from G revolving B.

The angle between D_2 and X_1 is:

$$\theta_1 = \cos^{-1}\left(\frac{99.17\,\text{mm}}{118.69\,\text{mm}}\right) = 33.33°. \tag{10.23}$$

Since the angle the Effector is elevated at is known to be $\alpha = 40°$, the angle between D_2 and Y_1 can be found as:

$$\theta_2 = 90° - 40° - 33.33° = 16.67°. \tag{10.24}$$

Knowing this angle, D_2, X_3 is found:

$$\sin\theta_2 = \frac{x_3}{D_2} \to x_3 = D_2 \sin\theta_2 = 118.69\,\text{mm} \cdot \sin 16.67^\circ = 34.0\,\text{mm}. \tag{10.25}$$

In fig. 10.36b, X_2 is known from SolidWorks measuring tool. The angle between X_4 and X_2 is known to be 40° coming from the elevation angle and since this is a right- angled triangle, the angle between X_4 and Y_2 is 90°. From this information, Y_2 and X_4 can be found. These are the needed distances to calculate the torque from force F revolving point B.

$$x_2 = 15.50\,\text{mm}$$

$$\cos\theta_3 = \frac{x_4}{x_2} \to x_4 = x_2 \cos\theta_3 = 15.50\,\text{mm} \cdot \cos 40^\circ = 11.87\,\text{mm} \tag{10.26}$$

$$\tan\theta_3 = \frac{y_2}{x_4} \to y_2 = x_4 \tan\theta_3 = 11.87\,\text{mm} \cdot \tan 40^\circ = 9.96\,\text{mm} \tag{10.27}$$

After finding all the required distances, the identifying of forces can begin:

$$\sum M_B = 0 \to Gx_3 - F_y x_4 - F_x Y_2 = 0 \tag{10.28}$$

From decomposing F, an expression for F_x and F_y is found:

$$F_x = F \sin\theta_3 \tag{10.29}$$

$$F_y = F \cos\theta_3 \tag{10.30}$$

Substituting F_x and F_y in (10.28) with (10.29) and (10.30) gives:

$$\sum M_B = 0 \to Gx_3 - F\cos\theta_3 x_4 - F\sin\theta_3 y_2 = 0 \tag{10.31}$$

Sorting (10.31) gives:

$$F(\cos\theta_3 x_4 + \sin\theta_3 y_2) = Gx_3 \tag{10.32}$$

Solving for F in (10.32) results in:

$$F = \frac{Gx_3}{(\cos\theta_3 x_4 + \sin\theta_3 y_2)} = \frac{50.08\,\text{N} \cdot 34.0\,\text{mm}}{(11.87\,\text{mm})\cos 40^\circ + (9.96\,\text{mm})\sin 40^\circ} = 109.8\,\text{N}. \tag{10.33}$$

Knowing the value for F found using (10.33), (10.29) and (10.30) can be used to determine the values of F_x and F_y:

$$F_x = F\sin\theta_3 = 109.8\,\text{N} \sin 40^\circ = 70.58\,\text{N} \tag{10.34}$$

and

$$F_y = F\cos\theta_3 = 109.8\,\text{N} \cos 40^\circ = 84.14\,\text{N}. \tag{10.35}$$

To find B, the summation of forces in y-direction is used:

$$\sum F_y = 0 \rightarrow B - G - F_y = 0 \rightarrow B = G + F_y = 50.08\,\mathrm{N} + 84.14\,\mathrm{N} = 134.22\,\mathrm{N}. \tag{10.36}$$

To find A, the summation of forces in x-direction is used:

$$\sum F_x = 0 \rightarrow A - F_x = 0 \rightarrow A = F_x = 70.58\,\mathrm{N} \tag{10.37}$$

Now all forces affecting the elevation base are known and placed. G, F_y and F_x will each create a torque revolving the axle at point B, while A and B work directly on the axle.

$$M_G = Gx_3 = 50.08\,\mathrm{N} \cdot 0.034\,\mathrm{m} = 1.70\,\mathrm{Nm} \tag{10.38}$$

$$M_Fy = -F_yx_4 = -84.14\,\mathrm{N} \cdot 0.01187\,\mathrm{m} = -1.00\,\mathrm{Nm} \tag{10.39}$$

$$M_Fx = -F_xy_2 = -70.58\,\mathrm{N} \cdot 0.00996\,\mathrm{m} = -0.70\,\mathrm{Nm} \tag{10.40}$$

$$A = 70.58\,\mathrm{N} \tag{10.41}$$

$$B = 134.22\,\mathrm{N} \tag{10.42}$$

Using the same safety factor as in the first analysis, $n = 2.5$, this is the final values for alternative 1:

$$M_Gn = Gx_3n = 50.08\,\mathrm{N} \cdot 0.034\,\mathrm{m} \cdot 2.5 = 4.25\,\mathrm{Nm} \tag{10.43}$$

$$M_Fyn = -F_yx_4 = -84.14\,\mathrm{N} \cdot 0.01187\,\mathrm{m} \cdot 2.5 = -2.50\,\mathrm{Nm} \tag{10.44}$$

$$M_Fxn = -F_xy_2 = -70.58\,\mathrm{N} \cdot 0.00996\,\mathrm{m} \cdot 2.5 = -1.75\,\mathrm{Nm} \tag{10.45}$$

$$A_n = 70.58\,\mathrm{N} \cdot 2.5 = 176.45\,\mathrm{N} \tag{10.46}$$

$$B_n = 134.22\,\mathrm{N} \cdot 2.5 = 335.55\,\mathrm{N}. \tag{10.47}$$

The placing of the loads in the FEA is the same as in the first analysis. The only difference is that in this analysis, force A is placed inside the axle-holes pointing in the positive x-direction. Regarding the mesh, the exact same settings are used.

The result from running the analysis on alternative 1 is seen in fig. 10.37. The maximum tension discovered in the part is 25.8 MPa, which is far less than the materials yield strength at 110 MPa.

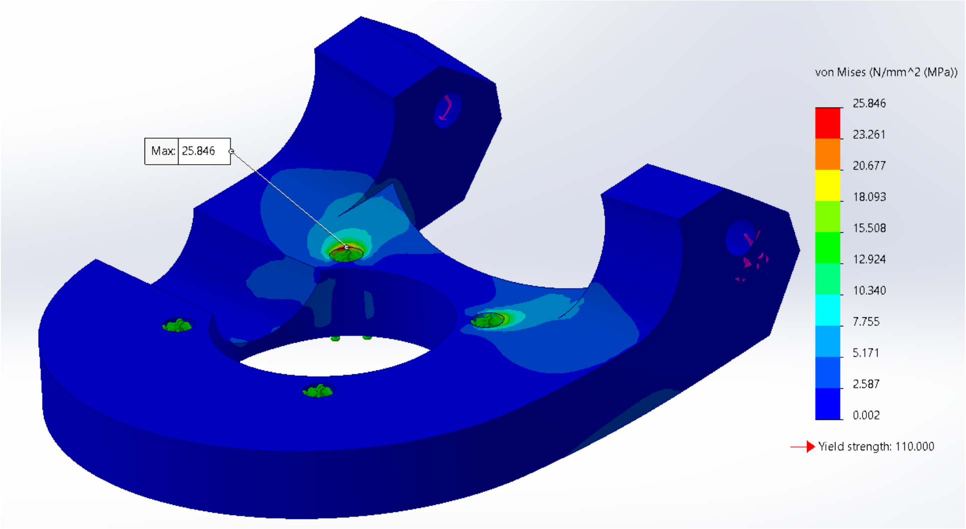


Figure 10.37: The elevation base after the analysis is run.

This analysis was also run on the original design and the result can be seen in fig. 10.38.

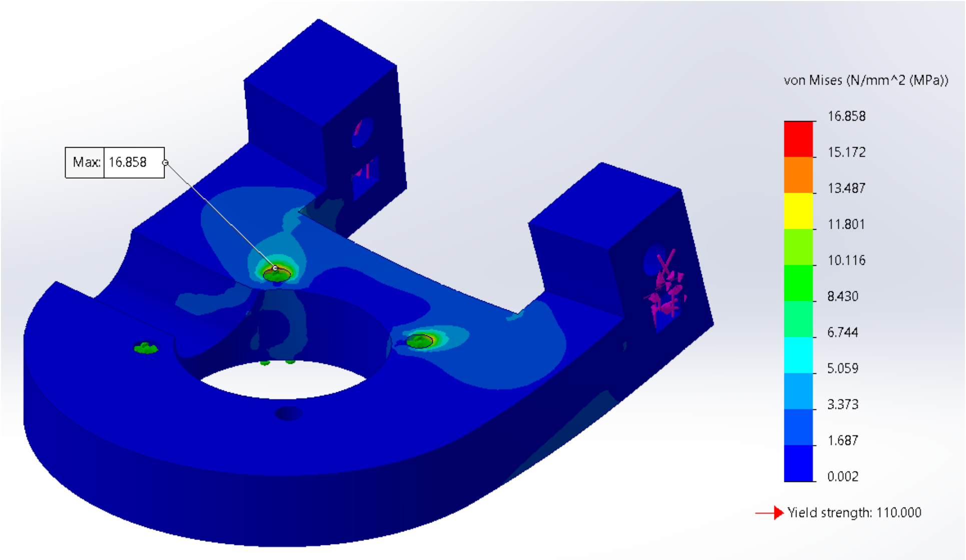


Figure 10.38: The original elevation base after the analysis is run.

As seen here, the maximum tension in the old design is 168 MPA. This is again less than in the new design. This is because of the reduction in material done when redesigning. Since the original part was designed for 3D-printing, the security of extra material weighed more than the reduction in mass. When redesigning, the reduction in mass has been more important.

As in the first analysis, alternative 2C uses another safety factor than alternative 1. When applying the safety factor of $n = 1.4$ to the given loads, the values used in the FEA is as follows:

$$M_G n = G x_3 n = 50.08\,\mathrm{N} \cdot 0.034\,\mathrm{m} \cdot 1.4 = 2.38\,\mathrm{Nm} \tag{10.48}$$

$$M_F yn = -F_y x_4 = -84.14\,\mathrm{N} \cdot 0.01187\,\mathrm{m} \cdot 1.4 = -1.4\,\mathrm{Nm} \tag{10.49}$$

$$M_F xn = -F_x y_2 = -70.58\,\mathrm{N} \cdot 0.00996\,\mathrm{m} \cdot 1.4 = -0.98\,\mathrm{Nm} \tag{10.50}$$

$$A_n = 70.58\,\mathrm{N} \cdot 1.4 = 98.8\,\mathrm{N} \tag{10.51}$$

$$B_n = 134.22\,\mathrm{N} \cdot 1.4 = 187.9\,\mathrm{N} \tag{10.52}$$

Again, the same mesh settings are used and the loads are placed on the same locations as before.

In fig. 10.39, the result from the analysis is shown. The maximum tension occurring in the part is 14.09 MPa, which is significantly less than the yield strength of Nylon 12 from appendix Y at 48.94 MPa.

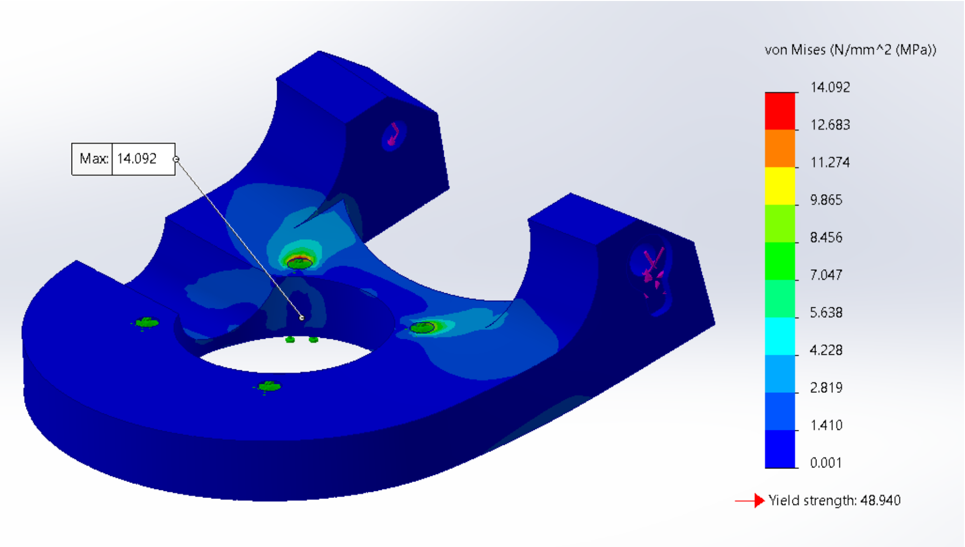


Figure 10.39: The result of the FEA of alternative 2C.

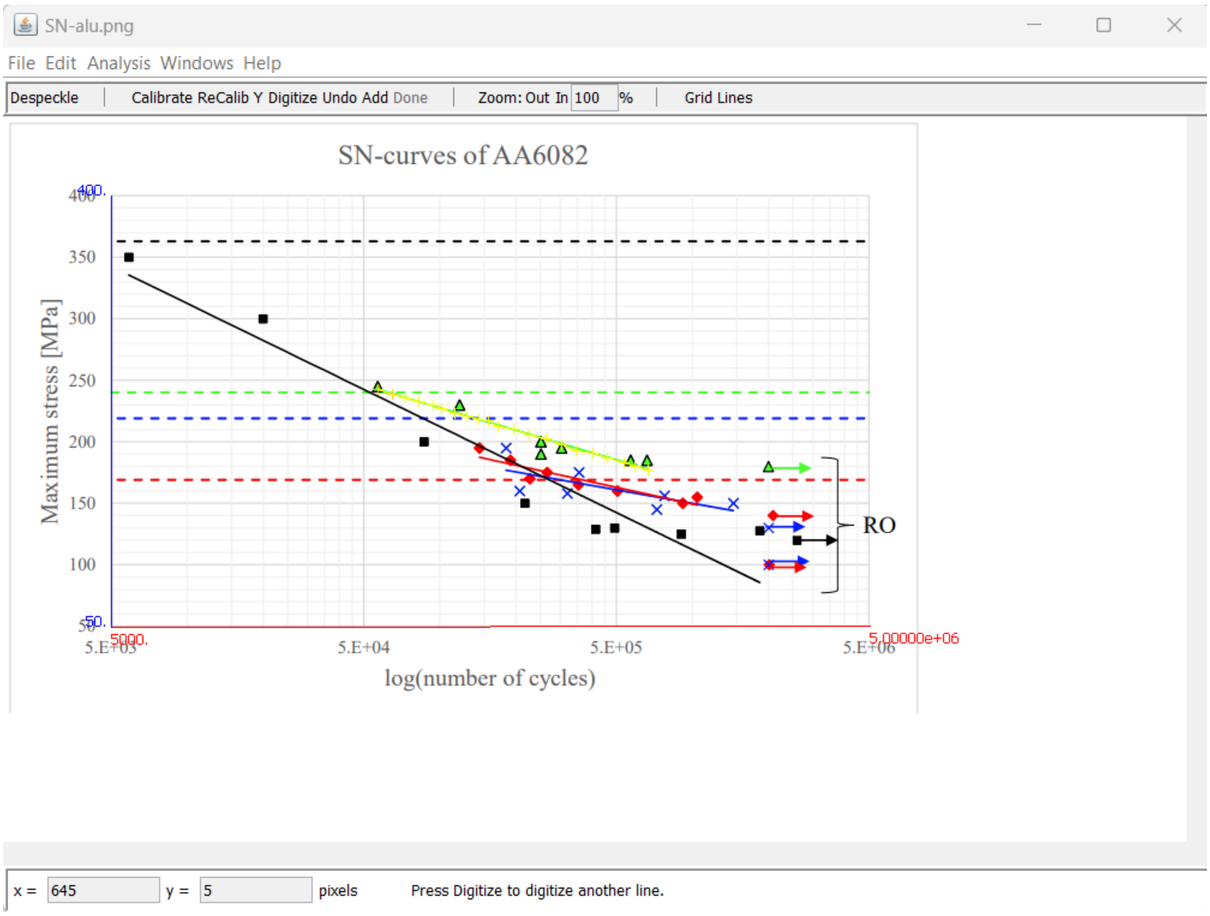
The results from these analysis shows that both design alternative 1 and 2C will withstand the loads exposed to when the Effector is fully elevated.

Static conclusion: After running analysis and figuring out how the loads of the Effector will affect the elevation base, it is clear that both alternative 1 and 2C is designed to withstand these loads by a good margin. The static load will in other words not be a problem for either designs.

The third analysis: Fatigue

Since the Effectors' position will vary between elevated and not elevated multiple times, it is relevant to see if fatigue can be an issue. Fatigue is defined as the creation of cracks in a material exposed to variable tension during a time period. The maximum tension is less than the yield strength of the material [111]. This analysis is, as mentioned, only run on alternative 1.

The material chosen is aluminum 6082-T4. This material has, unlike for example steel or titanium, no fatigue limit. This is where the SN-curve (curve presenting the fatigue data, S is stress and N is number of cycles until failure) flattens out horizontally, illustrating that the metal can survive at this limited stress for an infinite number of cycles. In aluminums case, this does not happen, and the curve gradually decreases until a low-stress fatigue failure [112].
When running this analysis in SolidWorks, the program requires some information about the material and the part. The Fatigue analysis is based on the results from the two static analysis run before. In addition to this, the materials SN-curve, or rather the data points creating the curve, are also needed. Since the chosen material, aluminum 6082-T4, is not a standard SolidWorks material, this data has to be constructed. This can either be done by testing the material and getting the raw data or using a computer program to transfer an existing curve from someone else's research into data points. Because of the time schedule, the second option was chosen. Marte Brynjulfsens SN-curve [112] was converted to data points using the program "Plot Digitizer" [5]. This program allows the user to import a picture of the preferred curve and enter information about the x and y axis before tracing the curve. The result of this trace is saved as data points. Fig. 10.5 illustrates the curve and the information generated. These data points are then added to SolidWorks in the materials SN-curve data.

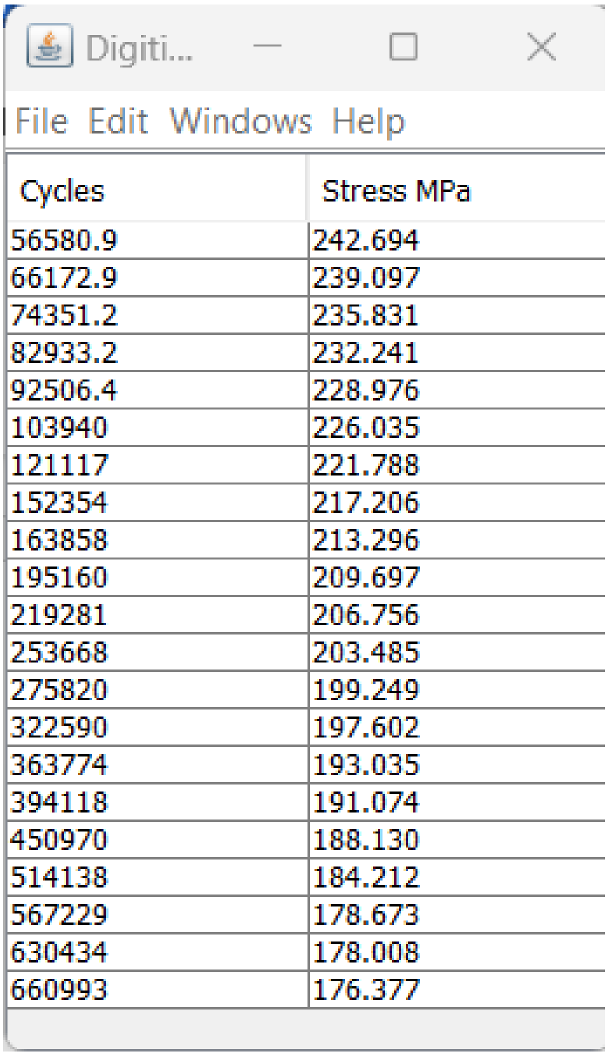


x = 645 y = 5 pixels

Press Digitize to digitize another line.

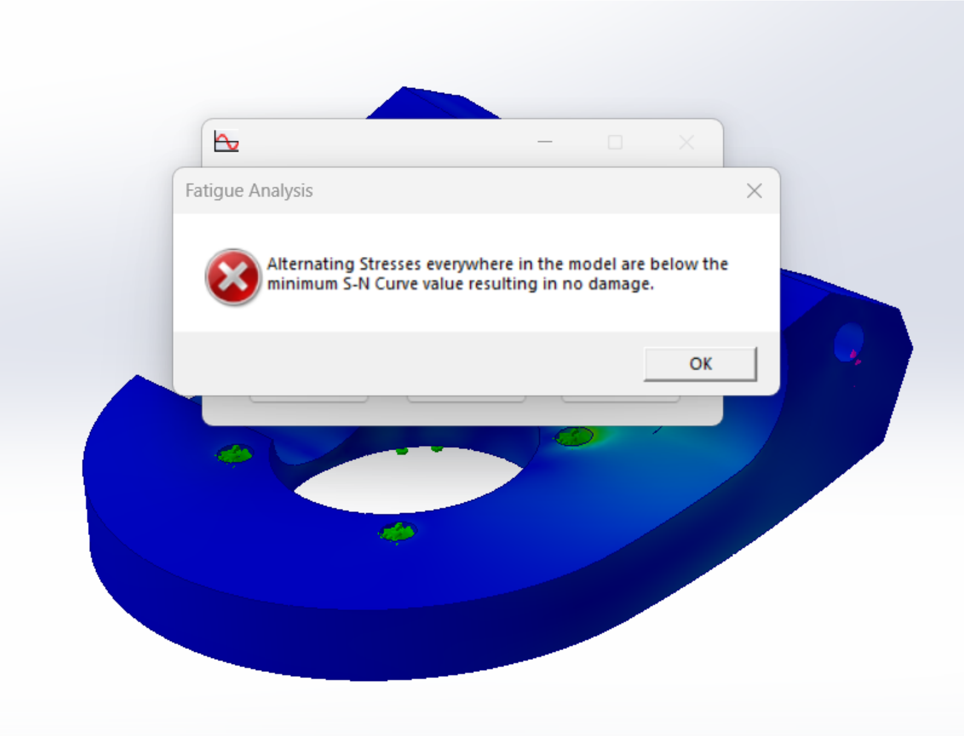
Figure 10.40: "Plot Digitizer" [5] used to convert graphs to data points.

Table 10.5: The data points from the SN-curve.



Cycles	Stress MPa
56580.9	242.694
66172.9	239.097
74351.2	235.831
82933.2	232.241
92506.4	228.976
103940	226.035
121117	221.788
152354	217.206
163858	213.296
195160	209.697
219281	206.756
253668	203.485
275820	199.249
322590	197.602
363774	193.035
394118	191.074
450970	188.130
514138	184.212
567229	178.673
630434	178.008
660993	176.377

The number of cycles run in the analysis was 1000000. The result of the analysis is seen in fig. 10.41. It is clear that with the loads the elevation base is exposed to, fatigue failure will not be an issue.


Fatigue Analysis

Alternating Stresses everywhere in the model are below the minimum S-N Curve value resulting in no damage.

Figure 10.41: The result from the fatigue analysis.

The note reads: Alternating stresses everywhere in the model are below the minimum S-N curve value resulting in no damage.

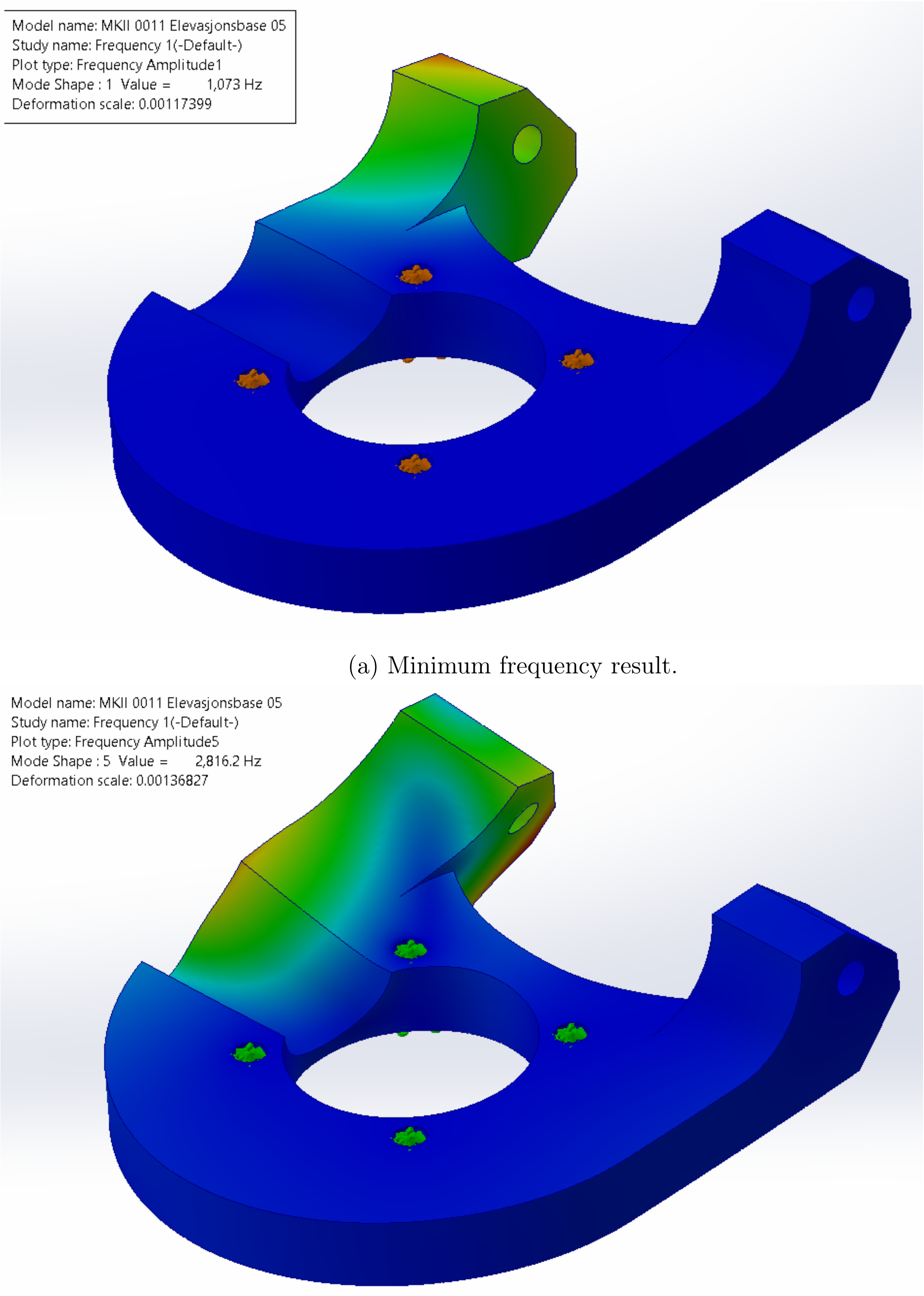
The fourth analysis: Vibration analysis

The vibration analysis of the elevation base was run in order to investigate this parts natural frequencies. The mechanical vibration exited by the Effector when elevating was investigated in chapter 9.6, and finding the natural frequency of the elevation base and comparing the two, would ensure the avoidance of resonance. The results from the test in chapter 9.6 showed that the Effector has a dominant resonance frequency of approx. 500 Hz.

When finding the natural frequency of a model in SolidWorks simulations, the model should be fixed where it will be fixed in reality. This was done the same way as it was done in the static analyses of the same part: using fixed geometry on the four screw holes. No external loads were added when investigating the natural frequency, since the goal was to find the parts natural frequency at rest.

Before running the analysis, the part has to be meshed. The same mesh settings used in every other FEA including the elevation base was also used in this analysis.

The result from the analysis can be seen in fig. 10.42. As seen, the elevation bases natural frequency has a minimum value of 1073 Hz, and a maximum value of 2816 Hz.



Model name: MKII 0011 Elevasjonsbase 05
Study name: Frequency 1(-Default-)
Plot type: Frequency Amplitude1
Mode Shape : 1 Value = 1,073 Hz
Deformation scale: 0.00117399

(a) Minimum frequency result.

Model name: MKII 0011 Elevasjonsbase 05
Study name: Frequency 1(-Default-)
Plot type: Frequency Amplitude5
Mode Shape : 5 Value = 2,816.2 Hz
Deformation scale: 0.00136827

(b) Maximum frequency result.

Figure 10.42: Minimum and maximum frequency.

As mentioned, the Effector has a resonance frequency of 500 Hz when elevating. The octave rule-of-thumb says that the systems natural frequency should nor be between half and twice as large as the applied frequency [113]. In this case, the applied frequency was the mechanical frequency from the Effector when elevating. This means that the elevation bases natural frequency should not be between 250 and 1000 Hz. Since the elevation bases minimum natural frequency was 1073 HZ, the risk of experiencing resonance is low.

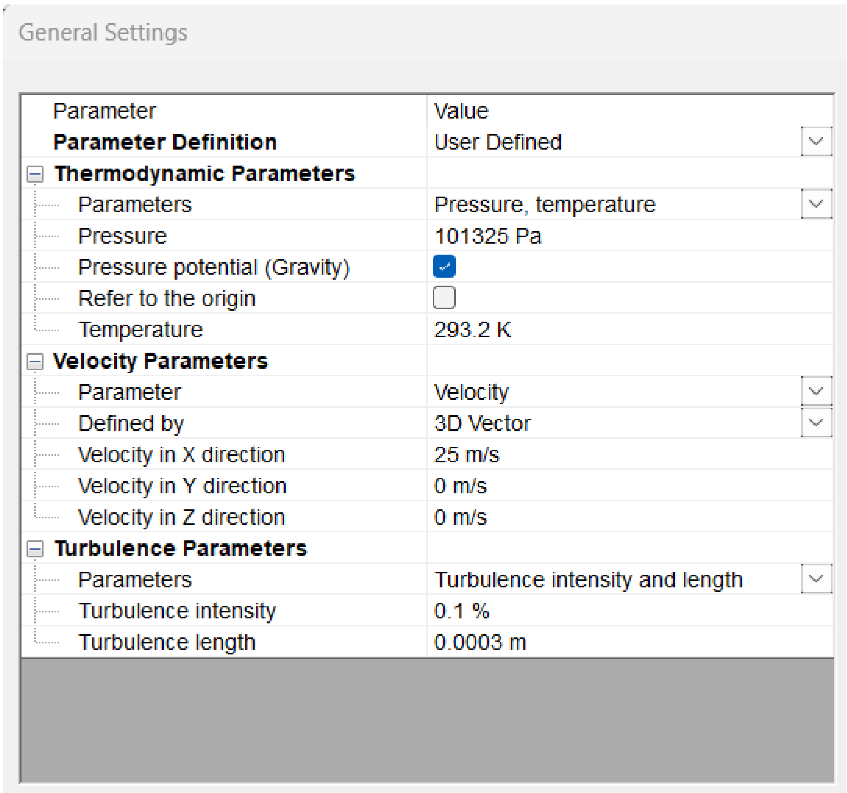
10.4.2 Projectile

KC

Flow analysis

After designing the new projectile, a simple flow analysis was initiated to investigate the lift and drag forces working on the projectile in flight. The analysis was run in SolidWorks Flow Simulation.

When starting a flow simulation, there are several settings that needs to be determined. This includes whether it is an internal or external analysis, what type of fluid the object is submerged in and different thermodynamic, velocity and turbulence parameters. In this analysis, the settings were set to an external analysis where the object is submerged in air. The parameters used can be seen in fig. 10.43.



General Settings

Parameter	Value
Parameter Definition	User Defined
Thermodynamic Parameters	
Parameters	Pressure, temperature
Pressure	101325 Pa
Pressure potential (Gravity)	☑
Refer to the origin	☐
Temperature	293.2 K
Velocity Parameters	
Parameter	Velocity
Defined by	3D Vector
Velocity in X direction	25 m/s
Velocity in Y direction	0 m/s
Velocity in Z direction	0 m/s
Turbulence Parameters	
Parameters	Turbulence intensity and length
Turbulence intensity	0.1 %
Turbulence length	0.0003 m

Figure 10.43: Parameters used when running the flow analysis

As seen in fig. 10.43, the velocity in the X direction was set to 25 m/s. This is because the previous group estimated the projectiles output speed to 23 m/s [1].

The goal of the analysis was to investigate the lift and drag forces occurring on the projectile. Therefore, the analysis global goals were set to find the forces in X and Y direction. The coordinate system used in this analysis is illustrated in fig.

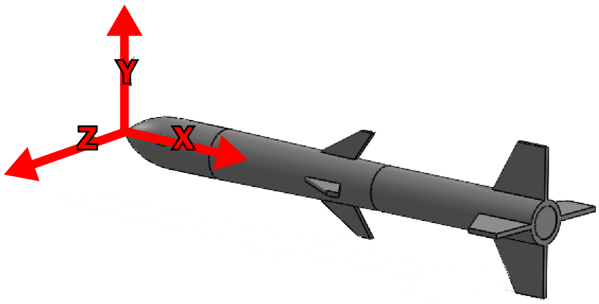


Figure 10.44: The coordinates system used when running the flow analysis.

The force working on the projectile in Y direction represents the lift, while the force working om the projectile in the X direction represents the drag.

Before running the analysis, the computational domain must be defined and meshed. The computational domain is the volume surrounding the model where the flow calculations will happen. In this analysis, the computational domain was meshed with both a global and a local mesh. The local mesh was used to ensure extra accurate calculations closest to the model. The mesh is shown in fig. 10.45. Here, it is possible to see the distinct difference between the global and the local mesh.

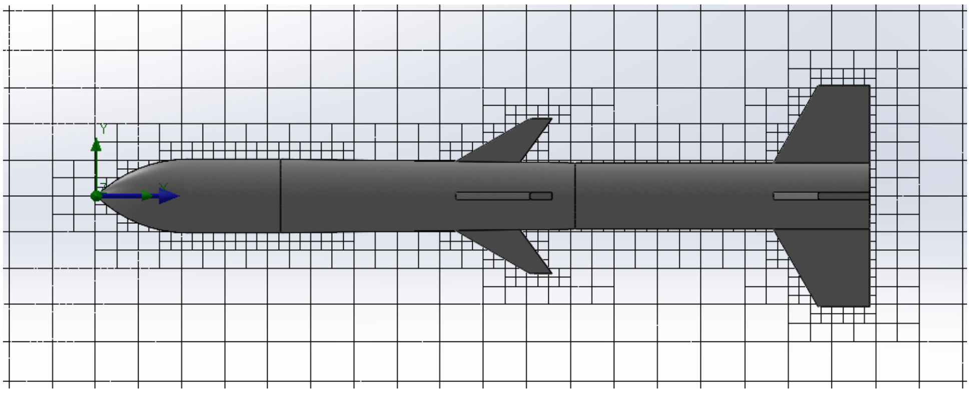
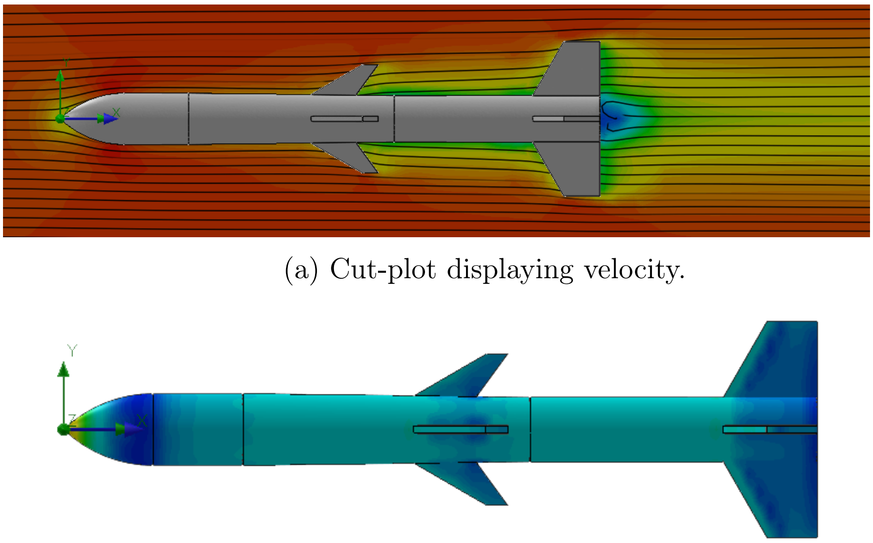


Figure 10.45: The computational domain mesh.

After the settings, goals and mesh was set, the analysis could be run. This was done by simply clicking "Run". When checking the results, two types of plot were the most interesting. These can be seen in fig. 10.46.



(a) Cut-plot displaying velocity.

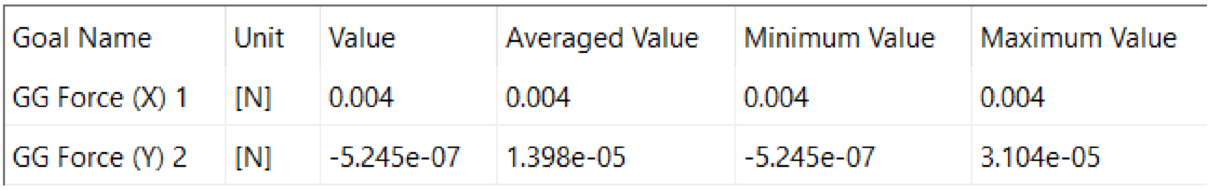
(b) Surface plot displaying pressure.

Figure 10.46: Cut-plot and surface plot.

Fig. 10.46a shows a cut-plot displaying velocity. In this plot, the formation of some turbulence is seen at the rear end of the projectile. This is as expected. Fig. 10.46b shows a surface plot displaying pressure on the projectiles surface. This plot shows that the projectiles nose cone is exposed to an elevated pressure compared to the rest of the projectile body. This is also as expected.

The goal with this analysis was to investigate the lift and drag forces, where the force in X direction represented drag and the force in Y direction represented lift. Fig. 10.6 shows the plotted goal results.

Table 10.6: The plotted goal result.



Goal Name	Unit	Value	Averaged Value	Minimum Value	Maximum Value
GG Force (X) 1	[N]	0.004	0.004	0.004	0.004
GG Force (Y) 2	[N]	-5.245e-07	1.398e-05	-5.245e-07	3.104e-05

From table 10.6, the lift and drag was identified as:

- The lift force: average value of 0.00001398 N.
- The drag force: average value of 0.004 N.

10.5 Comparing new and old accuracy

10.5.1 Design of measuring equipment

KC

Azimuth

In order to reach the stakeholder requirement FR-04 with the associated system requirements SR-04-02 and SR-04-03, the Effector has to be precise when elevating and rotating into position. Before deciding whether these features need improvement or not, the existing system was tested.

There are several ways to conduct tests like these, both mechanical and electrical. Some methods are expensive and advanced, but for this project the test method was kept as simple as possible.

SR-04-02 is a system requirement stating that the Effectors azimuth needs to have an accuracy within ±1.0°. To check if the existing system is accurate within this limit, a simple mechanical solution was chosen:

The idea when testing the azimuth is simple. It consists of a simple plate engraved with the necessary degrees and an arrow to pinpoint what degree the Effector is located at. This value is compared to the value set in the GUI to see how far off the Effector is. The plate and the arrow was designed in SolidWorks and produced using USNs Epilog Fusion M2 32 laser cutter. The material used, was 3.0 mm thick plywood.

The initial idea for the measuring-plate, was to make a large plate covering at least 180°. This would be fastened by a hole fitting the rotational base of the Effector. After some design-attempts, it became clear that this many degrees were not necessary, and the measuring-plate could advantageously be designed smaller.

The first design draft, shown in fig. 10.47, resembles the initial idea, but is only designed to cover 50°: ±25° with zero on the middle of the plate. As seen, there is a hole to fit the rotational base to place the measuring-plate. Also, there are two clearance holes fitting existing holes at the Effector to keep the plate still and in place.

When using these holes, the fitting to the rotational base is no longer necessary. Because of this, the solution was redesigned.

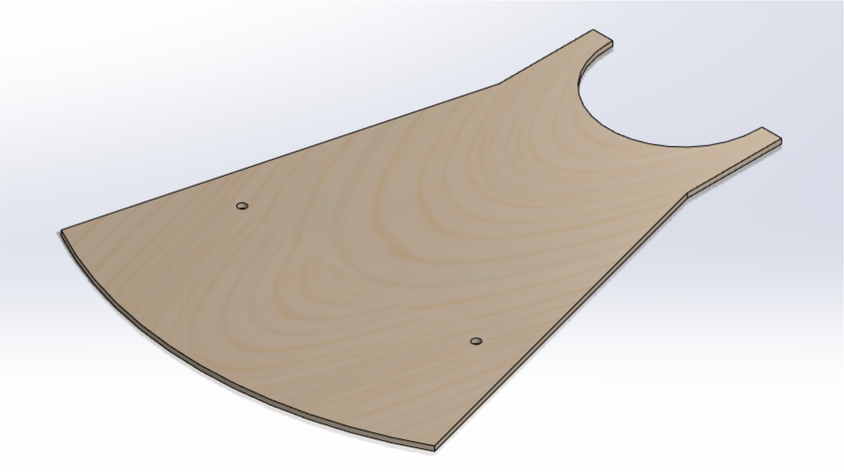


Figure 10.47: First design draft.

The second design, as seen in fig.10.48, is a lot like the first draft. The only difference is that here the clearance-holes is the only thing keeping the measuring-plate in place. Since this is enough for a steady placement, the extra material used in the first design is unnecessary.

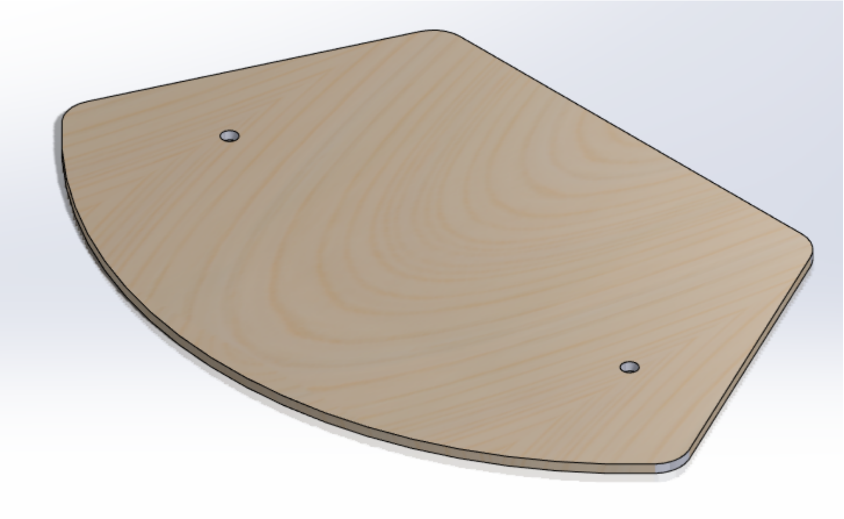


Figure 10.48: Second design draft.

It is important to not use more material than needed. With this in mind, the material used in the two designs are compared:

- The first design has an area of A ≈ 98000 mm^2
- The second design has an area of A ≈ 54000 mm^2

This means that by proceeding with the second option, approx. 44000 mm^2 of material was saved. This means that the used material decreased with approx. 45%. The values describing the area is found in SolidWorks. Since SR-04-02 requires the Effectors azimuth to be accurate within ±1.0°, it is important to have this level of accuracy on the measuring-plate. This means that, engraved in the plywood, every whole and half degree must be shown. To get this right, it is important to locate and use the axis in which the Effector rotates around. As seen in fig. 10.49, this axis is located in the middle of the rotational base (black line). This means that when drawing the angles, they must use this axis as their origin.

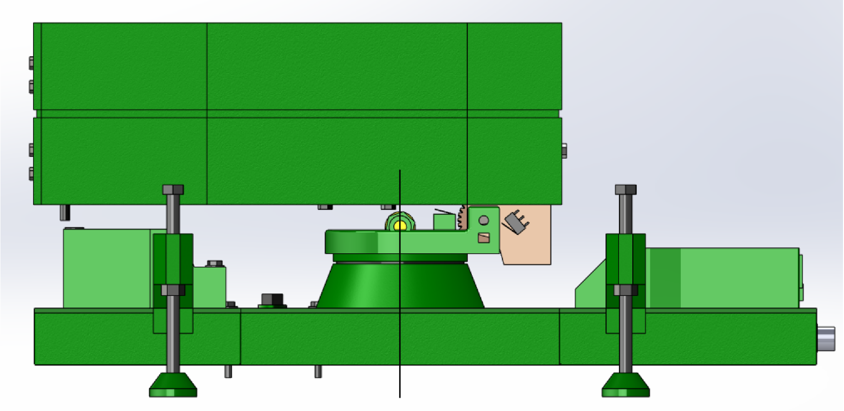


Figure 10.49: Azimuth origin.

The result after placing the angles is seen in fig. 10.50. The lines from origin and out represents every fifth degree, while the orange lines represent every half degree. Here it is also shown location of the origin.

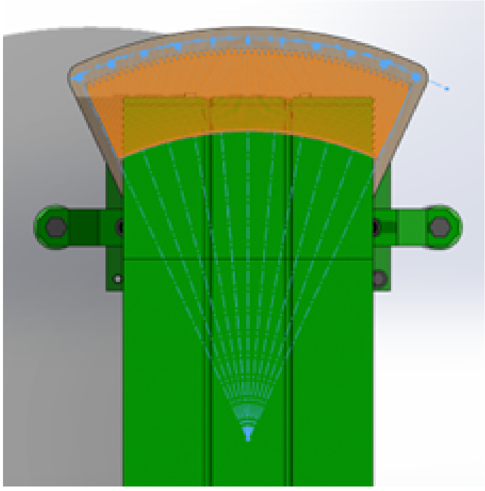


Figure 10.50: Azimuth plate seen from above.

To ensure the accuracy when engraving these lines, it was important for the laser cutter to be able to engrave lines with a narrow line thickness. When using the laser cutter, the FlexiDesigner program was used. In this program, properties like line thickness can be determined. After some researching and testing both the software and the laser cutter, it was found that a line thickness of 0,08 mm was enough.

The final result can be seen in fig. 10.51.

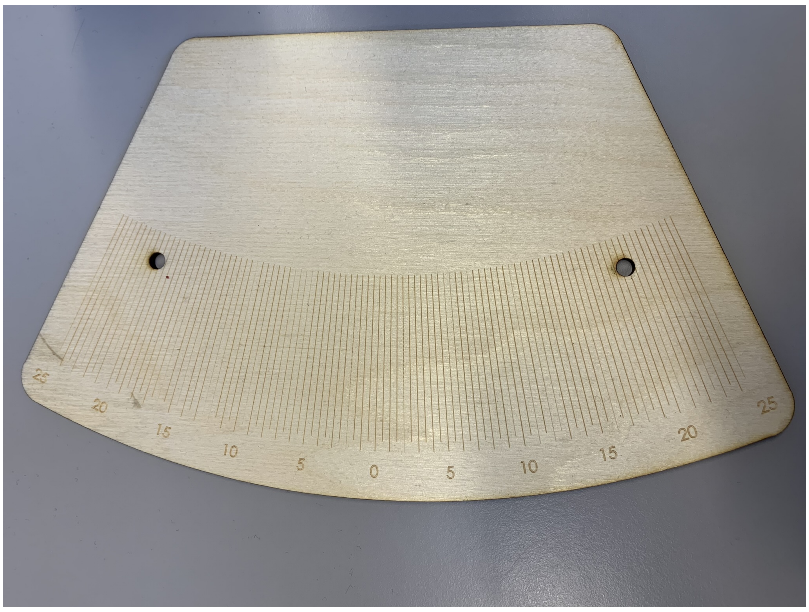


Figure 10.51: The final result.

Elevation

The measuring-plate to check the accuracy of the elevation was made in a similar way. A version standing vertically next to the Effectors top structure with a belonging arrow was designed. This was both designed and produced by the same methods as the measuring-plate for the azimuth.

The difference between these two plates, is the axis the Effector rotates around. In the azimuth direction, the axis is the center of the rotation-base. For the elevation on the other hand, the axis is the axle connecting the top – and bottom structure. Both the design and the origin-axis are seen in fig. 10.52.

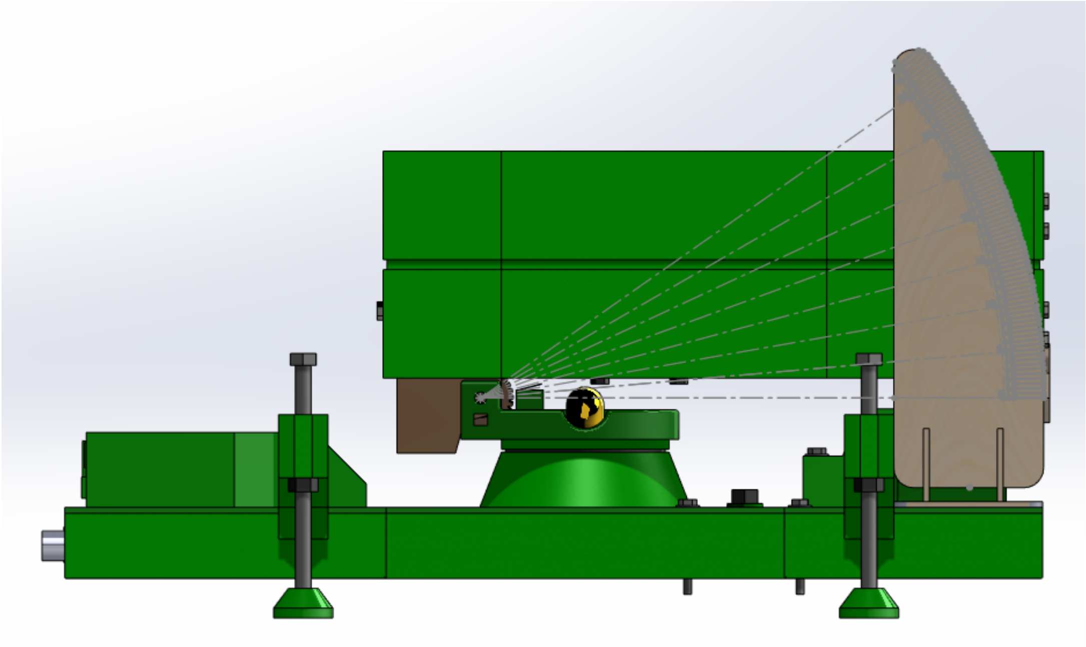


Figure 10.52: Measuring plate for elevation showing origin

This measuring-plate was, like the plate for azimuth, designed to measure accuracy within ±0.5°. This is because the system requirement SR-04-03 requires an accuracy of this value in elevation, just as for azimuth.

Fig. 10.53 shows the final result attached to the Effector ready to conduct accuracy-tests.

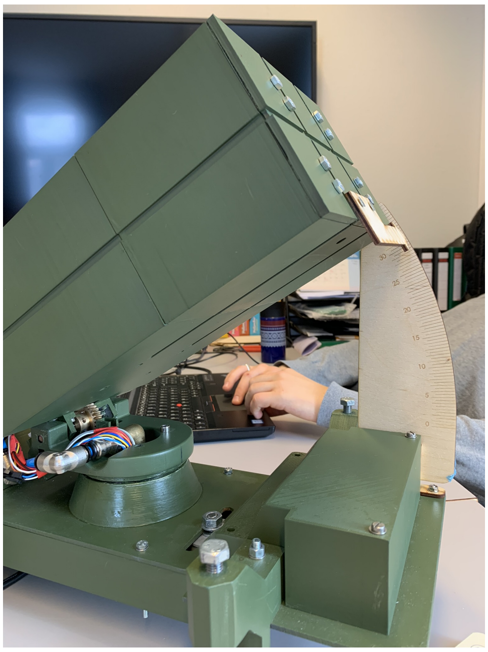
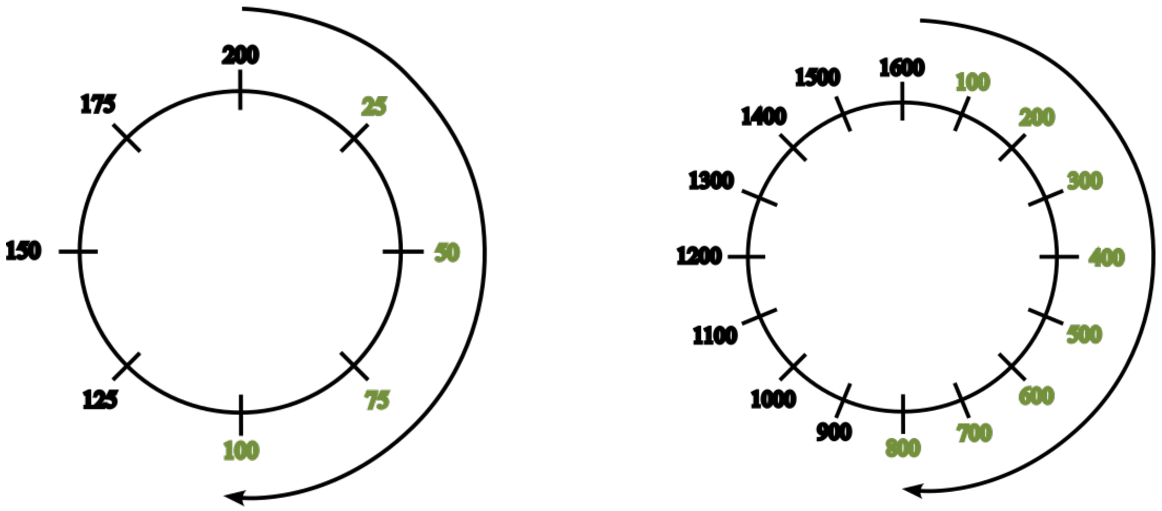


Figure 10.53: Final result of elevation-plate.

10.5.2 Measurement of original azimuth and elevation accuracy



(a) Encoder.

(b) Stepper motor.

Figure 10.54: Discrete steps of a half rotation.

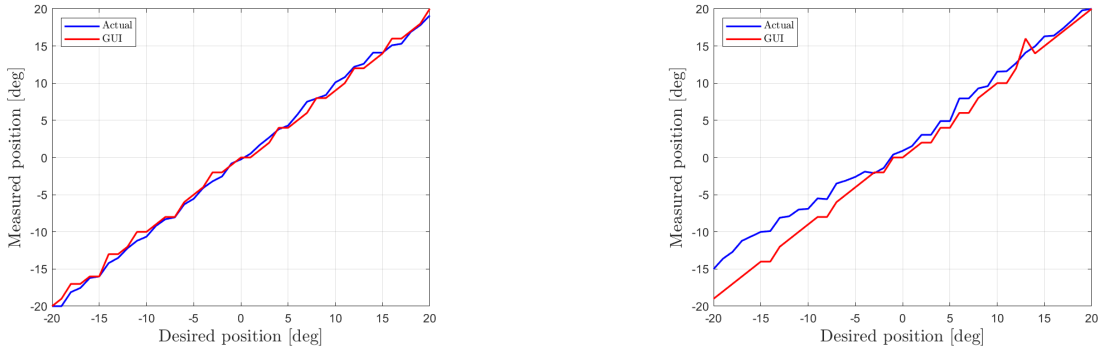
DS

When measuring the state of the Effector with the measuring instruments designed the Effector was driven in both Clockwise (CW) and Counterclockwise (CCW) direction. When the Effector is moving in the CW the value displayed in the GUI decreases, and when the Effector is driven in the CCW the value increases. The following section will go trough the results from the measurements performed on the Effector. This is in order to investigate if the work done by Project Mini Launcher was sufficient when it comes to the position accuracy on both axes. The Effector is moved to desired position along the azimuth and elevation direction from the GUI. The actual position is recorded from the measuring instrument, and compared to the position given by the GUI.

Azimuth

MH

The resulting measurement of the azimuth axis can be seen in fig. 10.55, this shows the hysteresis along the Azimuth rotation in both the CW, seen in fig. 10.55b, and the CCW, seen in fig. 10.55a. The hysteresis along the Azimuth axis shows that the Effector struggles to get to the desired position. The Effectors actual position along with the position that is shown in the GUI, jumps around the desired position. In both the CW and CCW directions the deviation jumps from the negative side to the positive side.

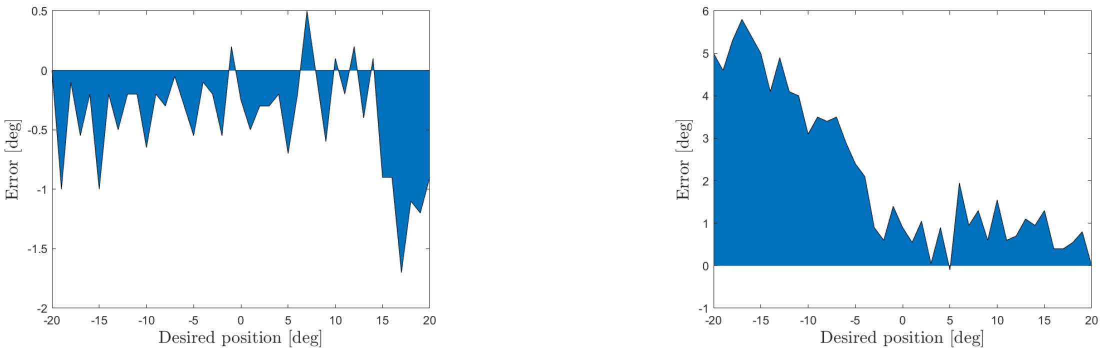


(a) CCW rotation.

(b) CW rotation.

Figure 10.55: Original hysteresis of azimuth.

The error on each particular degree of rotation is shown in fig. 10.56. When moving the Effector in both directions from 20° to −20°, seen in fig. 10.56a and −20° to 20°, shown in fig. 10.56b, it shows that the error is random in the different positions.



(a) CCW rotation error.

(b) CW rotation error.

Figure 10.56: Original error of azimuth.

In Table 10.7 the result from the repeatability test i shown, the purpose of this test is to see whether or not the Effector hits the same position when moving from position A to B. In this case the Effector is moved from the starting position of 0° to 180° and then back to 0°. From the Error row in Table 10.7 it can be seen that the deviation is large.

Table 10.7: Original repeatability of azimuth.

P_1 [deg]	P_2 [deg]	Desired [deg]	GUI [deg]	Error [deg]
0	-1.6	0	0	-1.6
0	-2.3	0	0	-2.3
0	3.9	0	0	3.9
0	-1	0	0	-1
0	-2.25	0	0	-2.25

The results of a different repetability test can be seen in Table 10.8, here the Effector is moved from 15° to −15°. The resulting error is comparable to the results shown in

Table 10.7. The deviation from the desired position is still high even tough the distance traveled is small

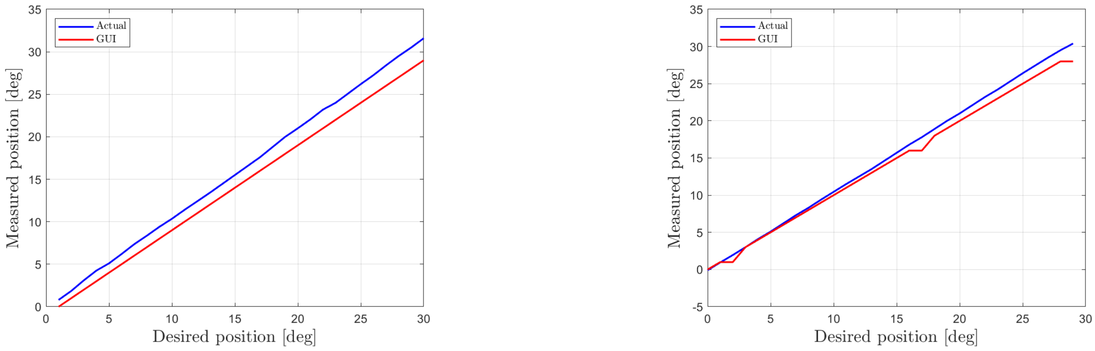
Table 10.8: Original repeatability of azimuth.

P_3 [deg]	P_4 [deg]	Desired [deg]	GUI [deg]	Error [deg]
15	−13.4	−15	−14	1.6
−15	13.8	15	14	−1.2
15	−14	−15	−14	1
−15	13.95	15	14	−1.05
15	−14.7	−15	−14	0.3
−15	13.8	15	14	1.2
15	−15.2	−15	−14	−0.2
−15	13.2	15	14	−1.8
15	−15.55	−15	−14	−0.55
−15	12	15	14	−3

Elevation

MH

This section will go through the resulting measurements recorded when the Effector has been moved to positions along the elevation axis. The hysteresis result is seen in fig. 10.57 and the error deviation on each particular position is shown in fig. 10.58. The repeatability result is shown in Table 10.9.

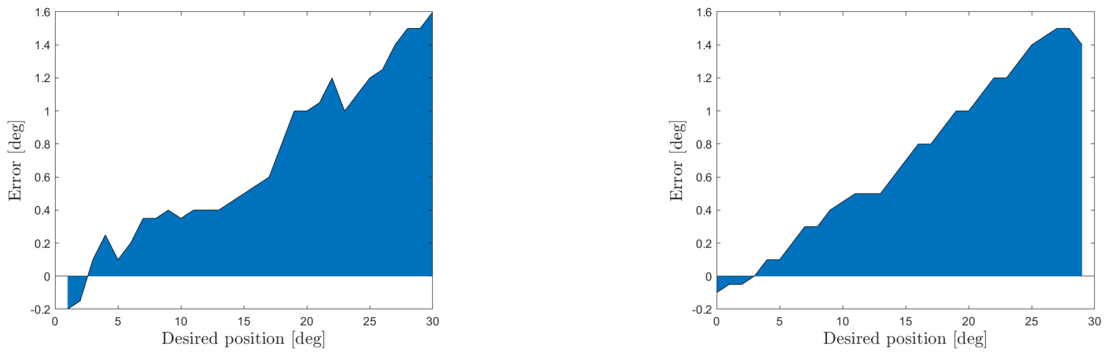


(a) Increase elevation.

(b) Decrease elevation.

Figure 10.57: Original hysteresis of elevation.

The hysteresis along the elevation axis can be seen in fig. 10.57, this shows that the Effectors deviation increases along the travel of the axis, this is due to the fact that the measuring instrument was mounted in the wrong location. This made it so that the measuring instrument was engraved wrong in regards to the position. Even tough this happened the test show that the elevation is somewhat linear, and that it has little random deviation when compared to the azimuth axis. The measurement result when the elevation was increasing is seen in fig. 10.57a, and when the elevation decreases is shown in fig. 10.57b.



(a) Increase elevation.

(b) Decrease elevation.

Figure 10.58: Original error of elevation.

The deviation error on each different position is shown in fig. 10.58, the result when the elevation is increasing is shown in fig. 10.57a. And lastly the result when the Effector is decreasing is shown in fig. 10.58b. Both results shows that the error increases along the axis, this is again due to the misplacement of the measuring instrument.

Table 10.9: Original repeatability of elevation.

P_1 [deg]	P_2 [deg]	Desired [deg]	GUI [deg]	Error [deg]
0	31.5	30	29	1.5
31.5	0.2	0	0	0.2
0.2	31.5	30	29	1.5
31.5	0.1	0	0	0.1
0.1	31.3	30	29	1.3
31.3	0.1	0	0	0.1
0.1	31.3	30	29	1.3
31.3	0.1	0	0	0.1
0.1	31.3	30	29	1.3
31.3	0.2	0	0	0.1

The results of the repeatability test is shown in Table 10.9, here the Effector is driven from the 0° to 30°, and back down to 0°. The results shows that the elevation axis has an acceptable repeatability.

10.5.3 Measurement of improved azimuth and elevation accuracy

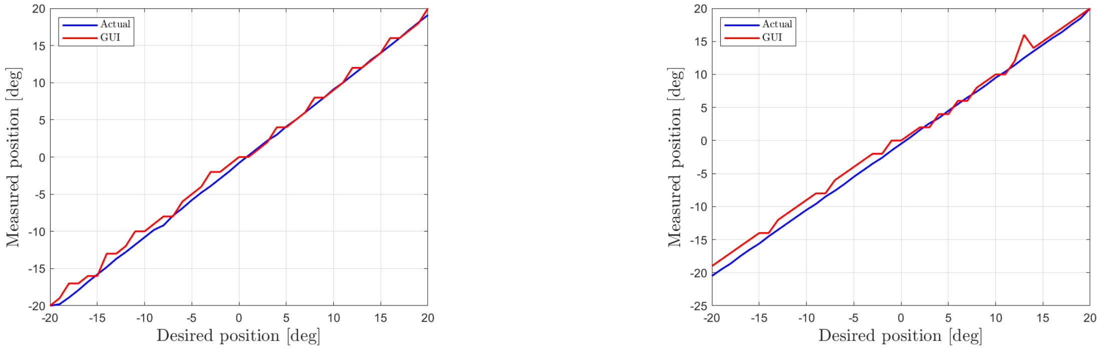
MH

To document that the changes done on the Effector had an impact, a new measurement of the accuracy of the azimuth an elevation axis was performed. The test is performed in the same way as the previous test. A major difference when this measurement was performed compared to the last time is that the encoder on the azimuth axis is defect, and as a result it is removed from the code. The Effector runs on steps given to the stepper motor without feedback, the encoder is still connected to the system and therefore the GUI still updates the position.

Azimuth

KC|MH

The resulting measurement of the azimuth axis can be seen in fig. 10.59, this shows the hysteresis along the Azimuth rotation in both the CW, seen in fig. 10.59b, and the CCW, seen in fig. 10.59a. The hysteresis along the Azimuth axis shows that the Effector moves to the desired position with little to no deviation. The red line which shows the position that is seen in the GUI deviates from the true position, this is because the encoder is defect.

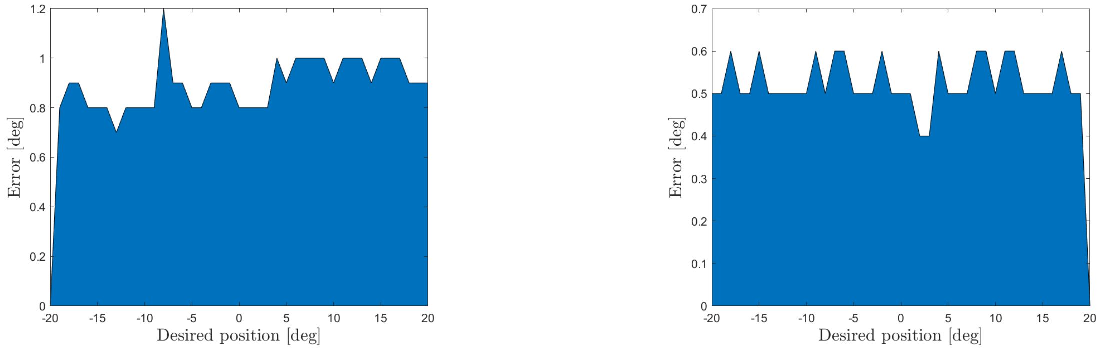


(a) CCW rotation.

(b) CW rotation.

Figure 10.59: Improved hysteresis of azimuth.

The error on each particular degree of rotation is shown in fig. 10.60. When moving the Effector in both directions from 20° to −20°, seen in fig. 10.60a and −20° to 20°, shown in fig. 10.60b, it is seen that the error from the desired position is much less random. The error is centered around 0.8°, this is what is expected of a system with open loop regulation. The bearing install has given the Effector a greater accuracy when compared to before, even without a functioning encoder, and it can now be adjusted in to have minimal error by implementing a PID controller into the Arduino code.



(a) CCW rotation error.

(b) CW rotation error.

Figure 10.60: Improved error of azimuth.

In Table 10.7 the result from the repeatability test i shown, the purpose of this test is to see whether or not the Effector hits the same position when moving from position A to B. In this case the Effector is moved from the starting position of 0° to 180° and then back to 0°. From the Error row in Table 10.7 it can be seen that the deviation is smaller when compared the the original state

Table 10.10: Improved repeatability of azimuth.

P_1 [deg]	P_2 [deg]	Desired [deg]	GUI [deg]	Error [deg]
-1.45	-0.4	-1.4	0	-1
-1.5	-0.5	-1.5	0	-1
-1.4	-0.5	-1.4	0	-0.9
-1.45	-0.45	-1.45	0	-1
-0.4	-0.7	-0.4	0	0.3

The results of a different repetability test can be seen in Table 10.8. Here the Effector is moved from 15° to $-15°$. The resulting error is comparable to the results shown in Table 10.7. The deviation from the desired position is still high even tough the distance traveled is small. Despite the deviation in the desired position, the repeatability is significantly better: varying with only a total of 0.8° from 15° to $-15°$.

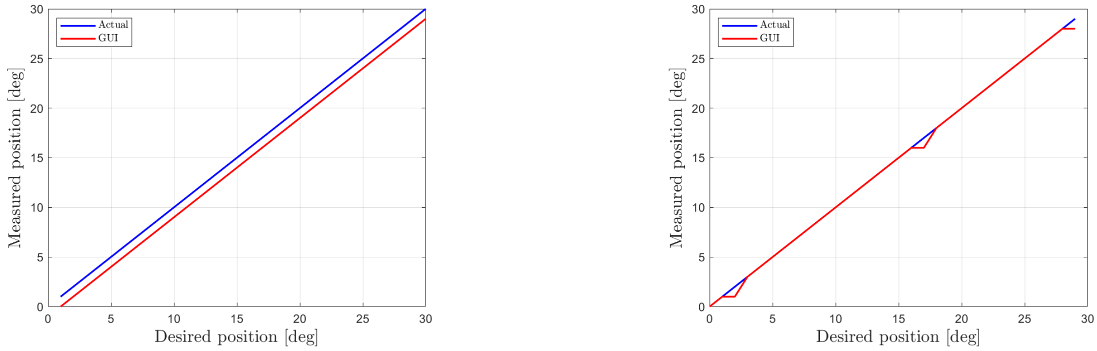
Table 10.11: Improved repeatability of azimuth.

P_3 [deg]	P_4 [deg]	Desired [deg]	GUI [deg]	Error [deg]
14.9	-15.2	-15	-14	-0.2
-15.2	13.8	15	14	-1.2
13.8	-15.5	-15	-14	-0.5
-15.5	13.55	15	14	-1.45
13.55	-15.6	-15	-14	-0.6
-15.6	13.5	15	14	-1.5
13.5	-15.6	-15	-14	-0.6
-15.6	13.7	15	14	-1.6
13.4	-16	-15	-14	-1
-16	13.2	15	14	-1.8

Elevation

KC|MH

The resulting measurements of the elevation axis can be seen in fig. 10.61. This shows the hysteresis in elevation when both increasing, seen in fig. 10.61a, and decreasing, seen in fig. 10.61b the elevation angle. This shows approx. none deviation when the Effector moves to the desired position. The red line represents the position as seen in the GUI, the blue line represents the measured value.

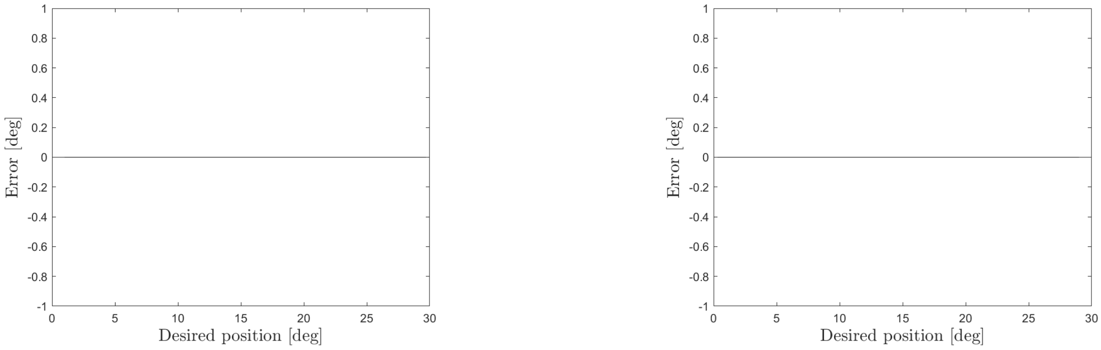


(a) Increase elevation.

(b) Decrease elevation.

Figure 10.61: Original hysteresis of elevation.

The hysteresis along the elevation axis can be seen in fig. 10.62. This shows that the elevation has no deviation from the desired value. Fig. 10.62a shows the deviation when increasing the Effectors elevation, fig. 10.62b shows the deviation when decreasing the Effectors elevation.



(a) Increase elevation.

(b) Decrease elevation.

Figure 10.62: Error of elevation.

The results of the repeatability test is shown in Table 10.12, here the Effector is driven from the 0° to 30°, and back down to 0°. The results shows that the elevation axis has a satisfactory repeatability. Even though the measuring of the desired position shows a deviation of maximum 1.5°, the repeatability shows a deviation of max 0.2°. This is well within the system requirement SR-04-03, saying that the Effector shall have elevation accuracy within ±1.0°.

Table 10.12: Improved repeatability of elevation.

P_1 [deg]	P_2 [deg]	Desired [deg]	GUI [deg]	Error [deg]
0	31.5	30	29	1.5
31.5	0.2	0	0	0.2
0.2	31.5	30	29	1.5
31.5	0.1	0	0	0.1
0.1	31.3	30	29	1.3
31.3	0.1	0	0	0.1
0.1	31.3	30	29	1.3
31.3	0.1	0	0	0.1
0.1	31.3	30	29	1.3
31.3	0.2	0	0	0.1

10.6 Summary

KC

At the start of this chapter, several issues regarding the mechanical state of the Effector was discussed. To refresh the memory, section 10.2 can be re-read. The main problems listed in this chapter was:

- The azimuth not being satisfactory accurate.
- Troubles regarding the belt drive, including 3D-printed gears, Encoder placing, backlash and poor tolerances on the tension wheel.
- The elevation mechanism not functioning properly.
- The projectiles fracturing easily.

Through the sections in this chapter, solutions to these problems has been introduced. This has resulted in new designs and parts increasing the Effectors quality. Both the azimuth and elevation is now satisfactory accurate according to system requirements SR-04-02 and SR-04-03 and the elevation is reliable. The projectile has gotten a redesign and is now no longer printed in an FDM-printer, eliminating the trouble regarding adhesion between the print layers. The belt drive has been upgraded with a new, more stable tension wheel and new 3D-printed gears. Despite this, the belt drive mechanism is still in need of alterations.

Fig. 10.63 illustrates the total system including the Effector, Sensor, the enclosure and the PS.

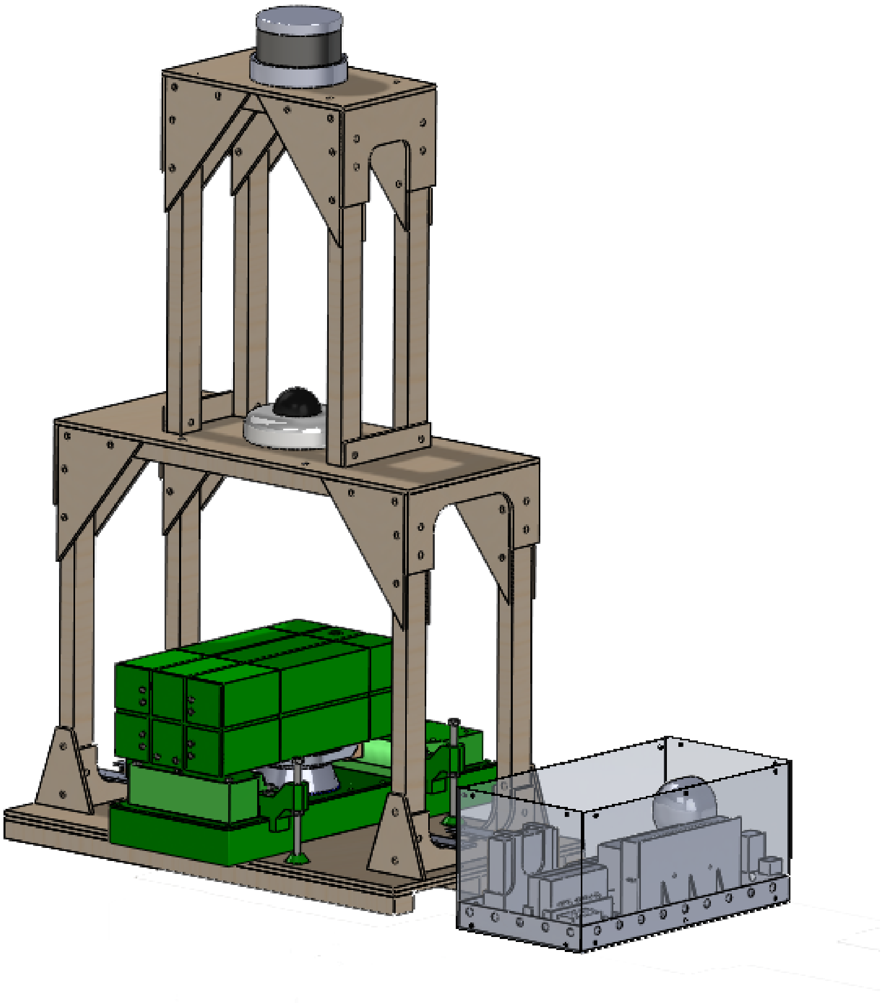


Figure 10.63: Illustration of the total system.

All in all, the mechanical state of the Effector has improved during the project.

Chapter 11

Evaluation

SH

This chapter presents evaluations and challenges that have been encountered throughout the project. Various aspects are taken into consideration, such as administration through the project management, areas for improvement and final remarks through the conclusion. This evaluation aims to provide an assessment of the project, shedding light on both its strengths and areas that warrant further attention.

11.1 Administration

SH

11.1.1 Teamwork

The teamwork throughout the project was exemplary. It can be characterized with a strong sense of effective collaboration and willingness to help each other out. Being human, it is inevitable that some disagreements arose during the project. However, the disagreements were always handled effectively and in a fair way, typically by votes. In the start phase of the project we did a lot of tasks jointly among us, allowing us to understand how each group member worked and draw inspiration from one another, setting a common standard. This collaborative approach gave a sense of unity and a shared vision for the project. It should be noted that collaborating on all these tasks might have resulted in spending excessive time on certain tasks, but from the perspective of the end of the project, it confirms that the investment was worthwhile. The additional time spent on these tasks contributed to the overall quality of the project.

As the project progressed, the focus gradually shifted towards a combination of individual work and teamwork within specific disciplines. Each team member brought their unique expertise and skills to the table, creating a diverse and dynamic environment. The projects daily meetings started sitting down. Progressing through the project timeline it became clear that those meetings often took too much time and dragged out. The stand-up meeting were introduced and the meetings instantly became more efficient. The daily meetings and coordination among team members ensured that everyone stayed aligned with project goals, and the potential for keeping up with the time-line was set, however as discussed in section 2.4, it was shifted, but the goals were still met.

The open and inclusive team culture played an important role in encouraging knowledge sharing and keeping a supportive work environment. The group members were always willing to offer assistance to each other, provide feedback and contribute to the collective success of the project.

Effective teamwork was facilitated by utilizing various communication and collaboration tools, such as regular team meetings, shared documentation and the platforms and tools that Scrumban offers.

By combining our individual strengths and good teamwork, we were able to overcome challenges, make informed decisions and deliver a high-quality result. We created an environment for personal and professional growth.

11.1.2 File sharing platform

The file sharing platform utilized was Teams. USN has a license deal with Teams, and therefore the cloud storage supplied from them is a good option for the project. The platform was used for collaboration and ensure that all project files were easily accessible to the team members. Teams provided a centralized location where we could store and share all documents and other project related material. It was important for us to have all files shared and accessible for everyone at all times.

For some software this had its limitations though. For example, OrCAD was used among others for the wiring diagram. This document was not able to be open at the same time on different computers, it was not updated live. Therefore only one person could use it at a time. This was solved by good communication between the relevant individuals.

To not lose files the project had several backups. At the beginning of the project, it was deemed enough to backup once a week. As the project progressed further and files and documents were developed at a faster rate, the backups were consequently performed more often. At the end of the project a backup was created every day of the entire file system, as losing more than a day worth of progress could be detrimental. Losing an entire day would not be good for the project either, but there is a limit of how many backups can be done. The backups were kept for a week at a time before discarding, to save space. Backups have been very useful for the project as early on some Arduino code was overwritten and lost and another time when an entire file system was deleted by accident. Thankfully there was no setback at all, due to the backups.

11.1.3 Meeting leader and minutes

We implemented a rotating system for the roles of meeting leader and minute writer. The meeting leader was also responsible for sending out the meeting notices and writing an agenda. In each meeting, one person took on the role of leader and another person was assigned as the minute writer. The following meeting, the minute writer from the previous meeting assumed the role of leader, and the next person in the rotation took over the task of writing minutes. This system ensured equal participation and distribution of responsibilities within the team.

11.1.4 Timetables

The overview of the overall time spent on the project is shown in Table 11.1. All around, the team members achieved the expected target of 600 hours. However, there is an exception. The overview showcases an imbalance. Due to "force majeure", a group member had to be absent for a longer period of time. Otherwise the distribution is fairly equal as shown in fig. 11.1. Every member of the team has demonstrated exceptional dedication and commitment to the project, investing significant time and effort to shape and advance it to its current state. The teams collective contributions have been vital in driving the project forward and achieving notable progress.

Table 11.1: Timetable for project Mini Guardian.

	Sum		
	Admin	**Teknisk**	**Total**
Daniel	164	72	236
Jennifer	179	461	640
Karine	282.5	338.5	621
Marius	264	402	666
Stefan	277	377	654
Veronica	220.5	385	605.5
Sum	1387	2035.5	3422.5

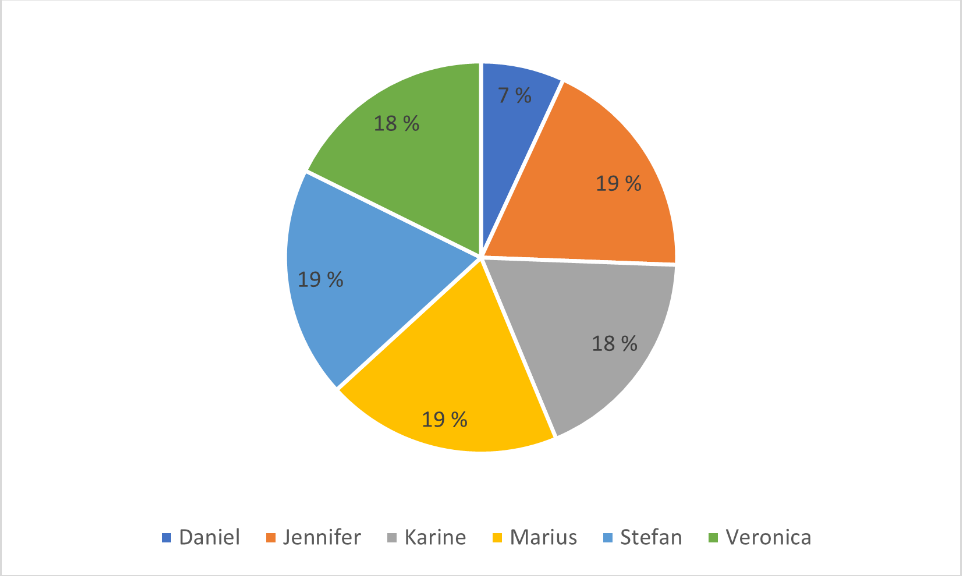


Figure 11.1: Pie chart showcasing number of hours per person in percentage [%]

11.2 Improvements

Even though the System is fully operational, there are several future improvements and enhancements that can be done. Some of the possible improvements remarked by project Mini Guardian can be found in this section.

11.2.1 Software improvements

JM

The system has undergone significant enhancements, incorporating crucial software functionality. However, there are still several areas that require further improvements. Specifically, the algorithm utilized for color detection can be further enhanced to improve its ability to detect a wider range of colors.

To enhance the system's capabilities, the integration of machine learning can enable the detection of various objects with different shapes. This would greatly improve its overall performance and versatility.

Furthermore, it would be beneficial to restructure the code to enhance readability. Although we inherited the code from last year and conducted some cleaning up, improving its organization and clarity was not prioritized at the time.

Additionally, there is an opportunity to incorporate the LiDAR feed within the graphical user interface (GUI). A designated space has been allocated within the GUI to seamlessly integrate this feature, providing users with a comprehensive and unified experience.

11.2.2 Electrical improvements

MH | SH

On the electrical side of the Effector, significant changes have been implemented. Despite this there are still things that need to be implemented or improved.

- **Reference position:** A reference position is still not implemented for the elevation axis, making it so that this axis needs to be manually adjusted to zero. The elevation axis is still lacking a over travel switch.
- **Voltage divider circuit:** The voltage divider used to lower the voltage from 24 V to ≈5 V works for its intention, but it absorbs some power which is not optimal. Although the PCB is ready for purchasing, there are other alternatives. The solution is dependent on limited space.
- **Power supply:** The 24 V power supply used for the project is underdimensioned compared to the amount of power the System could have utilized. Increasing the size of this should be prioritized.
- **Control system:** The motors are running without a closed loop control system which creates a deviation when moving the Effector.
- **Wiring:** A proper wiring harness should be created with the use of correct tools to make it correct. The Effector is wired up with a mix of Wago and Molex plugs, which is fine but this configuration takes up a lot of space which is minimal on the Effector.

11.2.3 Mechanical improvements

KC

Even though the Effectors mechanical state has improved a lot during this project, there are still elements in need of upgrades. The main aspects are listed here:

- **The belt drive:** This system should get a thorough inspection, including considering other options. Evaluate if the 3D-printed gears are good enough for this system and move the encoder to a location where the encoder axle is not exposed to radial loads exceeding 4.9 N.

- **Air pressure system:** This system is still leaking air, and needs to be reviewed.
- **Projectile:** Even though the projectiles passed though a redesign this year, they could benefit from a new review and more testing. This could increase the projectiles range and robustness.

11.3 Conclusion

SH

In conclusion, project Mini Guardian has achieved its goal of developing a scaled down model of the air defense system NASAMS, showcasing a successful proof of concept. The project team focused on implementing a LiDAR sensor and an AXIS camera to replicate key sensing capabilities essential for an effective air defense system.

Through extensive research and testing, the project team selected and integrated the LiDAR sensor and AXIS camera, allowing for object detection, classification and identification. The LiDAR sensor accurately detected objects in the Systems surroundings, while the AXIS camera provided visual information for target identification and categorization as either Hostile, Friendly or Unknown.

The project also addressed power supply issues inherited from the previous bachelors project, ensuring that the Systems power distribution was appropriately dimensioned. The remodeled Fire Distribution Center, transformed into a robust Power Supply, eliminated the need of other external connections in such a fashion that everything connects to the same source.

The successful implementation of project Mini Guardian demonstrates the teams competence in sensor infrastructure and integration. This proof of concept validates the projects viability and highlights its potential for further development of a miniature air defense systems.

With additional advancements and refinements, the Mini Guardian concept has the potential to be fully operational with no dead zones, having a complete 360° overview of its surroundings. If a LiDAR with more optimal sensor specifications tailored to this project is purchased, the System can further enhance its range, utilizing the full capabilities of the Effectors range.

In conclusion, project Mini Guardian has successfully demonstrated its ability to replicate crucial components of the NASAMS on a smaller scale, offering valuable insights into the complexities and challenges of developing such a System.

References

[1] J. K. Kristiansen, S. Tråstadkjølen, T. S. Nilsen, O. C. W. Kopland, M. Slettedal, and C. Mehren, "Project mini launcher," bachelor's thesis, University of South-Eastern Norway, 2022.

[2] I. Ouster, *Ouster Product Catalog - Q1 2023.* Ouster, Inc., 2023.

[3] W. Bolton, *Mechatronics: Electronic Control Systems in Mechanical and Electrical Engineering.* Pearson Education, 7 ed., 2018.

[4] OpenRocket, "Openrocket." `https://openrocket.info/index.html`, 2023. Online; accessed February - 2023.

[5] Joseph A. Huwaldt, "Plot digitizer." `https://sourceforge.net/projects/plotdigitizer/`, 2015. Online; accessed 18-april-2023.

[6] Kongsberg Defence & Aerospace, "Nasams air defence system." `https://www.kongsberg.com/kda/what-we-do/defence-and-security/integrated-air-and-missile-defence/nasams-air-defence-system/`, 2022. Online; accessed 13-May-2023.

[7] Department of Defence, MIL-STD-1477C, *Symbols for army displays (Metric)*, 1996.

[8] T. Johnsen, B. Hafskjold, S. Fagerlund, T. E. Strand, and A. Jensen, "Data fusion for improved air picture generation in air defence systems," Tech. Rep. MP-SCI-143-13, NATO, December 2005.

[9] C. Larman, *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and the Unified Process, Second Edition.* Pearson, 2001.

[10] C. Ellis, *The project book : the complete guide to consistently delivering great projects.* Milton, Qld: Wiley, 1st. ed., 2019.

[11] D. M. Buede and W. D. Miller, *The engineering design of systems: models and methods.* Wiley series in systems engineering and management, New York: Wiley, 3rd ed., 2016.

[12] K. Schwaber and J. Sutherland, *The scrum guide - The Definitive Guide to Scrum: The Rules of the Game.* Scrumguides, 2020.

[13] Scrum.org, the Home of Scrum, "What is scrum." `https://www.scrum.org/resources/what-is-scrum`, 2023. Online; accessed 24-January-2023.

[14] K. Eby, "The Newbie's Complete Guide to Kanban by Top Experts." `https://www.smartsheet.com/complete-kanban-project-management-guide-newbies-top-pm-experts`, 2016. Online; accessed 24-January-2023.

[15] Miro, "The visual collaboration plattorm for every team." `https://miro.com/index/`, 2023. Online; accessed 20-05-2023.

[16] A. Sols, *Systems engineering : theory and practice*. Madrid: Universidad Pontificia Comillas, 2014.

[17] D. Olsen, *The Lean Product Playbook: How to Innovate with Minimum Viable Products and Rapid Customer Feedback*. New York: Wiley, 1 ed., 2015.

[18] K. Baxter, *Risk management*. Fast track to success, Harlow, England: Financial Times/Prentice Hall, 1st ed., 2010.

[19] C. Chapman and S. Ward, *Project Risk Management: Process, Techniques and Insights*. Wiley, 2nd ed., 2003.

[20] "ISO/IEC/IEEE 15288 First edition 2015-05-15: ISO/IEC/IEEE International Standard - Systems and software engineering – System life cycle processes," 2015.

[21] D. D. Walden, G. J. Roedler, K. J. Forsberg, R. D. Hamelin, and T. M. Shortell, *Systems engineering handbook : a guide for system life cycle processes and activities*. Hoboken, N.J: Wiley, 4th. ed., 2015.

[22] IEEE, IEEE Std 829-2008, *IEEE Standard for Software and System Test Documentation*, 2008.

[23] "IEEE/ISO/IEC International Standard for Software and systems engineering–Software testing–Part 3:Test documentation," 2021.

[24] Nidec, *Manual setting type optical encoders REC/RES*, 2023.

[25] E. Adil, M. Mikou, and A. Mouhsen, "A novel algorithm for distance measurement using stereo camera," Tech. Rep. 1, The Institution of Engineering and Technology, April 2022.

[26] M. C. Budge and S. R. German, *Basic radar analysis*. Artech House radar library, Boston Massachusetts: Artech House, 2015.

[27] Velodyne, "What is lidar?." `https://velodynelidar.com/what-is-lidar/`, 2020. Online; accessed 18-May-2023.

[28] Insta360, "1.1.1 know the camera." `https://onlinemanual.insta360.com/pro1/en-us/basic/prepare/1`, 2023. Online; accessed 21-03-2023.

[29] I. Velodyne Lidar, *Velodyne Lidar Puck^{TM} Versatile Real-time Lidar Sensor*. Velodyne Lidar, Inc., 2019.

[30] KitWare, "Veloview: The velodyne lidar viewer based on paraview lidar." `https://www.paraview.org/veloview/`, 2023. Online; accessed 17-Mars-2023.

[31] T. L. Saaty, *Models, Methods, Concepts & Applications of the Analytic Hierarchy Process.* International Series in Operations Research & Management Science, New York, NY: Springer, 2nd ed., 2012.

[32] Creative Decisions Foundation, "Superdecisions." `https://www.superdecisions.com/downloads/index.php?section=win3_0_beta`, 2023. Online; accessed 15-February-2023.

[33] E. Mu, *Practical Decision Making : An Introduction to the Analytic Hierarchy Process (AHP) Using Super Decisions V2.* SpringerBriefs in Operations Research, Cham: Springer International Publishing : Imprint: Springer, 1st ed., 2017.

[34] I. Velodyne Lidar, *Velodyne Lidar ULTRA Puck^{TM} High resolution real-time LiDAR for autonomous systems.* Velodyne Lidar, Inc., 2019.

[35] I. Velodyne Lidar, *Velodyne Lidar HDL-32E High resolution real-time 3D LiDAR sensor.* Velodyne Lidar, Inc., 2019.

[36] LSLidar, *C32W Wide-view Mechanical LiDAR.* Leishen Intelligent System, 2020.

[37] R. P. Ltd., "Raspberry pi documentation." `https://www.raspberrypi.com/documentation/accessories/camera.html`, 2023. Online; accessed 21-05-2023.

[38] R. B. GmbH, "Velodyne Lidar ULTRA Puck$^{\text{TM}}$ High resolution real-time LiDAR for autonomous systems." `https://www.bosch-mobility.com/en/solutions/sensors/front-radar-sensor/`, 2023. Online; accessed 21-05-2023.

[39] Arbe, "Phonic Imaging Radar." `https://arberobotics.com/product/#product-resources`, 2023. Online; accessed 21-05-2023.

[40] A. C. AB, *AXIS M5074 PTZ Camera.* AXIS Communications AB, 2022.

[41] S. Bennett, *Schaum's outline of UML.* Schaum's outline series, New York: McGraw-Hill, 2nd ed., 2004.

[42] Zhi Yan and Tom Duckett and Nicola Bellotto, "Online learning for 3d lidar-based human detection: Experimental analysis of point cloud clustering and classification methods," *Autonomous Robots*, 2019. Online; accessed 25-April-2023.

[43] N. B. Zhi Yan, Tom Duckett, *Online Learning for Human Classification in 3D LiDAR-based Tracking.* Lincoln Centre for Autonomous Systems, University of Lincoln, UK, 2018.

[44] Open Robotics, "Ros." `https://www.ros.org/`, 2021. Online; accessed 18-May-2023.

[45] William Woodall, "rviz." `http://wiki.ros.org/rviz`, 2018. Online; accessed 19-May-2023.

[46] OpenCV, "About." `https://opencv.org/about/`, 2023. Online; accessed 28-February-2023.

[47] Tensorflow. `https://tensorflow.org`, 2023. Online; accessed 28-February-2023.

[48] Ultralytics, “Yolov5: The friendliest ai architecture you’ll ever use.” https://ultralytics.com/yolov5, 2023. Online; accessed 25-April-2023.

[49] PyTorch. https://pytorch.org/, 2023. Online; accessed 25-April-2023.

[50] Python, “About python.” https://www.python.org/about/, 2023. Online; accessed 20-05-2023.

[51] The Qt Company, “Development framework for cross-platform applications.” https://www.qt.io/product/framework, 2023. Online; accessed 10-May-2023.

[52] MediaWiki, “Opencv with qt.” https://wiki.qt.io/OpenCV_with_Qt, 2020. Online; accessed 10-May-2023.

[53] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, “Deep learning for generic object detection: A survey,” 2019.

[54] ravi0531rp, “Ssd-mobilenet-v2-fpnlite.” https://github.com/ravi0531rp/SSD-MobileNet-V2-FPNlite-, 2021. Online; accessed 20-05-2023.

[55] Nicholas Renotte, “Tensorflow object detection in 5 hours with python | full course with 3 projects.” https://www.youtube.com/watch?v=yqkISICHH-U&t=5666s&ab_channel=NicholasRenotte, 2021. Online; accessed 20-05-2023.

[56] Jupyter, “About us: Project jupyter’s origins and governance.” https://jupyter.org/about, 2023. Online; accessed 20-May-2023.

[57] heartexlabs, “Labelimg is now part of the label studio community.” https://github.com/heartexlabs/labelImg, 2022. Online; accessed 25-April-2023.

[58] Roboflow. https://roboflow.com/, 2023. Online; accessed 25-April-2023.

[59] Google Colab, “Frequently asked questions.” https://research.google.com/colaboratory/faq.html, 2023. Online; accessed 25-April-2023.

[60] Lutz Roeder, “Lutz roeder’s netron.” https://netron.app/, 2023. Online; accessed 26-April-2023.

[61] J. Scully, R. Flynn, P. Gallagher, E. Carley, and M. Daly, “Improved Type III solar radio burst detection using congruent deep learning models,” 2023.

[62] Microsoft, “Visual Studio: IDE and Code Editor for Software Developers and Teams.” https://visualstudio.microsoft.com/, 2023. Online; accessed 29April-2023.

[63] UNeedCryDear, “Onnx conversion error for opencv dnn.” https://github.com/ultralytics/yolov5/issues/10665, 2023. Online; accessed 26-April-2023.

[64] UNeedCryDear, “yolov5-opencv-dnn-cpp.” https://github.com/UNeedCryDear/yolov5-opencv-dnn-cpp, 2023. Online; accessed 19-May-2023.

[65] eyyupT, “Opencv-cpp-show-webcam-stream-on-qt-gui.” https://github.com/eyyupT/OpenCV-CPP-show-webcam-stream-on-Qt-GUI, 2016. Online; accessed 07-May-2023.

[66] L. TP-Link Technologies Co, *TL-SG105PE(UN)1.0 Datasheet.* TP-Link Technologies Co., Ltd, 2020.

[67] Wireshark Foundation, "About wireshark." `https://www.wireshark.org/about.html`, 2023. Online; accessed 18-May-2023.

[68] Arduino, "Arduino mega 2560 rev3." `https://store.arduino.cc/products/arduino-mega-2560-rev3?_gl=1*xqo0n7*_ga*MjA5ODI1NTg1OS4xNjc3MDYzMjU5*_ga_NEXN8H46L5*MTY4NDUzMjAzMC4yNi4xLjE2ODQ1MzI2OTEuMC4wLjA.`, 2023. Online; accessed 20-May-2023.

[69] M. Margolis, *Arduino cookbook : recipes to begin, expand, and enhance your projects.* Sebastopol, CA: O'Reilly Media, 3rd ed., 2020.

[70] Tim Clephas, "Ros support for velodyne 3d lidars." `https://github.com/ros-drivers/velodyne`, 2023. Online; accessed 20-05-2023.

[71] E. R. Harold, "Udp datagrams and sockets," in *Java Network Programming*, United States: O'Reilly Media, Inc, 2008.

[72] C. M. Kohlhoff, "Daytime.5 - a synchronous udp daytime server." `https://www.boost.org/doc/libs/1_82_0/doc/html/boost_asio/tutorial/tutdaytime5.html`, 2023. Online; accessed 21-03-2023.

[73] C. Edwards, *Calculus : early transcendentals.* Harlow: Pearson, 7th ed.; new international ed., 2014.

[74] TullyFoote, "Ros/tutorials." `http://wiki.ros.org/ROS/Tutorials`, 2022. Online; accessed 20-05-2023.

[75] rukie, "Getting started with the velodyne vlp16." `http://wiki.ros.org/velodyne/Tutorials/Getting%20Started%20with%20the%20Velodyne%20VLP16`, 2019. Online; accessed 20-05-2023.

[76] PCdeal.no, "Dell precision 5530 i9-8950hk 32gb 512ssd 15" quadro p2000." `https://www.pcdeal.no/-studie/-universitetet-ingenior/dell-precision-5530-i9-8950hk-32gb-512ssd-15-quadro-p2000`, 2023. Online; accessed 4-may-2023.

[77] Department of Defence, MIL-STD-810G, *Environmental Engineering Considerations and Labroratory Tests*, 2008.

[78] GitHub Docs, "Hello world." `https://docs.github.com/en/get-started/quickstart/hello-world`, 2023. Online; accessed 24-February-2023.

[79] GitHub Docs, "About git." `https://docs.github.com/en/get-started/using-git/about-git`, 2023. Online; accessed 12-march-2023.

[80] OMRON, *Miniature Basic Switch.* OMRON, 2014.

[81] Xdynemo, "Design of cam profile in matlab." `https://xdynemo.wordpress.com/design-of-cam-profile-in-matlab/`, 2022. Online; accessed 15-April-2023.

[82] Nordcad, "Orcad-trial." https://www.nordcad.no/orcad-trial/, 2023. Online; accessed 19-april-2023.

[83] SolidWorks, "Solidworks electrical schematic." https://www.solidworks.com/product/solidworks-electrical-schematic, 2023. Online; accessed 19-april-2023.

[84] haBit Software, "Elcad skjema." http://www.habit.no/ElCad_Skjema.htm, 2023. Online; accessed 19-april-2023.

[85] Cadence Design Systems, "*OrCAD Capture.*" https://resources.pcb.cadence.com/orcad-datasheets/orcad-capture-6. Online; accessed 19-april-2023.

[86] Autodesk, "Electrical design software." https://www.autodesk.com/solutions/electrical-design, 2023. Online; accessed 19-april-2023.

[87] "Grafiske symboler for el- og ekom-dokumentasjon : NEK-håndbok = Graphical symbols for electrotechnical documentation : NEK-guidelines," 2017.

[88] Smart Prototyping, "Smart prototyping." https://www.smart-prototyping.com/PCB-Prototyping.html, 2023. Online; accessed 20-05-2023.

[89] PentaLogix, "Viewmate." https://www.pentalogix.com/t/software-products/viewmate, 2023. Online; accessed 20-05-2023.

[90] Mathworks, "Simscape electrical - model and simulate electronic, mechatronic, and electrical power systems." https://se.mathworks.com/products/simscape-electrical.html, 2022. Online; accessed 20-03-2023.

[91] S. Gudvangen, "Vibratory measurements.." Lecture notes, IRI, TNM, USN, Feb. 2022.

[92] S. Gudvangen, "Accelerometers.." Lecture notes, IRI, TNM, USN, Feb. 2022.

[93] Dytran Instruments Inc, "Series 3224 ultra miniature sensor." https://www.dytran.com/Model-3224A1-Miniature-Accelerometer-P1344/, 2023. Online; accessed 20-05-2023.

[94] D. I. Inc, OUTLINE/INSTALLATION DRAWING, MODEL3224A1*, 2W033, 2023 [Online]*. Dytran Instruments Inc, 2023.

[95] I. Dytran Instruments, OUTLINE/INSTALLATION DRAWING, MODEL 4105C*, 127-4105C , 1996 [Online]*. Dytran Instruments, Inc, 2005.

[96] I. Avid Technology, Fast Track Pro User Guide*, 9329-65014-00 , 2010 [Online]*. Avid Technology, Inc, 2010.

[97] Audacity, "About." https://www.audacityteam.org/about/, 2023. Online; accessed 20-05-2023.

[98] Adobe, "Audio recording and editing software." https://www.adobe.com/products/audition.html, 2023. Online; accessed 20-05-2023.

[99] C. Schroder, "The book of audacity : record, edit, mix, and master with the free audio editor," 2011.

[100] S. Gudvangen, "Fourier-based spectral analysis." Lecture notes, IRI, TNM, USN, Aug. 2021.

[101] Y. Zeng, H. Mao, D. Peng, and Z. Yi, "Spectrogram based multi-task audio classification," *Multimedia tools and applications*, vol. 78, no. 3, pp. 3705–3722, 2017.

[102] J. S. Aspen, *Maskindeler 1*. Oslo: Universitetsforlaget, 1968.

[103] J. Flynt, "Ball Bearings vs Roller Bearings: How Are They Different?." `https://3dinsider.com/ball-vs-roller-bearings/`, 2020. Online; accessed 23-February-2023.

[104] M. G. Mousavi, "Kort om korrosjon." Lecture notes, IRI, TNM, USN, 2022.

[105] NTN, 6912LLU/5K, *Technical data*, 2023.

[106] R. G. Budynas, J. E. Shigley, and J. K. Nisbett, *Shigley's mechanical engineering design*. New York: McGraw-Hill, 9th ed. ed., 2011.

[107] S. Kalpakjian, *Manufacturing engineering and technology*. Singapore: Pearson Education, in si units 7th ed., 2014.

[108] Adobe, "What are step files and how do you open them?." `https://www.adobe.com/creativecloud/file-types/image/vector/step-file.html`, 2023. Online; accessed 18-April-2023.

[109] M. Young and H. Okiishi, *Introduction to Fluid Mechanics*. Wiley, 5th ed., 2012.

[110] SolidWorks, "Mesh quality checks." `https://help.solidworks.com/2021/english/SolidWorks/cworks/c_Mesh_Quality_Checks.htm`, 2023. Online; accessed 19-April-2023.

[111] M. G. Mousavi, "Destruktiv materialprøving.." Lecture notes, IRI, TNM, USN, 2022.

[112] M. Brynjulfsen, "Fatigue of extruded aa6082 and aa7108 alloys," Master's Thesis, Norwegian University of Science and Technology, 2015.

[113] K. Enger, "Vibrasjoner." Lecture notes, IRI, TNM, USN, 2022.

[114] FormLabs, "Fuse 1+ 30w." `https://formlabs.com/eu/3d-printers/fuse-1/`, 2023. Online; accessed 27-April-2023.

[115] Ultimaker, "Ultimaker s5." `https://ultimaker.com/3d-printers/ultimaker-s5`, 2023. Online; accessed 27-April-2023.

[116] FormLabs, "Formlabs software." `https://formlabs.com/software/`, 2023. Online; accessed 27-April-2023.

[117] Formlabs, *Nylon 12 Powder*, 2020. Version v1.

[118] Ultimaker, *Ultimaker Tough PLA*, 2022. Version v2.00.

[119] Ultimaker, *Ultimaker ABS*, 2022. Version v5.00.

[120] KitWare, "Paraview." `https://www.paraview.org/`, 2023. Online; accessed 17-Mars-2023.

Appendix A

User manual

A Brukermanual - Mini Guardian

Denne brukermanualen er forbeholdt at datamaskinen tilhørende prosjektet brukes. Dette er også en videreføring av fjorårets brukermanual [1].

A.1 Forberedelser før bruk av Mini Guardian uten avfyring

1. Effektor plasseres på strukturen. Dersom strukturen ikke benyttes skal Effektoren plasseres på et stabilt og plant underlag.
2. Effektor må kobles til strømforsyning med kabel for strømtilførsel.
3. PC med GUI, LiDAR, AXIS kamera og Effektor kobles til strømforsyning med nettverkskabler.
4. Kontroller at sikkerhetsbryter på Effektor er i av-posisjon, og at sikkerhetslokket foran toppstruktur på Effektor er montert.
5. Toppstruktur til Effektor må være rotert til nullpunkt.
6. Skru på hovedbryter på strømforsyningen.
7. Start opp objektdetektering [A.4] og GUI [A.4].

A.2 Forberedelser før bruk av Mini Guardian med avfyring

1. Utfør alle 8 punkter fra "Forberedelser før bruk av Mini Guardian uten avfyring".
2. Effektor må tilkobles trykkluftslange.
3. Operatør og tilskuere må bruke vernebriller.
4. Fjern sikkerhetslokk ved å løfte lokket opp og ut.
5. Sett sikkerhetsbryter for avfyring i på-posisjon.

A.3 Avslutningsprosedyre for systemet

1. Sett sikkerhetsbryter for avfyring i av-posisjon.
2. Sett på sikkerhetslokk ved å skyve lokket inn og ned.
3. Kjør avslutingsprosess med LiDAR [A.4], og GUI ved å lukke GUI vinduet.
   NB: Avslutingsprosessen må alltid kjøres før systemet skrus av og strømmen kobles fra.
4. Skru av hovedbryter på strømforsyning.
5. Nettverkskabler i strømforsyning kobles fra.
6. Strømkabel kobles fra.
7. Trykkluftslange kobles fra.

A.4 VIKTIG INFORMASJON

- Ved opplasting av kode til Arduino Nano skal TX/RX kobles fra.
- Ved opplasting av kode til Arduino Mega skal kun strømtilførsel fra PC brukes.
  HUSK å skru av strømforsyning med strømtilførsel til Effektoren.
  Dersom Effektoren er koblet til strømforsyning via den grå forsyningskabelen, må nødstopp **IKKE** være trykket inn.
- Når systemet skal kobles til via strømforsyning, **HUSK** å endre nettverkstilkoblingen slik at den står på IP adresse "192.168.0.2".
- Dersom nødstopp blir tatt i bruk, **HUSK** å release den.
- Ikke skru Effektoren under horisonten, det vil føre til at toppstrukturen mister feste.

Start opp LiDAR med objektdetektering

Denne delen av manualen er en modifisert versjon av [75] tilpasset vårt system.
Koble til LiDARen og sørg for å ha ethernetkabelen til både PC og LiDAR koblet til strømforsyning.

1. **Åpne terminalen og gå til ROS workspacet:**

```
$ cd ∼/miniguardian_ws
```

Start opp alle nødvendige noder til objekt detekteringen i hvert sitt terminalvindu:

1. **Velodyne LiDAR VLP-16 node:**

```
$ roslaunch velodyne_pointcloud VLP16_points.launch
```

2. **Objekt detekterings node:**
   Dersom det er gjort endringer i koden, kjøres disse kommandoene først:

```
$ catkin_make –pkg object_detection
```

```
$ source devel/setup.bash
```

Deretter:

```
$ rosrun object_detection object_detection
```

Dersom ROS ikke finner filen du prøver å åpne, kjør denne kommandoen og prøv igjen:

```
$ source devel/setup.bash
```

3. **RViz node med frame «velodyne»:**

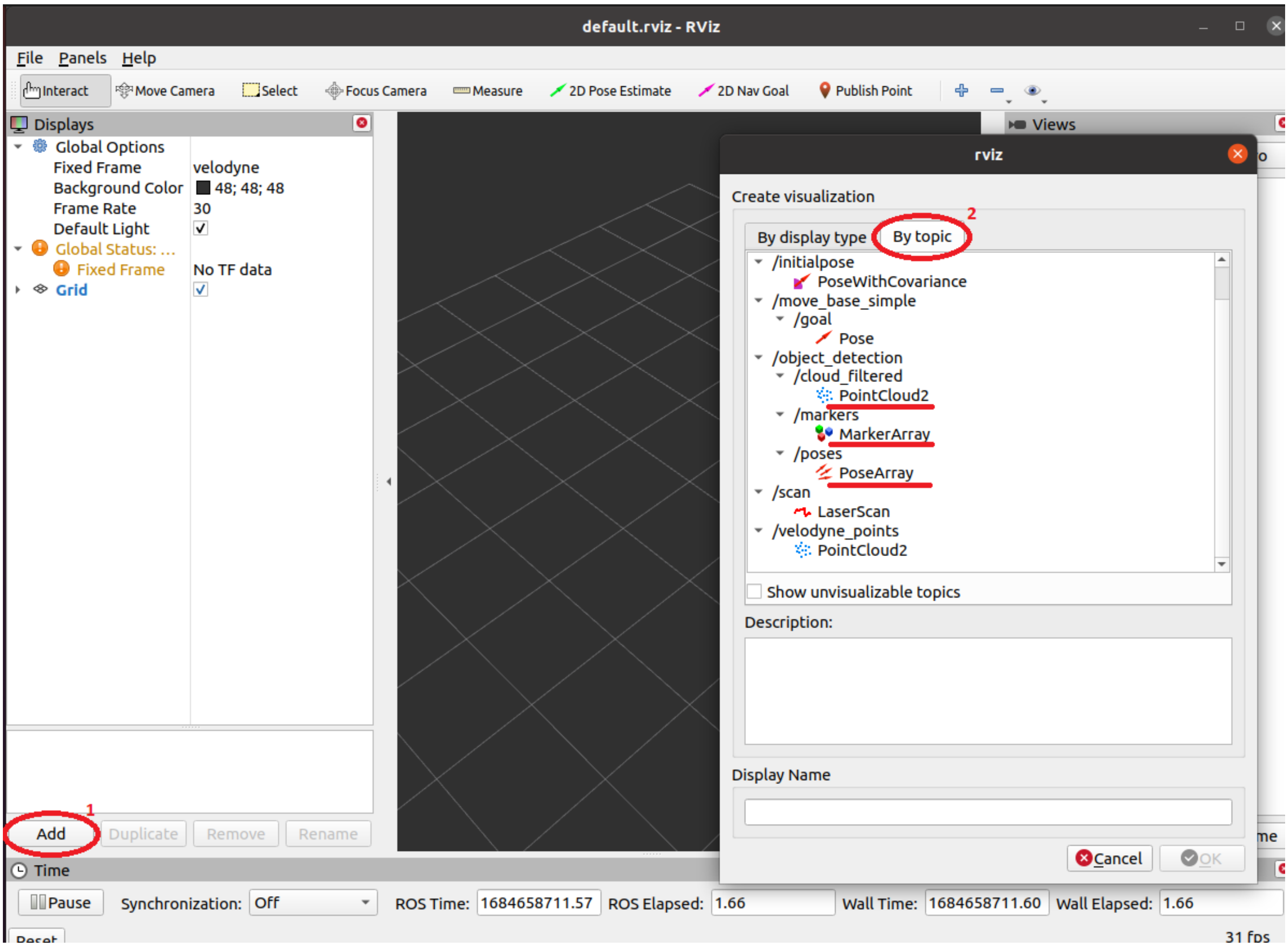
```
$ rosrun rviz rviz -f velodyne
```

Når du er inne i RViz, må topicene åpnes slik at objekt detekteringen fungerer.

1. Trykk «Add» nederst i venstre hjørne, og deretter «By topic»

2. Legg så til «velodyne_pointcloud» under Point Cloud 2:

3. Gjenta stegene over for topic «/cloud_filtered» under Point Cloud 2, «/markers» og «/posearray» under /object_detection.



Global Options
Fixed Frame: velodyne
Background Color: 48; 48; 48
Frame Rate: 30
Default Light: ☑
Global Status: ...
Fixed Frame: No TF data
Grid: ☑

Create visualization

```
/initialpose
    PoseWithCovariance
/move_base_simple
  /goal
      Pose
/object_detection
  /cloud_filtered
      PointCloud2
  /markers
      MarkerArray
  /poses
      PoseArray
/scan
    LaserScan
/velodyne_points
    PointCloud2
```

□ Show unvisualizable topics

Description:

Display Name

Synchronization: Off
ROS Time: 1684658711.57
ROS Elapsed: 1.66
Wall Time: 1684658711.60
Wall Elapsed: 1.66
31 fps

Figure 1: RViz.

Avslutt LiDAR med objektdetektering

Avslutt alle nodene med ctrl + c.
Koble ut Ethernet kabelen fra strømforsyning, og koble fra strømmen til LiDARen.

Oppsett ROS og Velodyne VLP-16

Sett opp datamaskinen til å kommunisere med Velodyne sensoren:

1. Koble til LiDARen og sørg for å ha ethernetkabelen koblet til datamaskinen din.

2. Foreløpig, skru av WiFi.

Konfigurer IP adresse:

1. Gå til datamaskinens meny, søk etter «Settings» og gå til «Network».

2. Under «Wired» velger du + for å legge til et nytt kablet nettverk, eller velg Velodyne LiDAR og sørg for at adressene stemmer overens med punkt 3.
   Ethernet kabelen må være koblet til for at «Wired» skal være tilgjengelig.

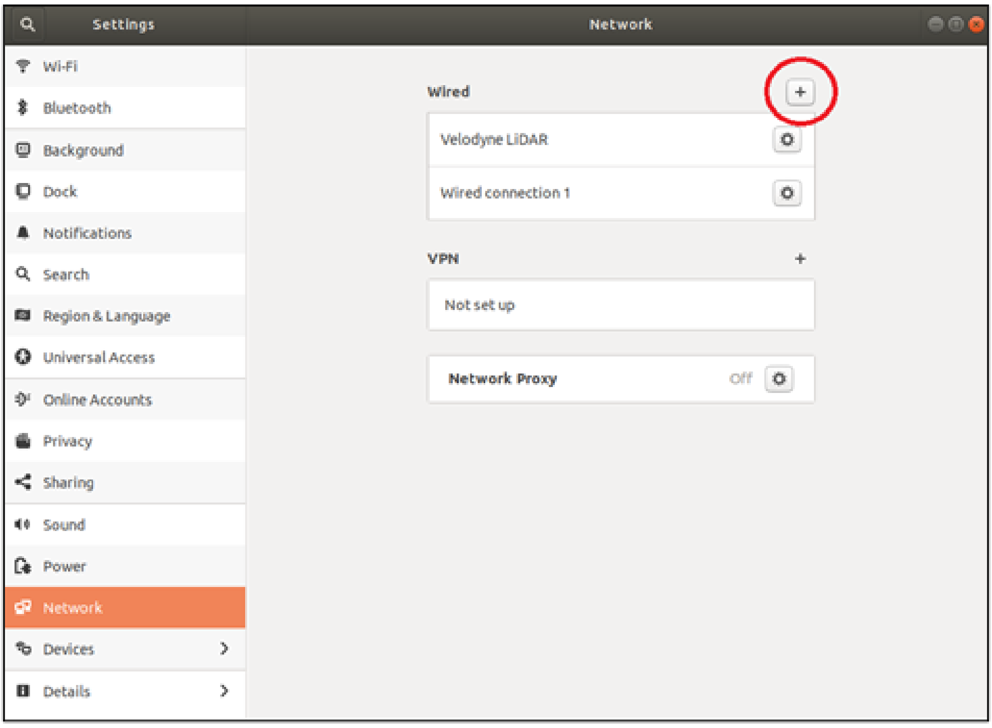
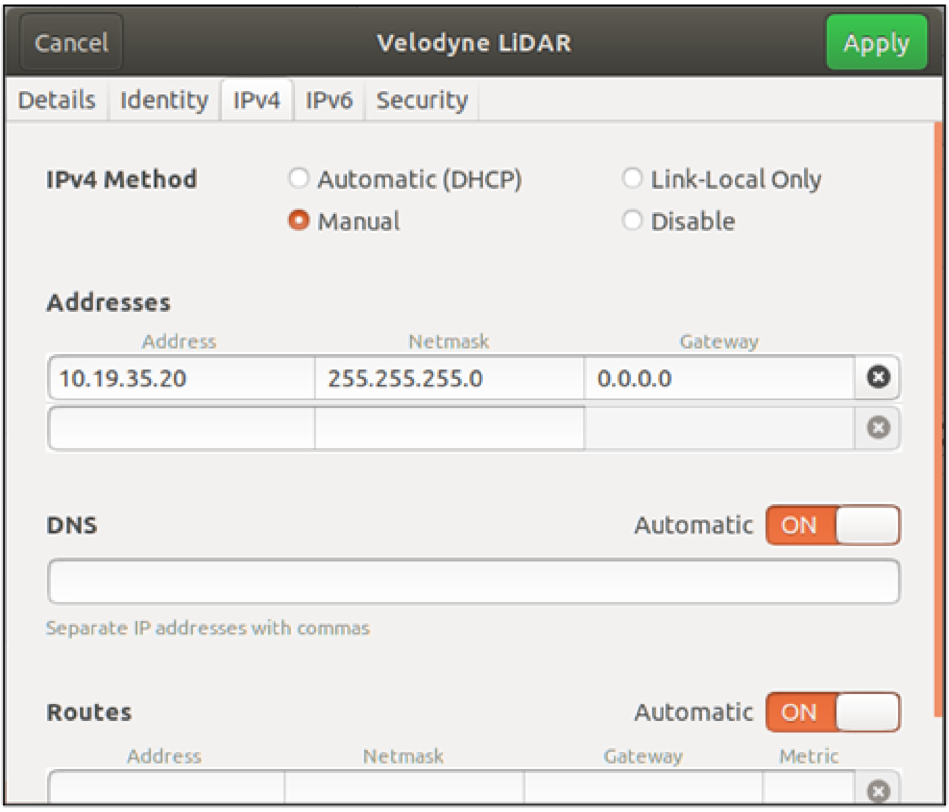


Figure 2: Velodyne LiDAR

3. Gi nettverket et passende navn, som «Velodyne LiDAR», og under IPv4 legger du inn IP adresse «10.19.35.20», Netmask «255.255.255.0» og Gateway «0.0.0.0» som vist nedenfor.

4. Trykk «Apply» for å lagre endringene.



IPv4 Method ○ Automatic (DHCP) ● Manual ○ Link-Local Only ○ Disable

Addresses

Address	Netmask	Gateway
10.19.35.20	255.255.255.0	0.0.0.0

DNS Automatic ON

Separate IP addresses with commas

Routes Automatic ON

Address	Netmask	Gateway	Metric

Koble sammen LiDAR og datamaskin. Bytt ut NAME med navnet på Ethernet utgangen:

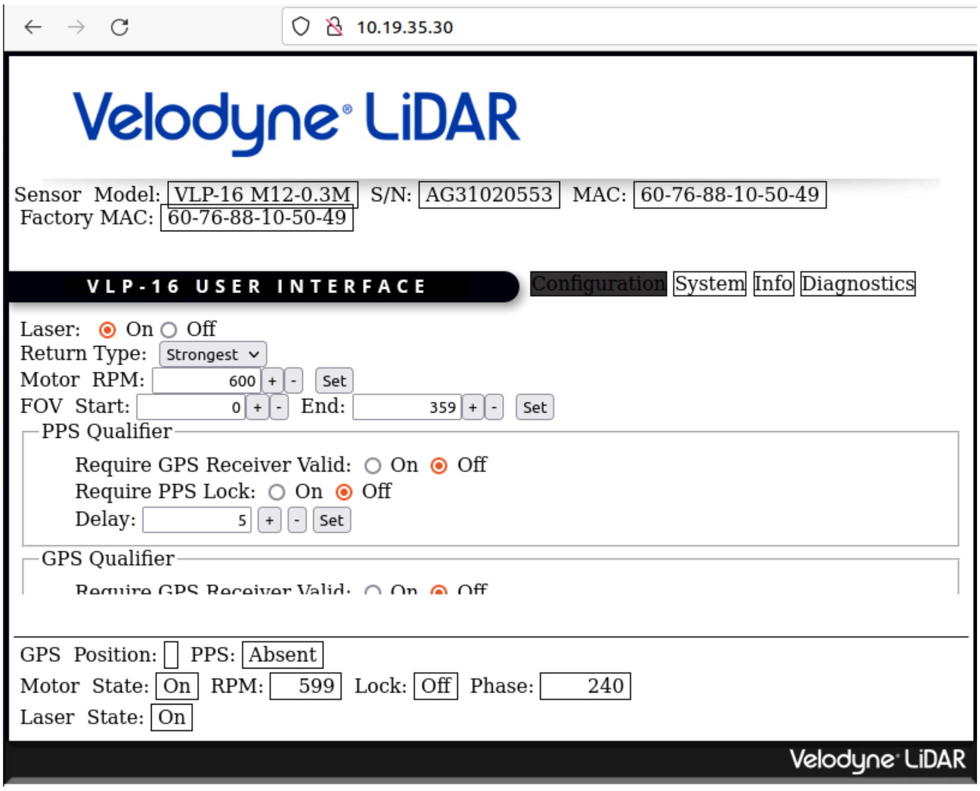
1. Koble IP-adressen til Ethernet utgangen.
   Navnet på Ethernet utgangen kan finnes ved å kjøre kommandoen «ifconfig». Finn IP-adressen til LiDARen, og navnet som står her er navnet på utgangen:

   ```
   $ sudo ifconfig NAME 10.19.35.20
   ```

2. Legg til en statisk rute til LiDARen. Bytt ut NAME med navnet på Ethernet utgangen:

   ```
   $ sudo route add 10.19.35.30 NAME
   ```

Sjekk konfigurasjonene: For å se om konfigurasjonene stemmer og om det er kontakt, åpne nettleseren og skriv inn nettverksadressen til LiDARen, som ble satt i punktene over:



Sensor Model: VLP-16 M12-0.3M S/N: AG31020553 MAC: 60-76-88-10-50-49
Factory MAC: 60-76-88-10-50-49

Laser: ● On ○ Off
Return Type: Strongest
Motor RPM: 600
FOV Start: 0 End: 359

PPS Qualifier
Require GPS Receiver Valid: ○ On ● Off
Require PPS Lock: ○ On ● Off
Delay: 5

GPS Qualifier
Require GPS Receiver Valid: ○ On ● Off

GPS Position: PPS: Absent
Motor State: On RPM: 599 Lock: Off Phase: 240
Laser State: On

Figure 3: Velodyne LiDAR IP.

Installer ROS avhengigheter. Bytt ut VERSION med din ROS distrubisjon:

1. Skru på WiFi
2. Åpne terminalen og kjør kommandoen:

```
$ sudo apt-get install ros-VERSION-velodyne
```

Installer VLP-16 driveren:

1. I terminalen, gå til ditt ROS workspace, finn «src» mappen og kjør følgende kommando:

```
$ cd ~/catkin_ws/src/ && git clone https://github.com/ros-
drivers/velodyne.git
```

2. Oppdaterer alle avhengigheter. Bytt ut YOURDISTRO med din ROS distrubisjon:

```
$ rosdep install –from-paths src –ignore-src –rosdistro YOURDISTRO -y
```

3. Bygg workspacet:

```
$ cd ~ /catkin_ws/ && catkin_make
```

Vis frem LiDAR dataen i RViz:

1. Kjør følgende kommando i terminalen:

   ```
   $ roscore
   ```

2. Åpne et nytt terminalvindu og kjør følgende kommando:

   ```
   $ roslaunch velodyne_pointcloud VLP16_points.launch
   ```

3. Du kan se all dataen som blir publisert og sendt til følgende topic:

   ```
   $ rostopic echo /velodyne_points
   ```

4. Så kjører du RViz med frame velodyne:

   ```
   $ rosrun rviz rviz -f velodyne
   ```

5. Trykk "Add" nederst i venstre hjørne, deretter "By topic" og velg /velodyne_points - PointCloud2".

Oppstart av GUI

1. Åpne Qt
2. Trykk "Open Project..." og velg /home/miniguardian/Documents/MiniGuardian/-Code/GUI/GUI.pro
3. Trykk på den grønne trekanten nederst i venstre hjørne for å kjøre GUIen.

Oppstart av Arduino Mega kode

1. Åpne Visual Studio Code
2. Trykk "File", deretter "Open Workspace from file..."
3. Velg filen under filstien /home/miniguardian/Documents/MiniGuardian/Code/Aruduino_Mega_ workspace
4. **HUSK å koble fra strømforsyning ved opplasting av kode.**

Oppstart av Arduino Nano kode

1. Åpne Arduino IDE
2. Trykk "File", deretter "Open"
3. Velg filen under filstien /home/miniguardian/Documents/MiniGuardian/Code/Aruduino_Nano_
4. **HUSK å koble fra TX/RX ved opplasting av kode.**

A.5 Elektrisk System.

Åpning av power supply boks

1. Fjern lister på motsatt side av 230V AC uttak.
2. Det er fire skruer med tverrspor hode som skrus ut med flat skrutrekker. Bak dette lokket kommer man inn til boksen.
3. Om det trengs kan også panelet med nødstopp fjernes. Disse skruene sitter fast med mutter på innsiden, kan være knotete å få ut. Fjern lister først.
4. Lokket på toppen løftes av, merk at kabler sitter tilkoblet til nødstopp.
5. Dersom pleksiglass bunnplaten behøves å taes ut, sitter denne fast med mutter i hjørner. Disse løsnes, koble også ut kabler til ethernet switch.
6. For å løfte ut bunnplaten må holderen til switchen fjernes først. Denne sitter fast med 4 M3 unbraco skruer.
7. Om man vil få tilgang til kabelkanal kan lokket dras forsiktig ut mot switchens strøm tilførsel. Ikke løft den opp.

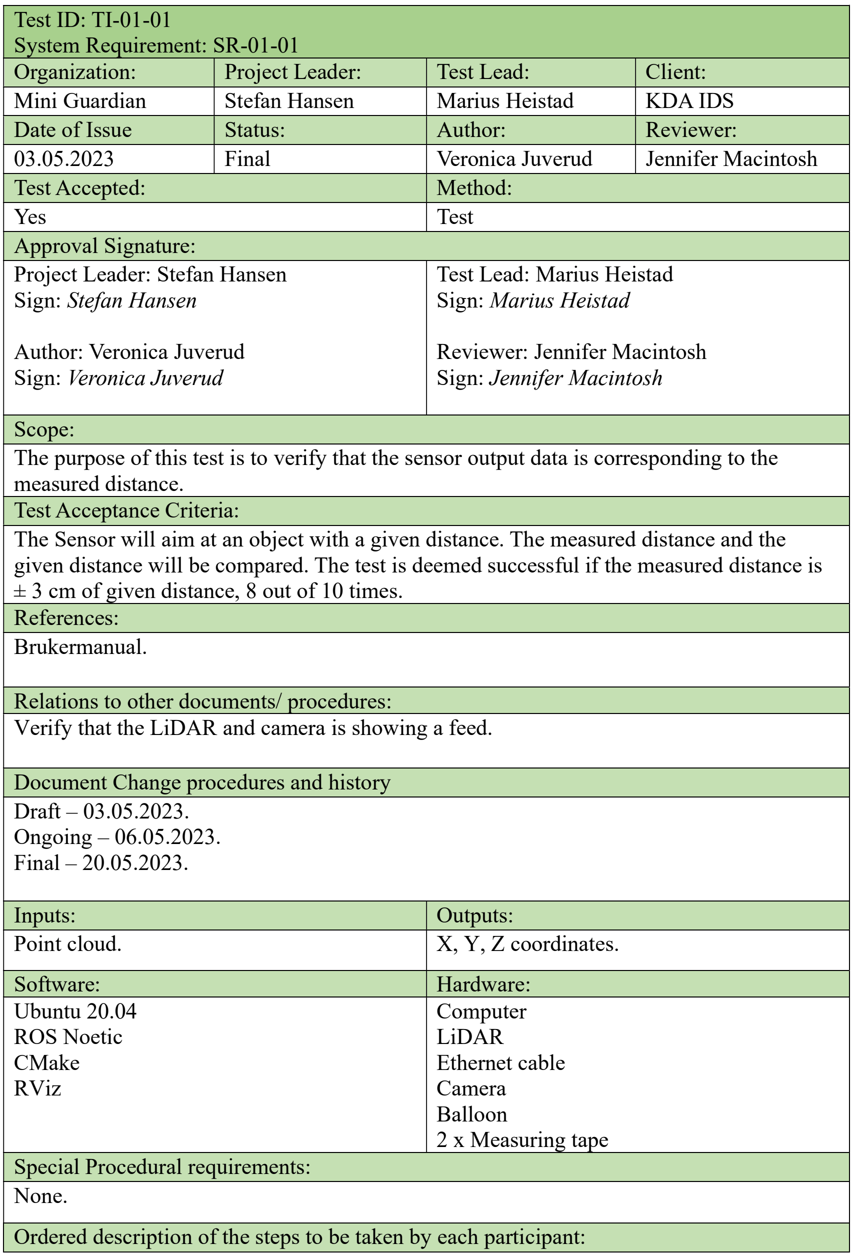
Effektoren har en rekke sikringer som er plassert på forskjellige steder. Denne listen forklarer i kortfattethet hvor disse befinner seg.

- Den første sikringen i systemet er inne i power supply boksen. Her er en 10A automatsikring koblet inn rett etter inntaket.
  - Denne har en jordfeilbryter. Den skal testes en gang i måneden for å sikre funksjonalitet. Dette gjøres ved å trykke på jordfeilbryterens Test-knapp mens sikringen er skrudd på. Når knappen trykkes inn skal sikringen slå ut og bryte tilførselen.
- For å komme til denne må boksen åpnes, les prosedyre ovenfor for instrukser.
- Automatsikring resettes på vanlig måte.

Appendix B

Testing documents

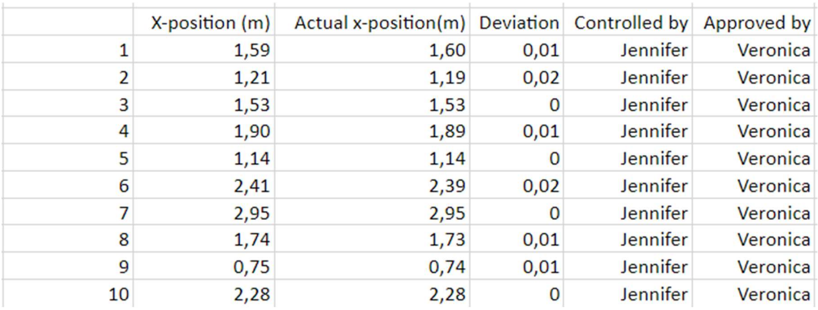
A Test report: TI-01-01



Test ID: TI-01-01 System Requirement: SR-01-01			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
03.05.2023	Final	Veronica Juverud	Jennifer Macintosh
Test Accepted:		Method:	
Yes		Test	
Approval Signature:			
Project Leader: Stefan Hansen Sign: *Stefan Hansen* Author: Veronica Juverud Sign: *Veronica Juverud*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Jennifer Macintosh Sign: *Jennifer Macintosh*	
Scope:			
The purpose of this test is to verify that the sensor output data is corresponding to the measured distance.			
Test Acceptance Criteria:			
The Sensor will aim at an object with a given distance. The measured distance and the given distance will be compared. The test is deemed successful if the measured distance is ± 3 cm of given distance, 8 out of 10 times.			
References:			
Brukermanual.			
Relations to other documents/ procedures:			
Verify that the LiDAR and camera is showing a feed.			
Document Change procedures and history			
Draft – 03.05.2023. Ongoing – 06.05.2023. Final – 20.05.2023.			
Inputs:		Outputs:	
Point cloud.		X, Y, Z coordinates.	
Software:		Hardware:	
Ubuntu 20.04 ROS Noetic CMake RViz		Computer LiDAR Ethernet cable Camera Balloon 2 x Measuring tape	
Special Procedural requirements:			
None.			
Ordered description of the steps to be taken by each participant:			

– Log: Excel sheet.
– Setup: Power up the LiDAR in a suitable test area.
Follow the user manual provided – Section “Start opp LiDAR med objektdetektering”.
Place a balloon in the LiDAR FoV.
Place measuring tape from center of the LiDAR and position it along the x-axis.
Use another measuring tape or a straight object, align it with the balloon and determine the exact distance from the first measuring tape.
Compare the distance given in the ROS terminal and the measured distance.
– Proceed: Compare the distance given in the ROS terminal and the measured distance. Write down both measurements.
– Measurement: A physical measurement in the LiDAR x-axis, and measurements read from the ROS terminal.
– Shut down: In case of any occurrences during the test execution that will temporarily suspend the test, repeat the user manual and setup steps to ensure the accuracy and consistency of the test results.
– Restart: Follow the user manual and setup steps.
– Stop: Follow the steps below “Avslutt LiDAR med objektdetektering” in this user manual:
– Wrap-up: The measurements are written down during the test, and deviation is calculated after the test is finished. The deviation determines whether the test is accepted.
– Contingencies: In the case of any interruptions during the test execution that could potentially impact the accuracy of the measurements, both physical measurements and measurement readings will be repeated after the situation has resolved.

Procedure result:



	X-position (m)	Actual x-position(m)	Deviation	Controlled by	Approved by
1	1,59	1,60	0,01	Jennifer	Veronica
2	1,21	1,19	0,02	Jennifer	Veronica
3	1,53	1,53	0	Jennifer	Veronica
4	1,90	1,89	0,01	Jennifer	Veronica
5	1,14	1,14	0	Jennifer	Veronica
6	2,41	2,39	0,02	Jennifer	Veronica
7	2,95	2,95	0	Jennifer	Veronica
8	1,74	1,73	0,01	Jennifer	Veronica
9	0,75	0,74	0,01	Jennifer	Veronica
10	2,28	2,28	0	Jennifer	Veronica

The deviation of the measurements taken during the test indicates that 8 out of 10 times, the measurements were within ± 3 cm, proving the success of the test.

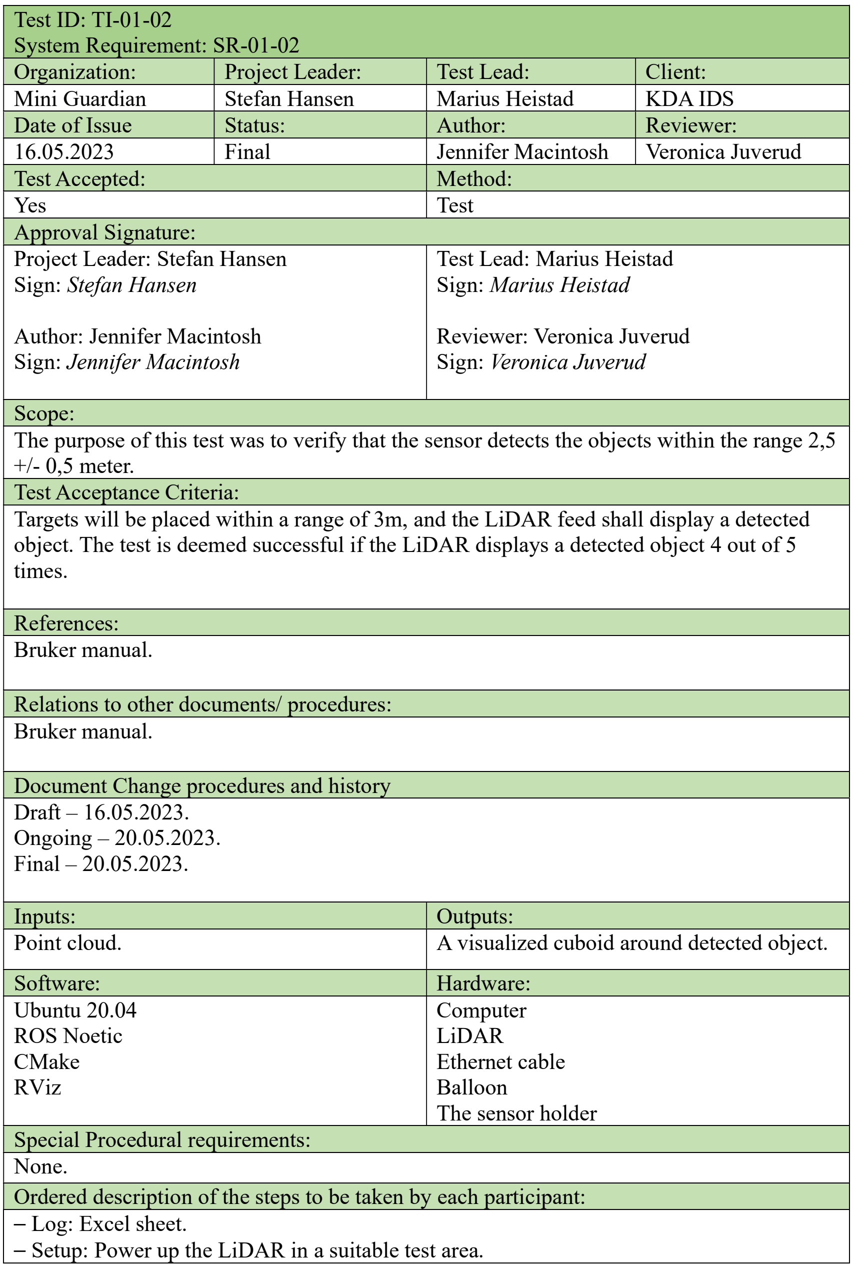
Environmental Information:

Dronesonen, 15:00, sun outside, 20 degrees

Anomalous events:

The balloon was blown away, solved it by taping it to the ground. Restarted the test.

B Test report: TI-01-02



Test ID: TI-01-02 System Requirement: SR-01-02			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
16.05.2023	Final	Jennifer Macintosh	Veronica Juverud
Test Accepted:		Method:	
Yes		Test	
Approval Signature:			
Project Leader: Stefan Hansen Sign: *Stefan Hansen* Author: Jennifer Macintosh Sign: *Jennifer Macintosh*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Veronica Juverud Sign: *Veronica Juverud*	
Scope:			
The purpose of this test was to verify that the sensor detects the objects within the range 2,5 +/- 0,5 meter.			
Test Acceptance Criteria:			
Targets will be placed within a range of 3m, and the LiDAR feed shall display a detected object. The test is deemed successful if the LiDAR displays a detected object 4 out of 5 times.			
References:			
Bruker manual.			
Relations to other documents/ procedures:			
Bruker manual.			
Document Change procedures and history			
Draft – 16.05.2023. Ongoing – 20.05.2023. Final – 20.05.2023.			
Inputs:		Outputs:	
Point cloud.		A visualized cuboid around detected object.	
Software:		Hardware:	
Ubuntu 20.04 ROS Noetic CMake RViz		Computer LiDAR Ethernet cable Balloon The sensor holder	
Special Procedural requirements:			
None.			
Ordered description of the steps to be taken by each participant:			
– Log: Excel sheet. – Setup: Power up the LiDAR in a suitable test area.			

Follow the user manual provided – Section “Start opp LiDAR med objektdetektering”.

Place a balloon in the LiDAR FoV.

– Proceed: Visually confirm that the balloon is seen in Rviz. Write down the x values provided in the terminal.

– Measurement: Measurements read from the terminal.

– Shut down: In case of any occurrences during the test execution that will temporarily suspend the test, repeat the user manual and setup steps to ensure the accuracy and consistency of the test results.

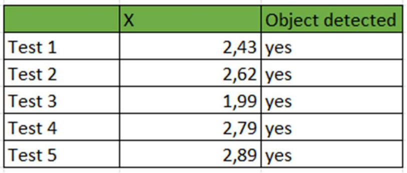
– Restart: Follow the user manual and setup steps.

– Stop: Follow the steps below “Avslutt LiDAR med objektdetektering” in this user manual:

– Wrap-up: The measurements are written down during the test. The results are then approved.

– Contingencies: In the case of any interruptions during the test execution that could potentially impact the accuracy of the measurements, the measurement readings will be repeated after the situation has resolved.

Procedure result:



	X	Object detected
Test 1	2,43	yes
Test 2	2,62	yes
Test 3	1,99	yes
Test 4	2,79	yes
Test 5	2,89	yes

The test was deemed successful and was approved. The object was detected 5 out of 5 times.

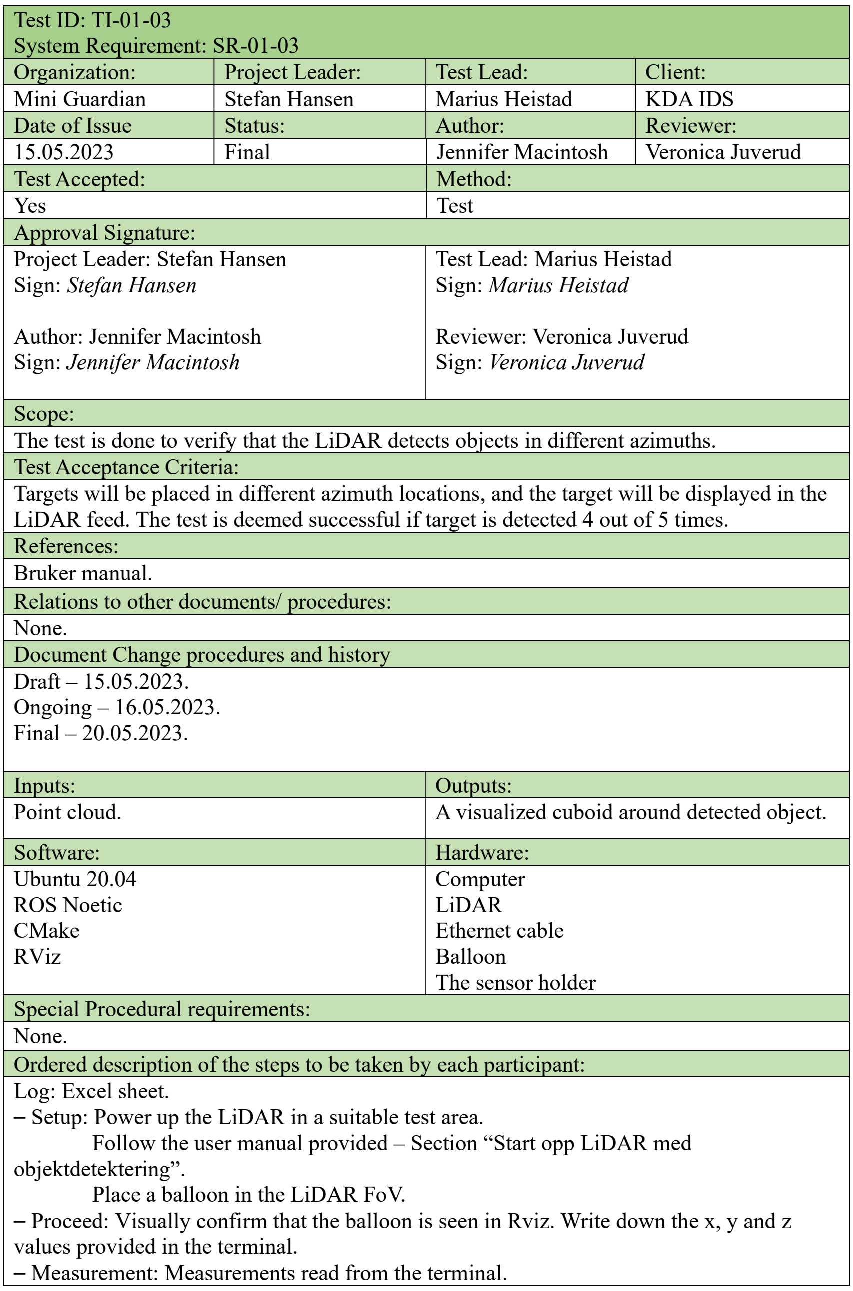
Environmental Information:

Dronesonen, 20:00, 20 degrees

Anomalous events:

None.

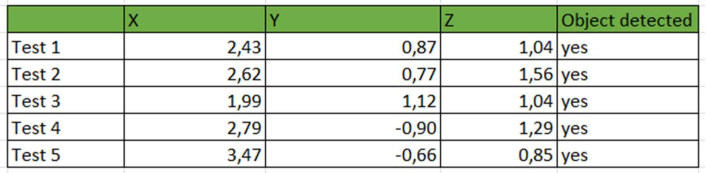
C Test report: TI-01-03



Test ID: TI-01-03 System Requirement: SR-01-03			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
15.05.2023	Final	Jennifer Macintosh	Veronica Juverud
Test Accepted:		Method:	
Yes		Test	
Approval Signature:			
Project Leader: Stefan Hansen Sign: *Stefan Hansen* Author: Jennifer Macintosh Sign: *Jennifer Macintosh*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Veronica Juverud Sign: *Veronica Juverud*	
Scope:			
The test is done to verify that the LiDAR detects objects in different azimuths.			
Test Acceptance Criteria:			
Targets will be placed in different azimuth locations, and the target will be displayed in the LiDAR feed. The test is deemed successful if target is detected 4 out of 5 times.			
References:			
Bruker manual.			
Relations to other documents/ procedures:			
None.			
Document Change procedures and history			
Draft – 15.05.2023. Ongoing – 16.05.2023. Final – 20.05.2023.			
Inputs:		Outputs:	
Point cloud.		A visualized cuboid around detected object.	
Software:		Hardware:	
Ubuntu 20.04 ROS Noetic CMake RViz		Computer LiDAR Ethernet cable Balloon The sensor holder	
Special Procedural requirements:			
None.			
Ordered description of the steps to be taken by each participant:			
Log: Excel sheet. – Setup: Power up the LiDAR in a suitable test area. Follow the user manual provided – Section “Start opp LiDAR med objektdetektering”. Place a balloon in the LiDAR FoV. – Proceed: Visually confirm that the balloon is seen in Rviz. Write down the x, y and z values provided in the terminal. – Measurement: Measurements read from the terminal.			

– Shut down: In case of any occurrences during the test execution that will temporarily suspend the test, repeat the user manual and setup steps to ensure the accuracy and consistency of the test results.
– Restart: Follow the user manual and setup steps.
– Stop: Follow the steps below "Avslutt LiDAR med objektdetektering" in this user manual:
– Wrap-up: The measurements are written down during the test. The results are then approved.
– Contingencies: In the case of any interruptions during the test execution that could potentially impact the accuracy of the measurements, the measurement readings will be repeated after the situation has resolved.

Procedure result:



	X	Y	Z	Object detected
Test 1	2,43	0,87	1,04	yes
Test 2	2,62	0,77	1,56	yes
Test 3	1,99	1,12	1,04	yes
Test 4	2,79	-0,90	1,29	yes
Test 5	3,47	-0,66	0,85	yes

The test was deemed successful. The object was detected 5 out of 5 times.

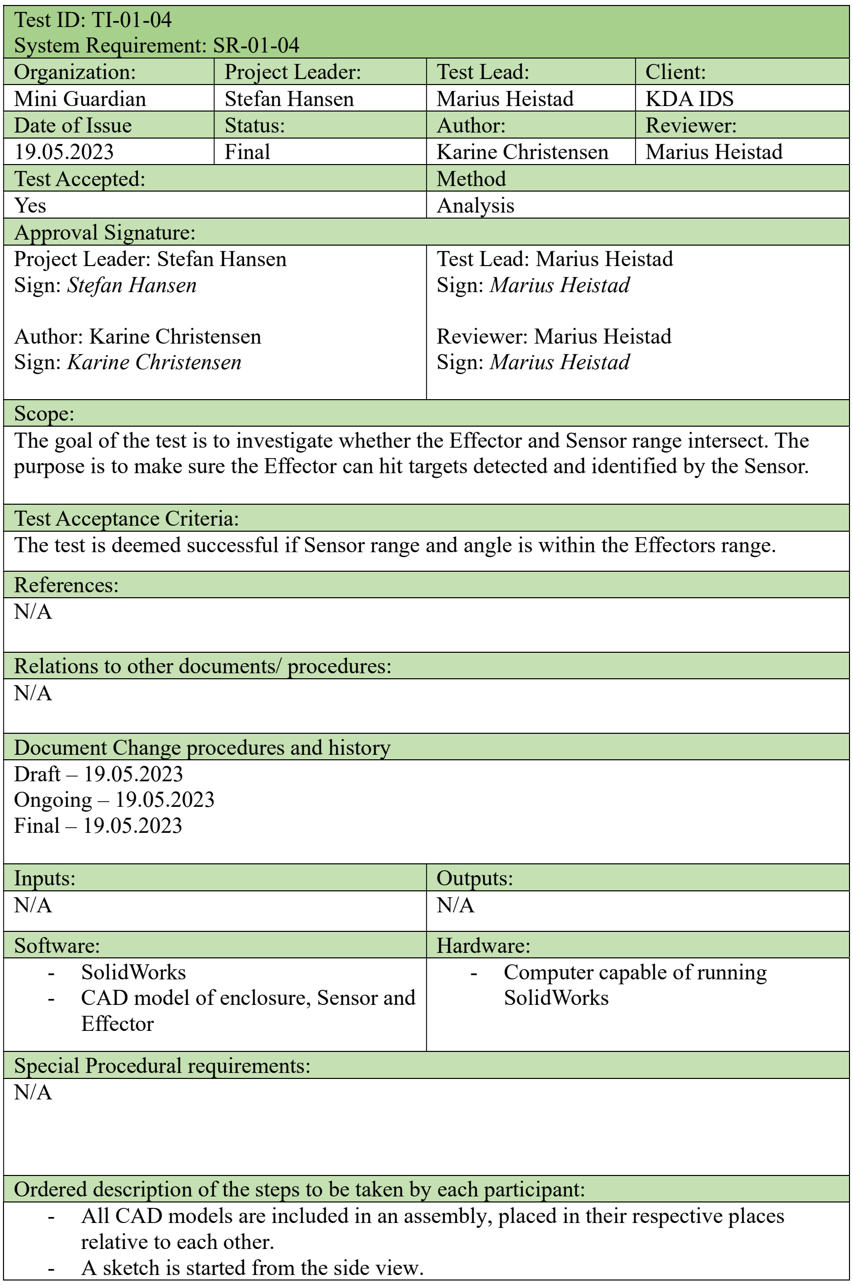
Environmental Information:

Dronesonen, 19:00, 20 degrees

Anomalous events:

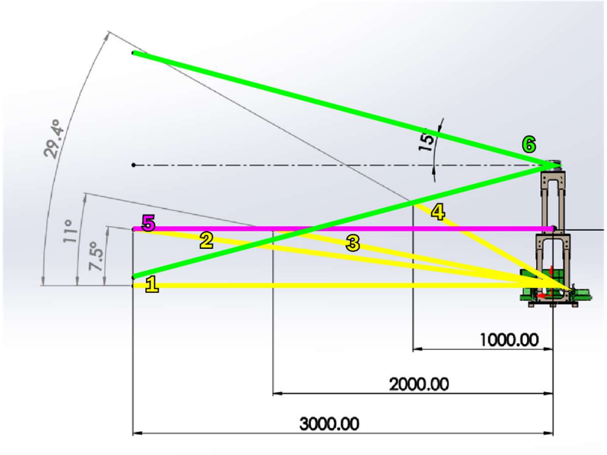
None.

D Test report: TI-01-04



Test ID: TI-01-04 System Requirement: SR-01-04			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
19.05.2023	Final	Karine Christensen	Marius Heistad
Test Accepted:		Method	
Yes		Analysis	
Approval Signature:			
Project Leader: Stefan Hansen Sign: *Stefan Hansen* Author: Karine Christensen Sign: *Karine Christensen*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Marius Heistad Sign: *Marius Heistad*	
Scope:			
The goal of the test is to investigate whether the Effector and Sensor range intersect. The purpose is to make sure the Effector can hit targets detected and identified by the Sensor.			
Test Acceptance Criteria:			
The test is deemed successful if Sensor range and angle is within the Effectors range.			
References:			
N/A			
Relations to other documents/ procedures:			
N/A			
Document Change procedures and history			
Draft – 19.05.2023 Ongoing – 19.05.2023 Final – 19.05.2023			
Inputs:		Outputs:	
N/A		N/A	
Software:		Hardware:	
- SolidWorks - CAD model of enclosure, Sensor and Effector		- Computer capable of running SolidWorks	
Special Procedural requirements:			
N/A			
Ordered description of the steps to be taken by each participant:			
- All CAD models are included in an assembly, placed in their respective places relative to each other. - A sketch is started from the side view.			

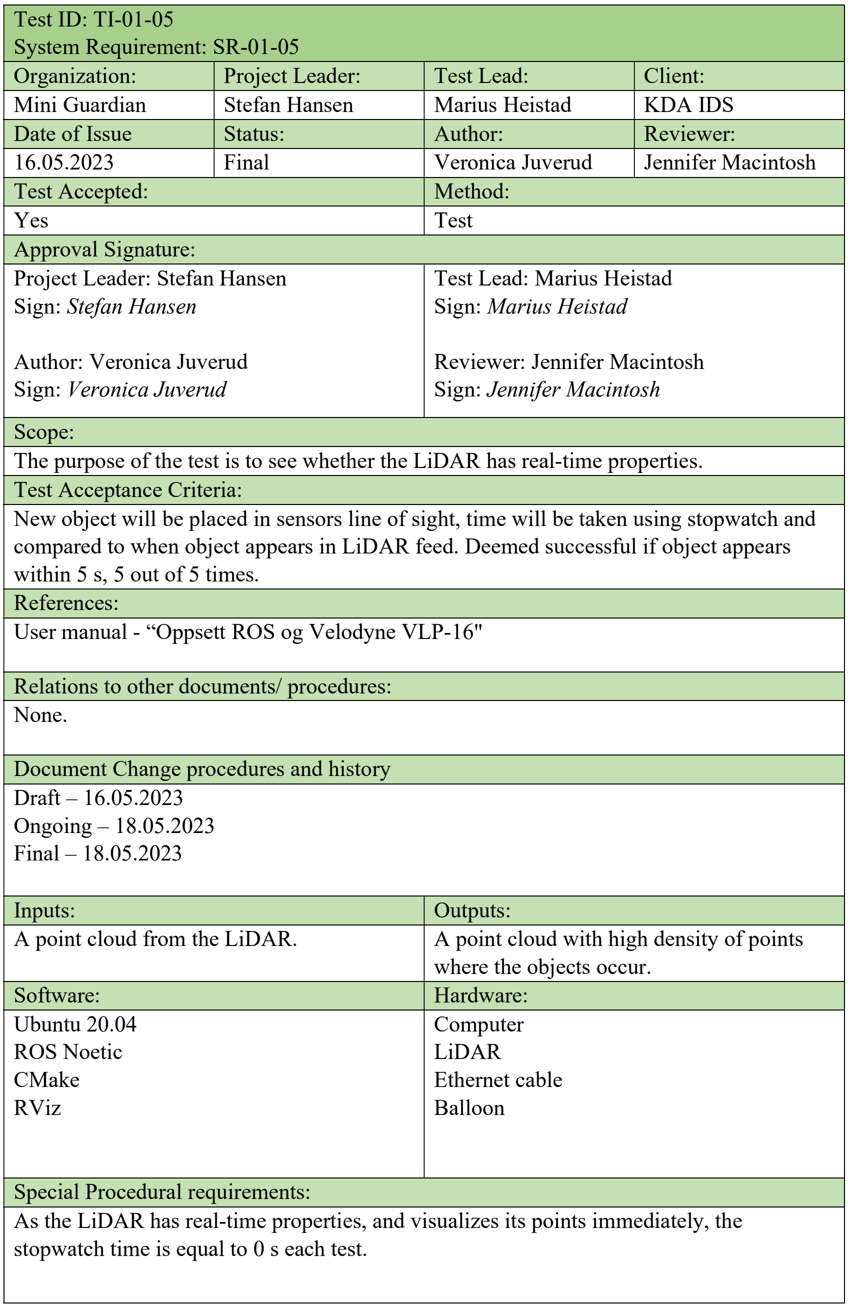
- The sketch includes lines describing the LiDAR FoV, the camera range, and the Effector range. Maximum distance is set to 3.0 m. At given distance and placing, the Effector range does not intersect with the Sensor at 0º. Since the Effector shall not fire at this angle, this position is disregarded. Lines to represent the Effector elevating are added. This shows that to hit a detected object at a given distance, the Effector must be elevated at these angles:
  - At 3.0 m, the Effector must elevate at least 7.5º to hit detected object.
  - At 2.0 m, the Effector must elevate at least 11º to hit detected object.
  - At 1.0 m, the Effector must elevate at least 29.4º to hit detected object.



- The figure illustrates the LiDAR FoV (green lines, 6), the camera range (pink line, 5) and Effector range elevated at different angles (yellow lines, 1,2,3 and 4)

Procedure result:
The test is deemed successful since the three ranges intersect at all investigated points relevant to hitting detected objects.
Environmental Information:
N/A
Anomalous events:
N/A

E Test report: TI-01-05

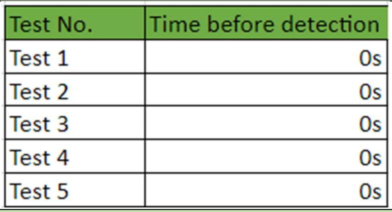


Test ID: TI-01-05 System Requirement: SR-01-05			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
16.05.2023	Final	Veronica Juverud	Jennifer Macintosh
Test Accepted:		Method:	
Yes		Test	
Approval Signature:			
Project Leader: Stefan Hansen Sign: *Stefan Hansen* Author: Veronica Juverud Sign: *Veronica Juverud*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Jennifer Macintosh Sign: *Jennifer Macintosh*	
Scope:			
The purpose of the test is to see whether the LiDAR has real-time properties.			
Test Acceptance Criteria:			
New object will be placed in sensors line of sight, time will be taken using stopwatch and compared to when object appears in LiDAR feed. Deemed successful if object appears within 5 s, 5 out of 5 times.			
References:			
User manual - “Oppsett ROS og Velodyne VLP-16"			
Relations to other documents/ procedures:			
None.			
Document Change procedures and history			
Draft – 16.05.2023 Ongoing – 18.05.2023 Final – 18.05.2023			
Inputs:		Outputs:	
A point cloud from the LiDAR.		A point cloud with high density of points where the objects occur.	
Software:		Hardware:	
Ubuntu 20.04 ROS Noetic CMake RViz		Computer LiDAR Ethernet cable Balloon	
Special Procedural requirements:			
As the LiDAR has real-time properties, and visualizes its points immediately, the stopwatch time is equal to 0 s each test.			

Ordered description of the steps to be taken by each participant:

– Log: An excel sheet, logging each test number and corresponding time.
– Setup: Follow user manual - “Oppsett ROS og Velodyne VLP-16", before displaying a balloon for it to appear in the LiDAR feed.
– Start: User manual - “Oppsett ROS og Velodyne VLP-16".
– Proceed: Inspecting the LiDAR feed simoultaneously as displaying a balloon and taking the time.
– Measurement: Stopwatch.
– Shut down: If the test is temporarily suspended due to unscheduled events, follow the user manual from “Oppsett av ROS og Velodyne VLP-16", “Vis frem LiDAR dataen i RViz”.
– Restart: Follow the user manual from “Oppsett av ROS og Velodyne VLP-16", “Vis frem LiDAR dataen i RViz”.
– Stop: Follow the user manual - “Avslutt LiDAR med objektdetektering”
– Wrap-up: The resulting data will be analyzed upon the acceptance criteria.
– Contingencies: Follow the user manual from “Oppsett av ROS og Velodyne VLP-16", “Vis frem LiDAR dataen i RViz”.

Procedure result:



Test No.	Time before detection
Test 1	0s
Test 2	0s
Test 3	0s
Test 4	0s
Test 5	0s

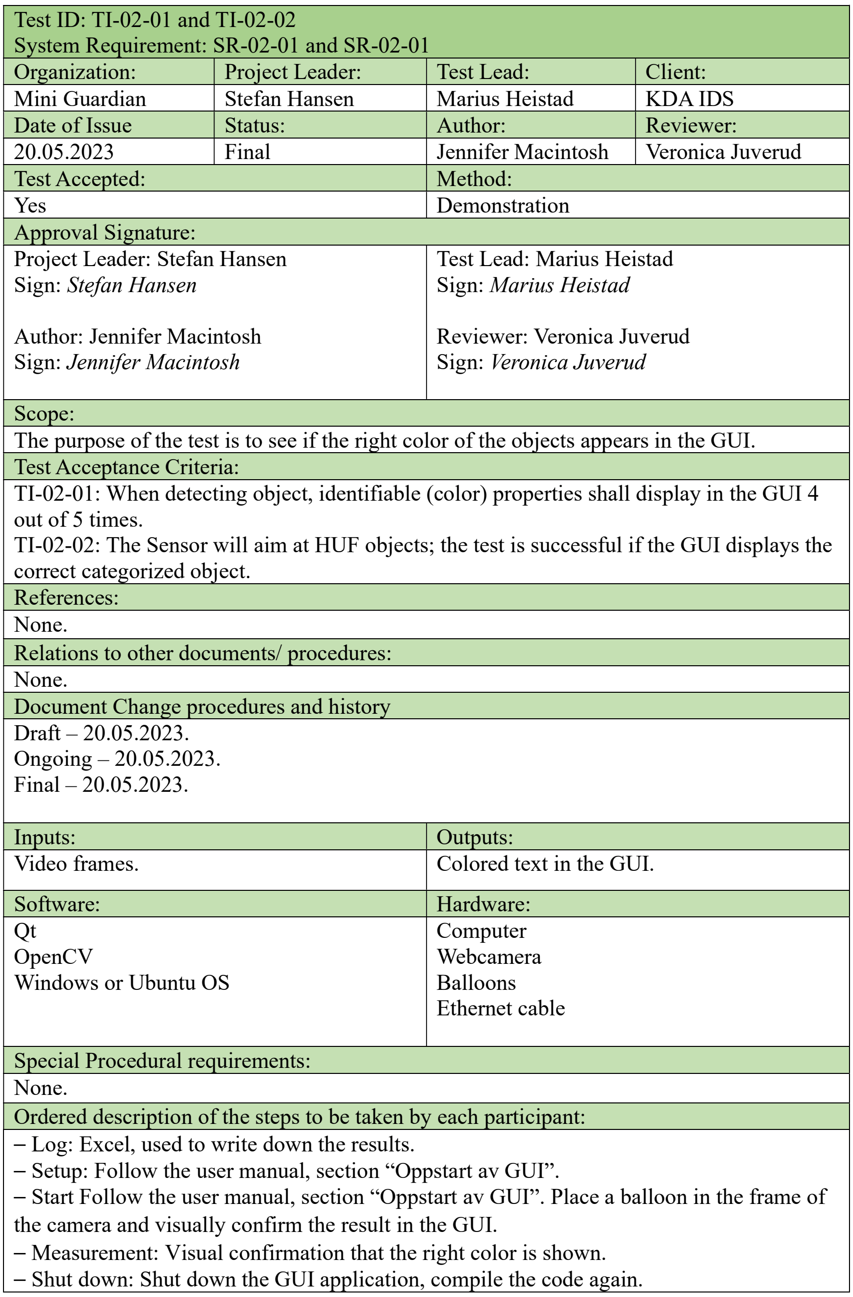
Environmental Information:

The test was performed at Innovasjonsloftet in room I-214 at 17:00. Sunshine outside, about 21 degrees.

Anomalous events:

None.

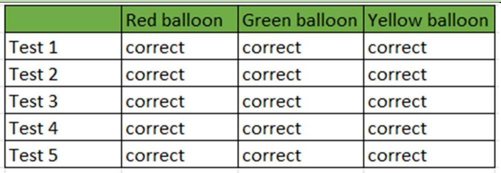
F Test report: TI-02-01 and TI-02-02



Test ID: TI-02-01 and TI-02-02 System Requirement: SR-02-01 and SR-02-01			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
20.05.2023	Final	Jennifer Macintosh	Veronica Juverud
Test Accepted:		Method:	
Yes		Demonstration	
Approval Signature:			
Project Leader: Stefan Hansen Sign: *Stefan Hansen* Author: Jennifer Macintosh Sign: *Jennifer Macintosh*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Veronica Juverud Sign: *Veronica Juverud*	
Scope:			
The purpose of the test is to see if the right color of the objects appears in the GUI.			
Test Acceptance Criteria:			
TI-02-01: When detecting object, identifiable (color) properties shall display in the GUI 4 out of 5 times. TI-02-02: The Sensor will aim at HUF objects; the test is successful if the GUI displays the correct categorized object.			
References:			
None.			
Relations to other documents/ procedures:			
None.			
Document Change procedures and history			
Draft – 20.05.2023. Ongoing – 20.05.2023. Final – 20.05.2023.			
Inputs:		Outputs:	
Video frames.		Colored text in the GUI.	
Software:		Hardware:	
Qt OpenCV Windows or Ubuntu OS		Computer Webcamera Balloons Ethernet cable	
Special Procedural requirements:			
None.			
Ordered description of the steps to be taken by each participant:			
– Log: Excel, used to write down the results. – Setup: Follow the user manual, section “Oppstart av GUI”. – Start Follow the user manual, section “Oppstart av GUI”. Place a balloon in the frame of the camera and visually confirm the result in the GUI. – Measurement: Visual confirmation that the right color is shown. – Shut down: Shut down the GUI application, compile the code again.			

– Restart: Shut down the GUI application, compile the code again.
– Stop: Shut down the GUI application.
– Contingencies: Unexpected events that can occur during the test execution is dealt with by restarting the test if necessary.

Procedure result:



	Red balloon	Green balloon	Yellow balloon
Test 1	correct	correct	correct
Test 2	correct	correct	correct
Test 3	correct	correct	correct
Test 4	correct	correct	correct
Test 5	correct	correct	correct

The test was successful. The algorithm detected the right color 5 out of 5 times.

Environmental Information:

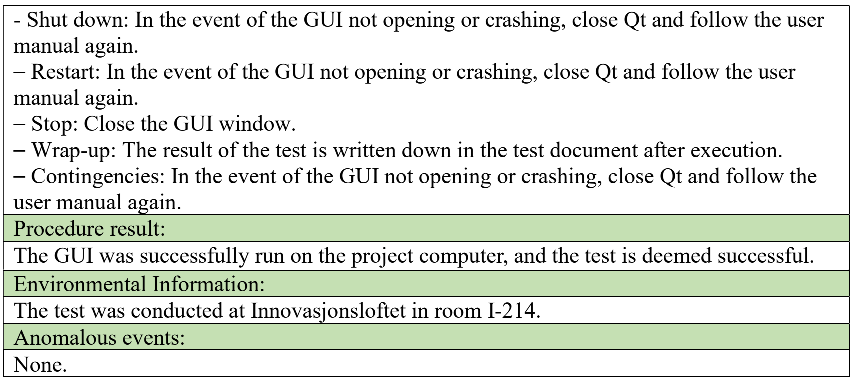
Our designated group room I-114, 20 degrees.

Anomalous events:

None.

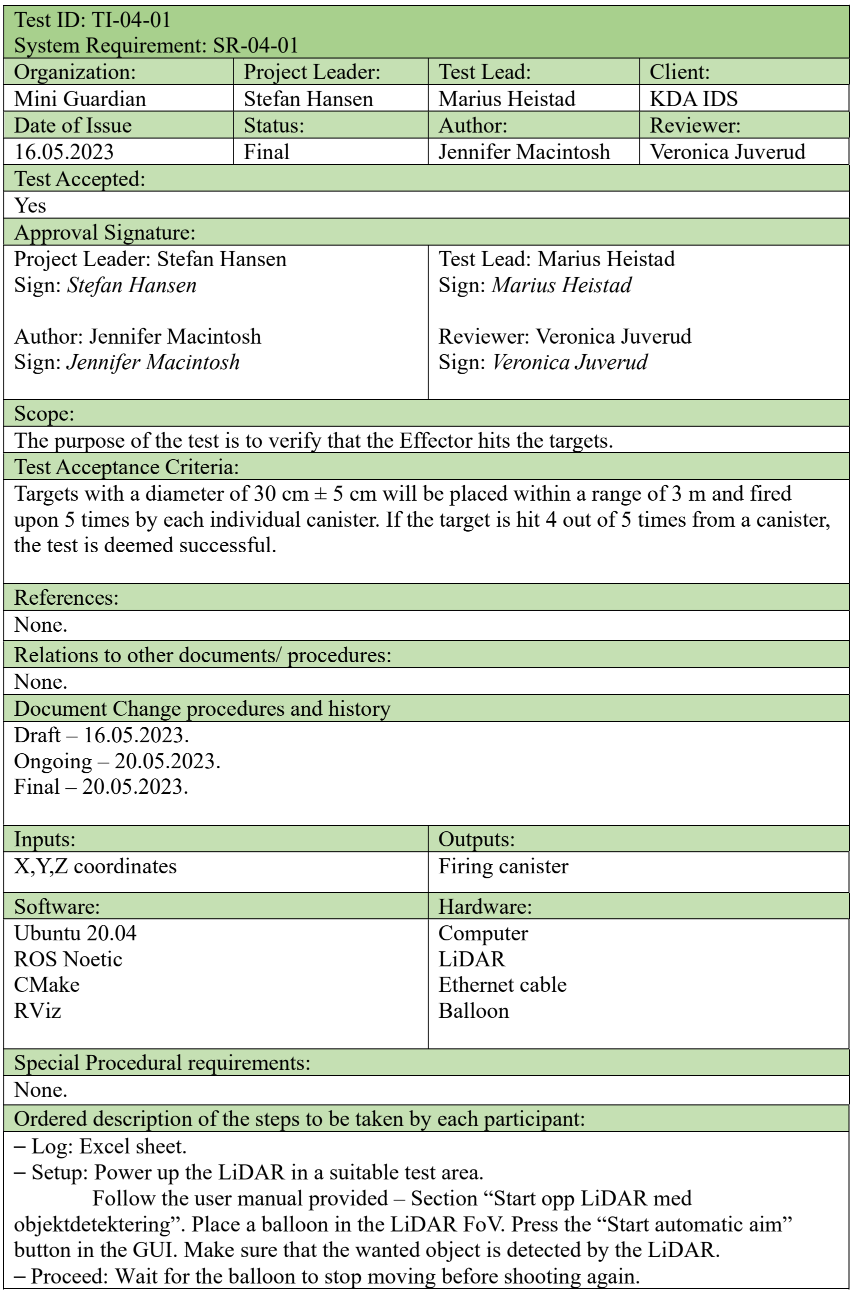
G Test report: TI-03-01

Test ID: TI-03-01 System Requirement: SR-03-01			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
18.05.2023	Final	Veronica Juverud	Jennifer Macintosh
Test Accepted:		Method:	
Yes		Demonstration	
Approval Signature:			
Project Leader: Stefan Hansen Sign: *Stefan Hansen* Author: Veronica Juverud Sign: *Veronica Juverud*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Jennifer Macintosh Sign: *Jennifer Macintosh*	
Scope:			
The test will assess whether the GUI can run on the project computer required for the project.			
Test Acceptance Criteria:			
An attempt to run the GUI on the project computer will be performed.			
References:			
User manual - "Oppstart av GUI"			
Relations to other documents/ procedures:			
User manual.			
Document Change procedures and history			
Draft – 18.05.2023 Ongoing – 18.05.2023 Final – 18.05.2023			
Inputs:		Outputs:	
A request to run the GUI		GUI window will open	
Software:		Hardware:	
Ubuntu 20.04 OpenCV Qt 5		Computer	
Special Procedural requirements:			
None.			
Ordered description of the steps to be taken by each participant:			
– Log: Writing in test document whether the GUI is running on project computer. – Setup: User manual - "Oppstart av GUI" – Proceed: Inspect whether the GUI is running on the project computer. - Measurement: No measurements taken.			



- Shut down: In the event of the GUI not opening or crashing, close Qt and follow the user manual again. – Restart: In the event of the GUI not opening or crashing, close Qt and follow the user manual again. – Stop: Close the GUI window. – Wrap-up: The result of the test is written down in the test document after execution. – Contingencies: In the event of the GUI not opening or crashing, close Qt and follow the user manual again.
Procedure result:
The GUI was successfully run on the project computer, and the test is deemed successful.
Environmental Information:
The test was conducted at Innovasjonsloftet in room I-214.
Anomalous events:
None.

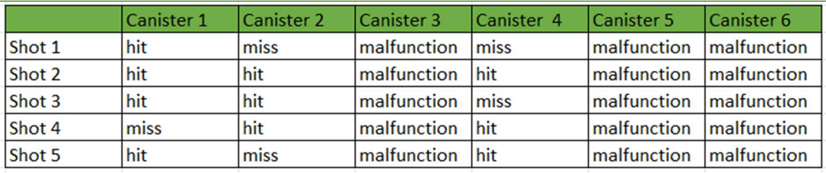
H Test report: TI-04-01



Test ID: TI-04-01 System Requirement: SR-04-01			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
16.05.2023	Final	Jennifer Macintosh	Veronica Juverud
Test Accepted:			
Yes			
Approval Signature:			
Project Leader: Stefan Hansen Sign: *Stefan Hansen* Author: Jennifer Macintosh Sign: *Jennifer Macintosh*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Veronica Juverud Sign: *Veronica Juverud*	
Scope:			
The purpose of the test is to verify that the Effector hits the targets.			
Test Acceptance Criteria:			
Targets with a diameter of 30 cm ± 5 cm will be placed within a range of 3 m and fired upon 5 times by each individual canister. If the target is hit 4 out of 5 times from a canister, the test is deemed successful.			
References:			
None.			
Relations to other documents/ procedures:			
None.			
Document Change procedures and history			
Draft – 16.05.2023. Ongoing – 20.05.2023. Final – 20.05.2023.			
Inputs:		Outputs:	
X,Y,Z coordinates		Firing canister	
Software:		Hardware:	
Ubuntu 20.04 ROS Noetic CMake RViz		Computer LiDAR Ethernet cable Balloon	
Special Procedural requirements:			
None.			
Ordered description of the steps to be taken by each participant:			
– Log: Excel sheet. – Setup: Power up the LiDAR in a suitable test area. Follow the user manual provided – Section “Start opp LiDAR med objektdetektering”. Place a balloon in the LiDAR FoV. Press the “Start automatic aim” button in the GUI. Make sure that the wanted object is detected by the LiDAR. – Proceed: Wait for the balloon to stop moving before shooting again.			

– Shut down: In case of any occurrences during the test execution that will temporarily suspend the test, repeat the user manual and setup steps to ensure the accuracy and consistency of the test results.
– Restart: Follow the user manual and setup steps.
– Stop: Follow the steps below "Avslutt LiDAR med objektdetektering" in the user manual.
– Wrap-up: The results are written down during the test.
– Contingencies: In the case of any interruptions during the test execution will be restarted.

Procedure result:



	Canister 1	Canister 2	Canister 3	Canister 4	Canister 5	Canister 6
Shot 1	hit	miss	malfunction	miss	malfunction	malfunction
Shot 2	hit	hit	malfunction	hit	malfunction	malfunction
Shot 3	hit	hit	malfunction	miss	malfunction	malfunction
Shot 4	miss	hit	malfunction	hit	malfunction	malfunction
Shot 5	hit	miss	malfunction	hit	malfunction	malfunction

The test was successful, a canister hit the target 4 out of 5 times.

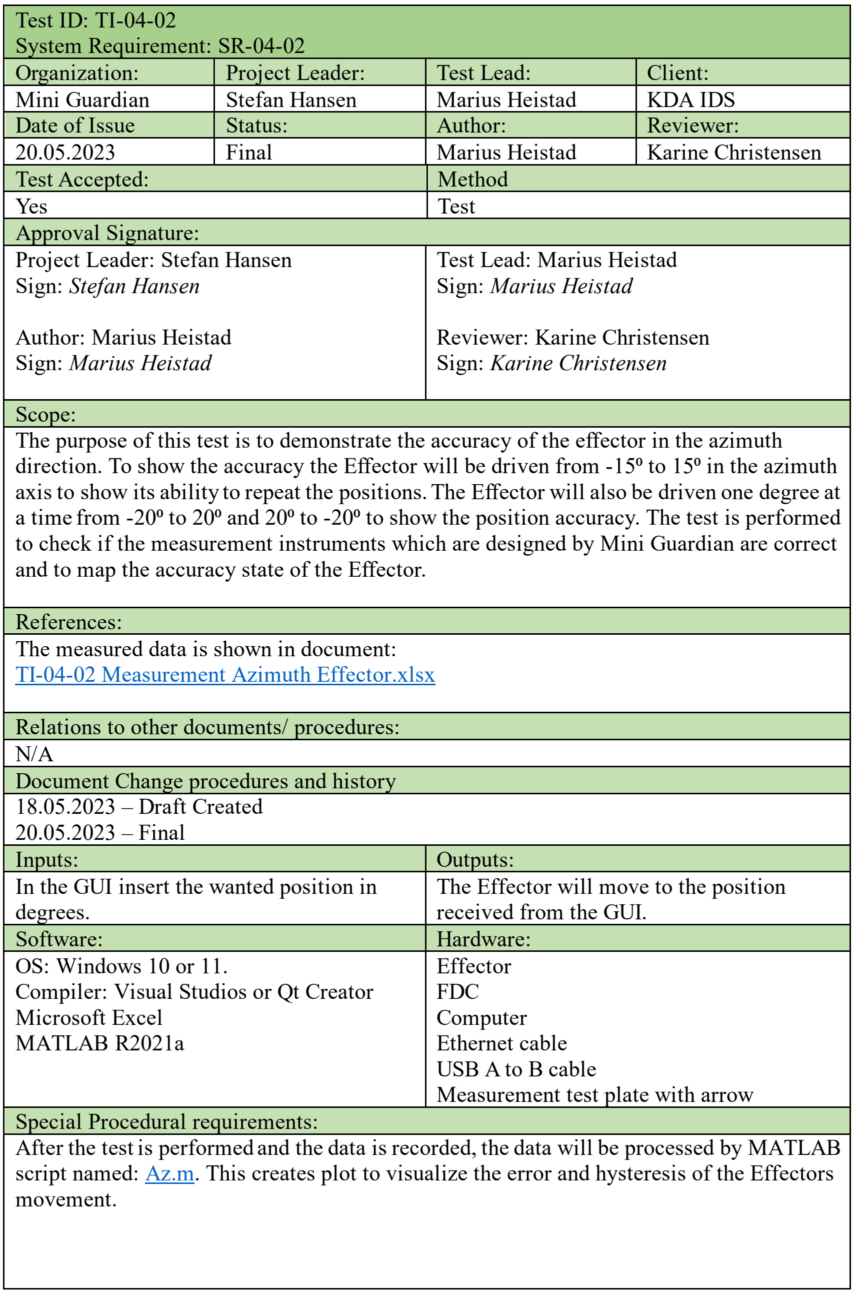
Environmental Information:

Dronesonen, 20 degrees, daylight

Anomalous events:

None.

I Test report: TI-04-02



Test ID: TI-04-02 System Requirement: SR-04-02			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
20.05.2023	Final	Marius Heistad	Karine Christensen
Test Accepted:		Method	
Yes		Test	
Approval Signature:			
Project Leader: Stefan Hansen Sign: *Stefan Hansen* Author: Marius Heistad Sign: *Marius Heistad*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Karine Christensen Sign: *Karine Christensen*	
Scope:			
The purpose of this test is to demonstrate the accuracy of the effector in the azimuth direction. To show the accuracy the Effector will be driven from -15º to 15º in the azimuth axis to show its ability to repeat the positions. The Effector will also be driven one degree at a time from -20º to 20º and 20º to -20º to show the position accuracy. The test is performed to check if the measurement instruments which are designed by Mini Guardian are correct and to map the accuracy state of the Effector.			
References:			
The measured data is shown in document: TI-04-02 Measurement Azimuth Effector.xlsx			
Relations to other documents/ procedures:			
N/A			
Document Change procedures and history			
18.05.2023 – Draft Created 20.05.2023 – Final			
Inputs:		Outputs:	
In the GUI insert the wanted position in degrees.		The Effector will move to the position received from the GUI.	
Software:		Hardware:	
OS: Windows 10 or 11. Compiler: Visual Studios or Qt Creator Microsoft Excel MATLAB R2021a		Effector FDC Computer Ethernet cable USB A to B cable Measurement test plate with arrow	
Special Procedural requirements:			
After the test is performed and the data is recorded, the data will be processed by MATLAB script named: Az.m. This creates plot to visualize the error and hysteresis of the Effectors movement.			

Ordered description of the steps to be taken by each participant:

The measurement test can be divided into two parts, they both use the same measurement instruments. However, the movement is different, one test is performed to measure the Effectors ability to repeat the position, and the other test is used to measure the point accuracy at given positions.

Start:

1. Before powering the system, the measurement plate needs to be assembled onto the front of the Effector together with the arrow.
2. After the measuring instrument is assembled correctly the arrow must be positioned close enough to read the measured position. This is done by moving the Effector using the elevation screw.
3. Remove the cover over the Arduino Mega, this is on the same side as the pneumatic and power is fed into the Effector.
4. The system is connected using the appropriate hardware, the software is launched. See the digital interface document for instructions: RD001 – Digital Interface.pdf
5. When the system is connected, and the software communicates correct, the testing can begin.

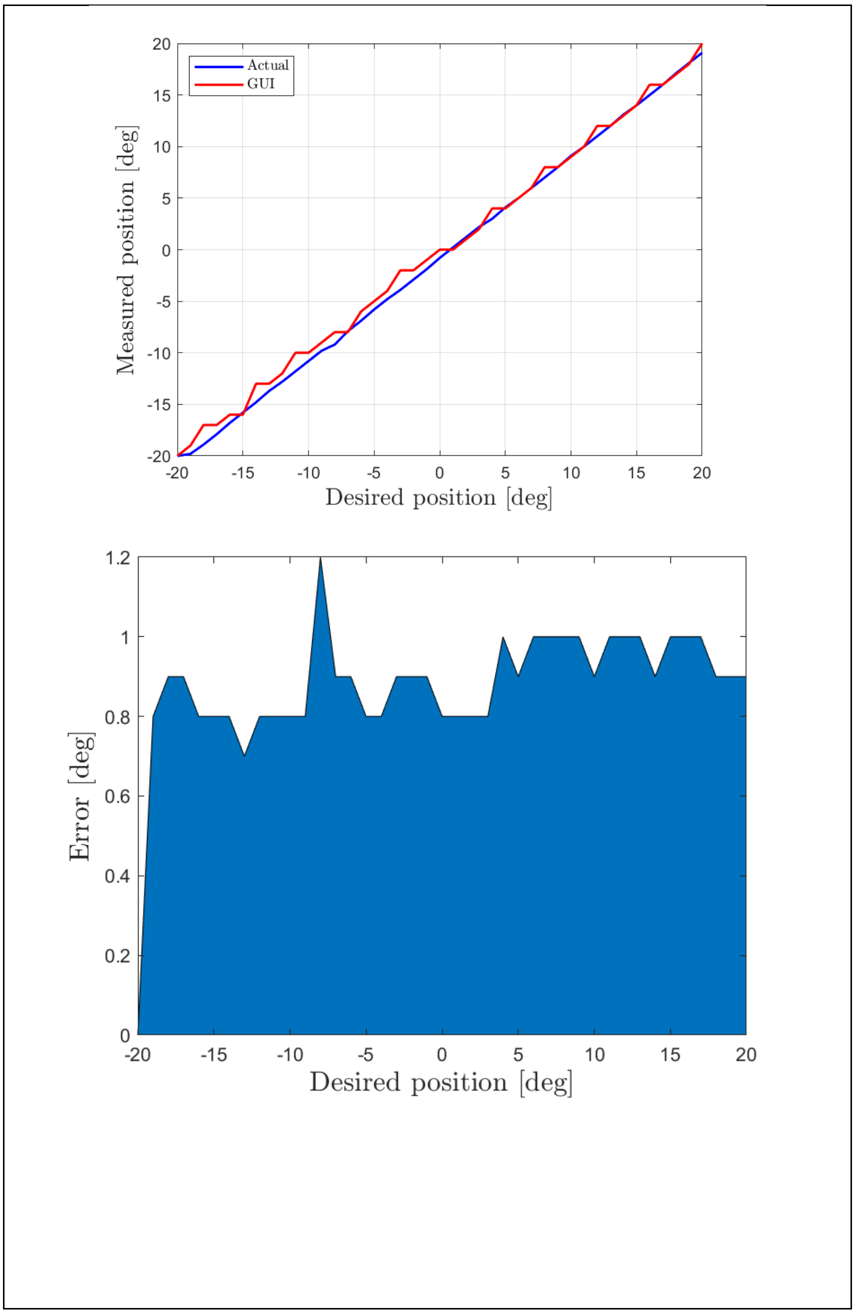
Proceed:

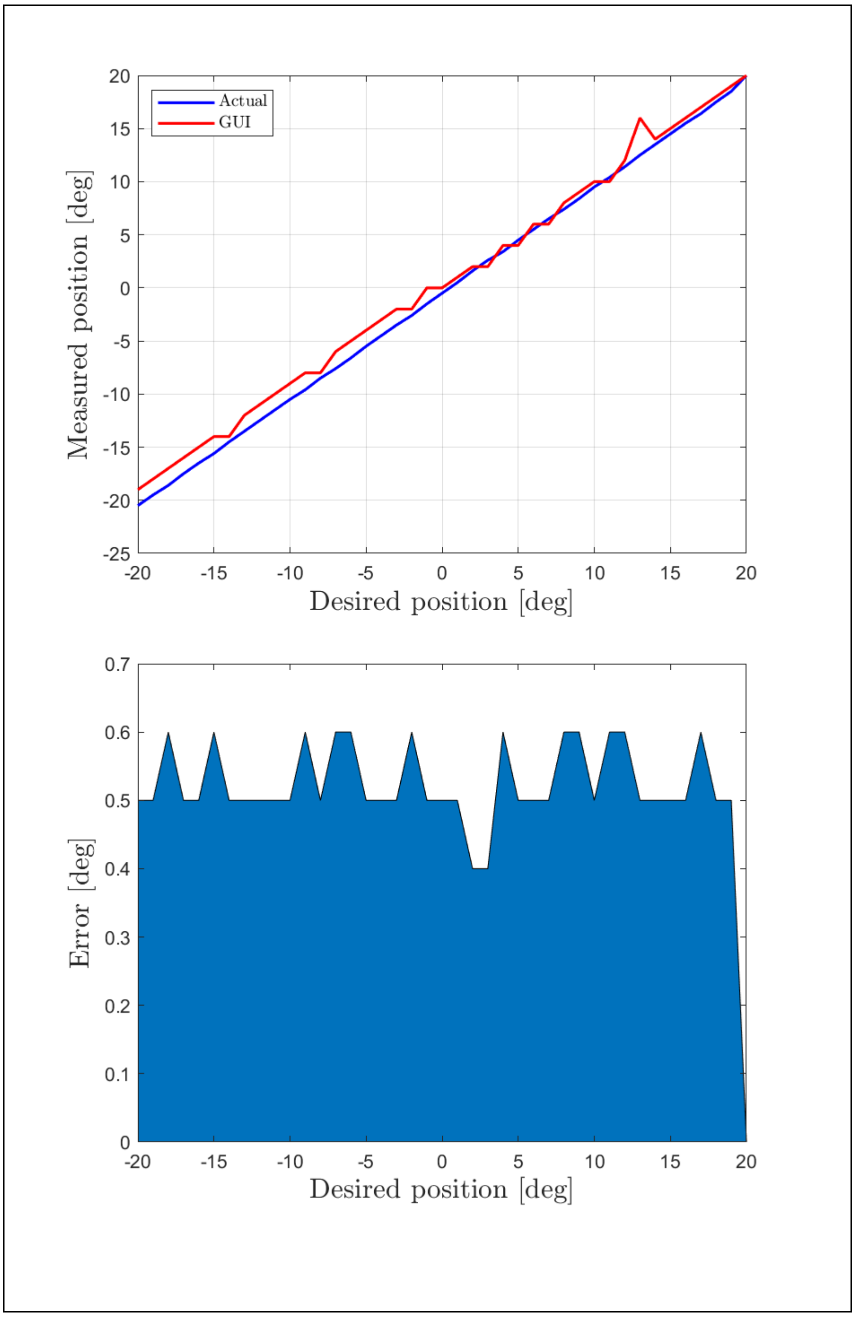
Repeatability test:
To measure the repeatability of the Effector, run calibration setting the Effector at starting position. Moving forward the Effector is then driven to 180º and then back to 0º, the deviation is then recorded in the spreadsheet. The measurement is done by seeing the degree the arrow points at on the measurement instrument. This test is repeated 5 times. When the 0º to 180º test is done a new repeatability test is preformed, the Effector is then moved from -15º to 15º, the deviation is recorded in the spreadsheet at each end point.

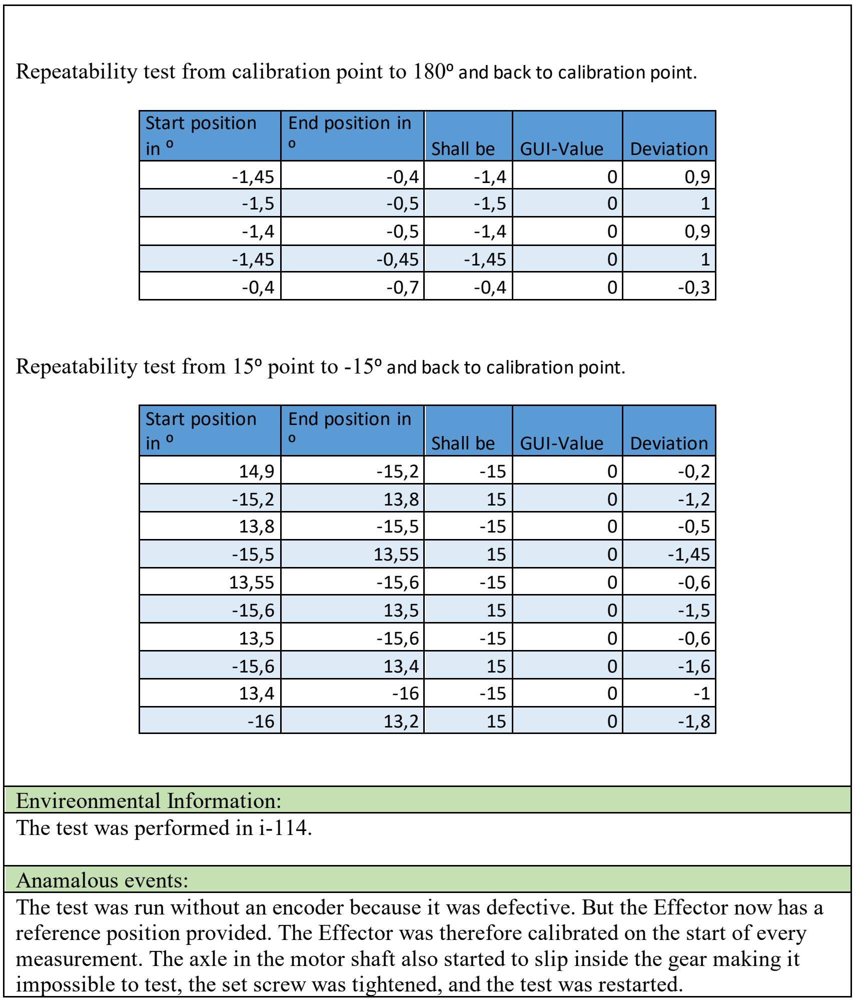
Position Accuracy:
To measure the position accuracy, the Effector is first calibrated to 0º. After this the Effector is first moved to -25º and then back to -20º. This is done to minimize the backlash effect from the belt and pulley system. The test is then performed, first the Effector is moved from the GUI to -19º, the deviation is recorded in excel sheet, together with position displayed in the GUI. Then it is moved +1º at a time until the final position, which is 20º, is reached. The test is then performed in the same order but from +20º to -20º.

Procedure result:

The captured data is recorded in a spreadsheet using Microsoft Excel, the result is then loaded and plotted using MATLAB. The results can be seen below; this shows that the accuracy of the Effector is inside of the requirements given by SR-04-02. The first pictures show the movement from -20º to 20º, and the last pictures shows the result when the Effector is moving from +20º to -20º.


Actual
GUI
Measured position [deg]
20
15
10
5
0
-5
-10
-15
-20
-20
-15
-10
-5
0
5
10
15
20
Desired position [deg]
Error [deg]
1.2
1
0.8
0.6
0.4
0.2
0
-20
-15
-10
-5
0
5
10
15
20
Desired position [deg]


Actual
GUI
Measured position [deg]
20
15
10
5
0
-5
-10
-15
-20
-25
-20
-15
-10
-5
0
5
10
15
20
Desired position [deg]
Error [deg]
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
-20
-15
-10
-5
0
5
10
15
20
Desired position [deg]



Repeatability test from calibration point to 180° and back to calibration point.

Start position in °	End position in °	Shall be	GUI-Value	Deviation
-1,45	-0,4	-1,4	0	0,9
-1,5	-0,5	-1,5	0	1
-1,4	-0,5	-1,4	0	0,9
-1,45	-0,45	-1,45	0	1
-0,4	-0,7	-0,4	0	-0,3

Repeatability test from 15° point to -15° and back to calibration point.

Start position in °	End position in °	Shall be	GUI-Value	Deviation
14,9	-15,2	-15	0	-0,2
-15,2	13,8	15	0	-1,2
13,8	-15,5	-15	0	-0,5
-15,5	13,55	15	0	-1,45
13,55	-15,6	-15	0	-0,6
-15,6	13,5	15	0	-1,5
13,5	-15,6	-15	0	-0,6
-15,6	13,4	15	0	-1,6
13,4	-16	-15	0	-1
-16	13,2	15	0	-1,8

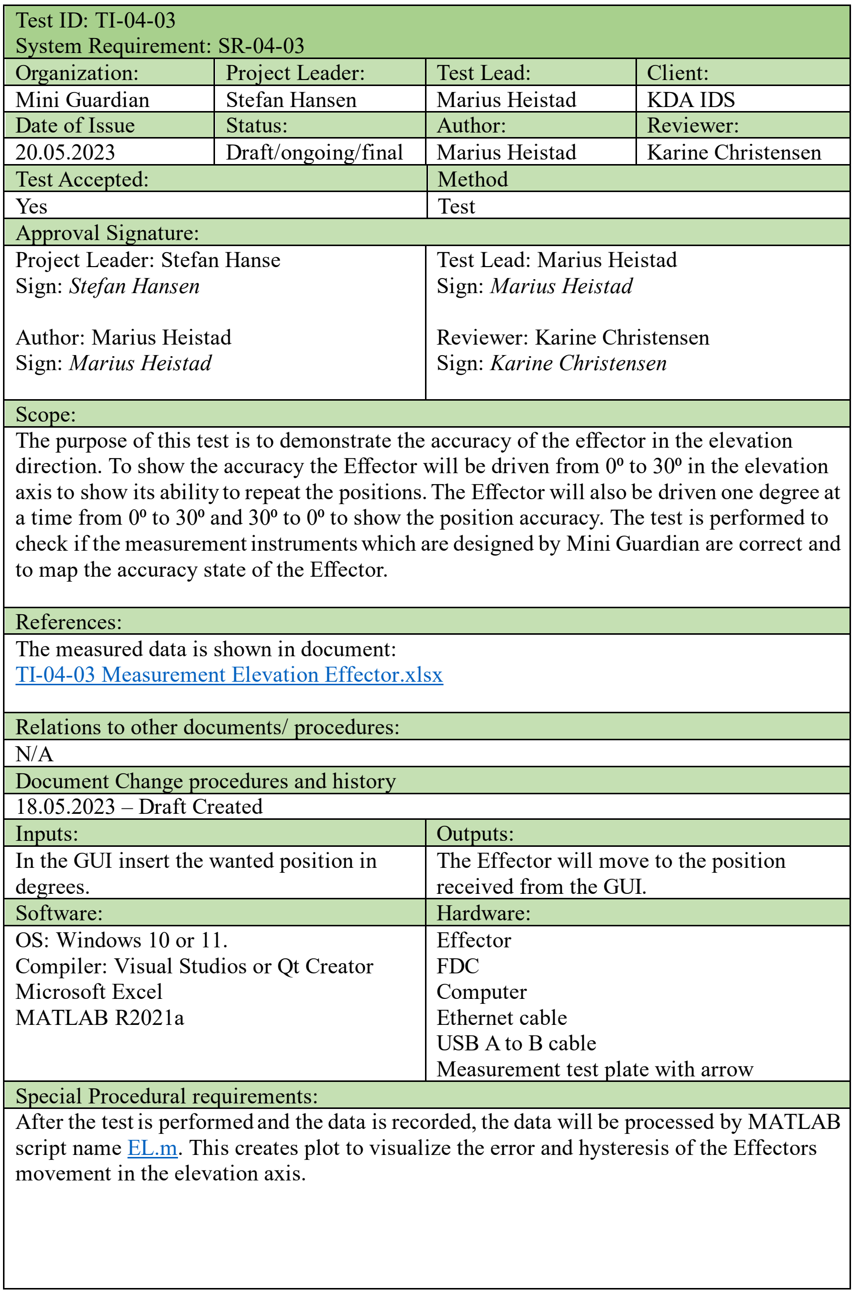
Envireonmental Information:

The test was performed in i-114.

Anamalous events:

The test was run without an encoder because it was defective. But the Effector now has a reference position provided. The Effector was therefore calibrated on the start of every measurement. The axle in the motor shaft also started to slip inside the gear making it impossible to test, the set screw was tightened, and the test was restarted.

J Test report: TI-04-03



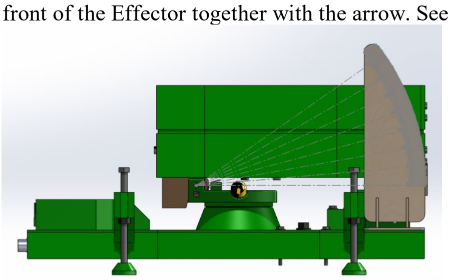
Test ID: TI-04-03 System Requirement: SR-04-03			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
20.05.2023	Draft/ongoing/final	Marius Heistad	Karine Christensen
Test Accepted:		Method	
Yes		Test	
Approval Signature:			
Project Leader: Stefan Hanse Sign: *Stefan Hansen* Author: Marius Heistad Sign: *Marius Heistad*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Karine Christensen Sign: *Karine Christensen*	
Scope:			
The purpose of this test is to demonstrate the accuracy of the effector in the elevation direction. To show the accuracy the Effector will be driven from 0º to 30º in the elevation axis to show its ability to repeat the positions. The Effector will also be driven one degree at a time from 0º to 30º and 30º to 0º to show the position accuracy. The test is performed to check if the measurement instruments which are designed by Mini Guardian are correct and to map the accuracy state of the Effector.			
References:			
The measured data is shown in document: TI-04-03 Measurement Elevation Effector.xlsx			
Relations to other documents/ procedures:			
N/A			
Document Change procedures and history			
18.05.2023 – Draft Created			
Inputs:		Outputs:	
In the GUI insert the wanted position in degrees.		The Effector will move to the position received from the GUI.	
Software:		Hardware:	
OS: Windows 10 or 11. Compiler: Visual Studios or Qt Creator Microsoft Excel MATLAB R2021a		Effector FDC Computer Ethernet cable USB A to B cable Measurement test plate with arrow	
Special Procedural requirements:			
After the test is performed and the data is recorded, the data will be processed by MATLAB script name EL.m. This creates plot to visualize the error and hysteresis of the Effectors movement in the elevation axis.			

Ordered description of the steps to be taken by each participant:

The measurement test can be divided into two parts, they both use the same measurement instruments. However, the movement is different, one test is performed to measure the Effectors ability to repeat the position, and the other test is used to measure the point accuracy at given positions.

Start:

1. Before powering the system, the measurement plate needs to be assembled onto the front of the Effector together with the arrow. See picture for reference:



2. After the measuring instrument is assembled correctly the arrow must be positioned close enough to read the measured position. This is done by moving the Effector using the elevation screw.
3. Remove the cover over the Arduino Mega, this is on the same side as the pneumatic and power is fed into the Effector.
4. The system is connected using the appropriate hardware, the software is launched. See the digital interface document for instructions: RD001 – Digital Interface.pdf
5. When the system is connected, and the software communicates correct, the testing can begin.

Proceed:

Repeatability test:

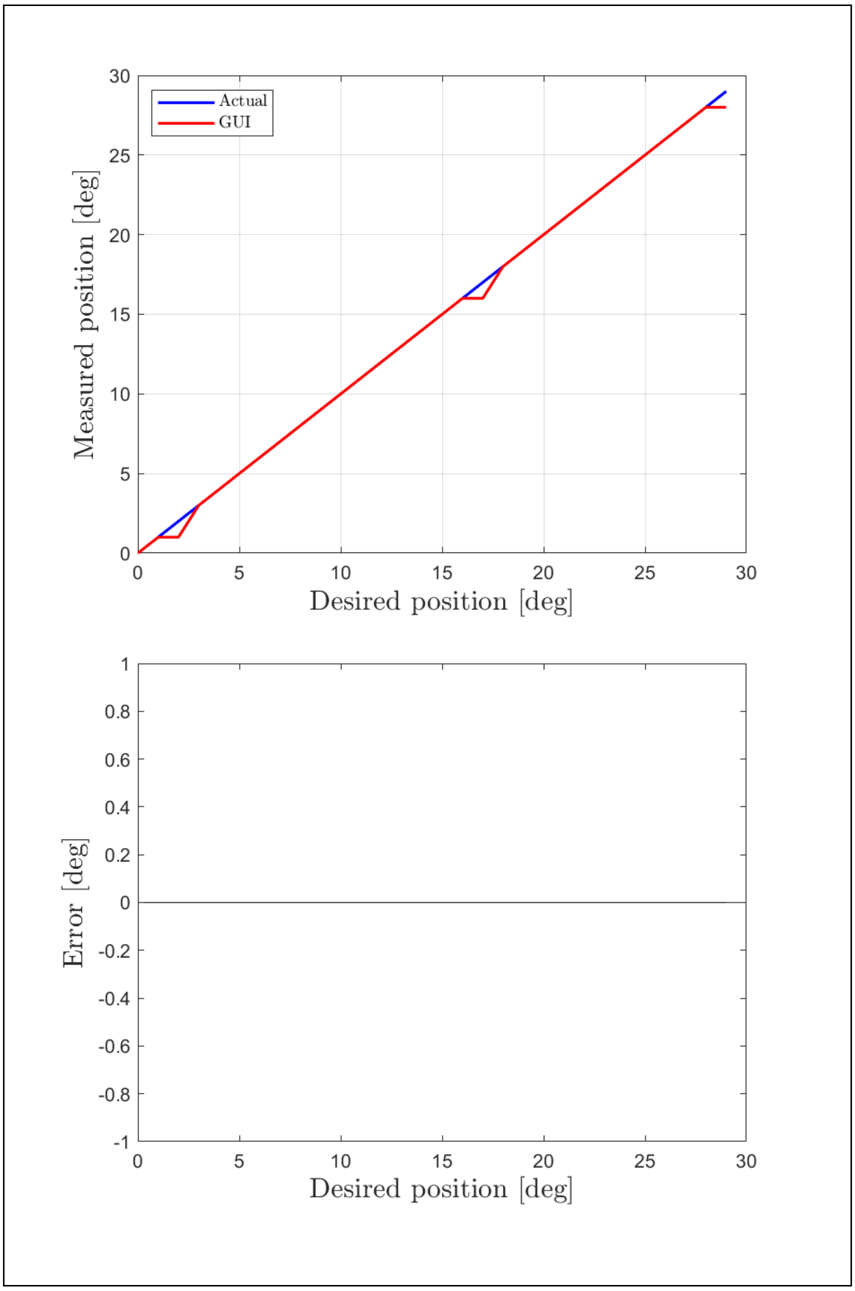
To measure the repeatability of the Effector move it to 0º and press the red reset button on the Arduino Mega. Moving forward the Effector is then driven to 30º and then back to 0º, the deviation is then recorded in the spreadsheet. The measurement is done by seeing the degree the arrow points at on the measurement instrument. This test is repeated 10 times.

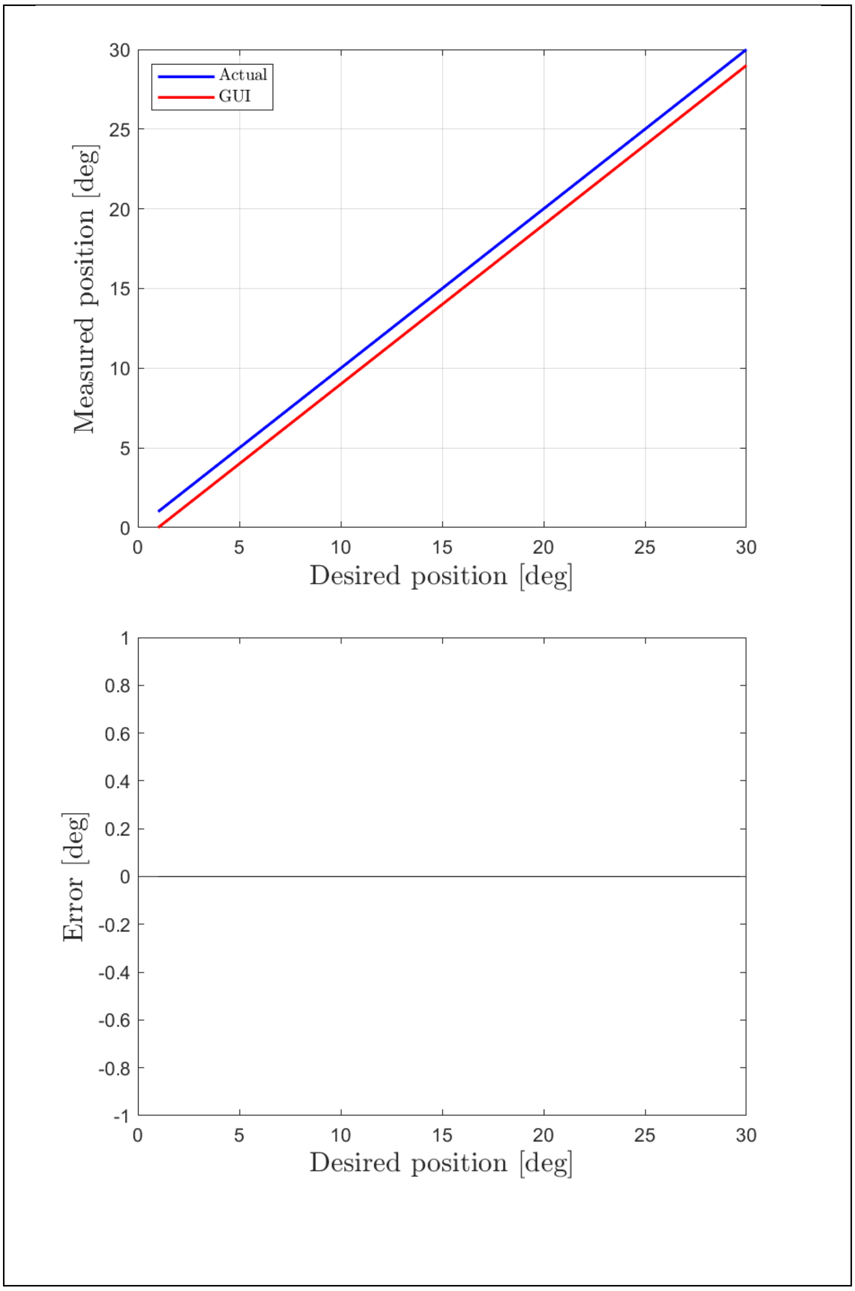
Position Accuracy:

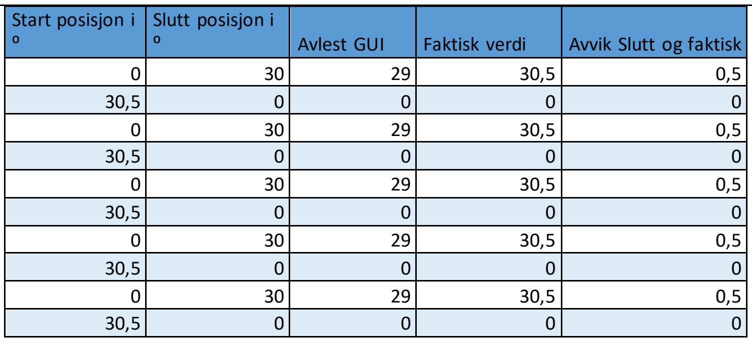
To measure the position accuracy, the Effector is first moved to 0º on the measuring plate. The Arduino Mega is then reset by pressing the red button, after the position is reset, the Effector is then moved upwards from 0º to 30º, 1º at a time. The deviation is recorded on every position and compared with the values given by the GUI and the desired position. When the movement from 0º to 30º is finished another run is made from 30º to 0º, the procedure is the same as before.

Procedure result:

The captured data is recorded in a spreadsheet using Microsoft Excel, the result is then loaded and plotted using MATLAB. The first pictures show the movement from 0º to 30º, and the last pictures shows the result when the Effector is moving from 30º to 0º. The results show that the Effector Is accurate enough for the test to be deemed successful and inside of the requirements given by SR-04-03.


Actual
GUI
Measured position [deg]
Desired position [deg]
0
5
10
15
20
25
30
Error [deg]
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1


Actual
GUI
Measured position [deg]
Desired position [deg]
0
5
10
15
20
25
30
Error [deg]
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
Desired position [deg]



Start posisjon i º	Slutt posisjon i º	Avlest GUI	Faktisk verdi	Avvik Slutt og faktisk
0	30	29	30,5	0,5
30,5	0	0	0	0
0	30	29	30,5	0,5
30,5	0	0	0	0
0	30	29	30,5	0,5
30,5	0	0	0	0
0	30	29	30,5	0,5
30,5	0	0	0	0
0	30	29	30,5	0,5
30,5	0	0	0	0

Envireonmental Information:

The test was performed in i-114 at normal room temperature. Some vibrations from the table made it so that the Effector was swaying a tiny amount when standing still.

Anamalous events:

Some vibrations from the table made it so that the Effector was swaying a tiny amount when standing still. This had little effect on the measurements as we waited until the Effector had stopped swaying before the deviation was recorded.

K Test report: TI-05-01

Test ID: TI-05-01 System Requirement: SR-05-01			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
21.05.2023	Final	Veronica Juverud	Jennifer Macintosh
Test Accepted:		Method:	
Yes		Test	
Approval Signature:			
Project Leader: Stefan Hansen Sign: *Stefan Hansen* Author: Veronica Juverud Sign: *Veronica Juverud*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Jennifer Macintosh Sign: *Jennifer Macintosh*	
Scope:			
The purpose of the test is to ensure that the Effector does not trigger upon non-hostile targets.			
Test Acceptance Criteria:			
In a safe environment, we will attempt to trigger the Effector when the target is categorized as friendly or unknown. The test is deemed successful if action is prohibited 5 out of 5 times.			
References:			
User manual.			
Relations to other documents/ procedures:			
User manual.			
Document Change procedures and history			
Draft – 21.05.2023 Ongoing – 21.05.2023 Final – 21.05.2023			
Inputs:		Outputs:	
Serial status from Arduino Nano		Canister not firing projectiles	
Software:		Hardware:	
Ubuntu 20.04 ROS Noetic Qt OpenCV		Computer Effector LiDAR AXIS camera Ethernet cables Power Supply	
Special Procedural requirements:			
None.			
Ordered description of the steps to be taken by each participant:			
– Log: Result of each attempt is written in a table in this test document.			

– Setup: Follow the user manual - “Forberedelser før bruk av Mini Guardian med avfyring”.
– Start: Follow the user manual - “Forberedelser før bruk av Mini Guardian med avfyring”.
– Proceed: The operator needs to attempt pressing the “Fire” buttons while the camera has identified an object as Friendly or Unknown.
– Measurement: No measurements will be made.
– Shut down: If the test is temporarily suspended due to unscheduled events, restart the GUI.
– Restart: For the test to be restarted at certain points, restart the GUI. If needed, follow the user manual, section “Forberedelser før bruk av Mini Guardian med avfyring” again.
– Stop: To stop the test in an ordely fashion, follow the user manual, section “Avslutningsprosedyre for systemet”.
– Wrap-up: After the test is finished, the results are evaluated.
– Contingencies: If unexpected events occur during the test execution, restart the GUI if necessary or follow the user manual, section “Avslutningsprosedyre for systemet” if shutdown is required.

Procedure result:

Attempt to fire upon non-hostile target	**Action prohibited?**
1	Yes
2	Yes
3	Yes
4	Yes
5	Yes

Environmental Information:

The test was done at Innovasjonsloftet in room I-214, 14:00, sunshine outside, approximately 21 degrees.

Anomalous events:

None.

L Test report: TI-05-02

Test ID: TI-05-02 System Requirement: SR-05-02			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
21.05.2023	Final	Veronica Juverud	Jennifer Macintosh
Test Accepted:			
Failed			
Approval Signature:			
Project Leader: Stefan Hansen Sign: *Stefan Hansen* Author: Veronica Juverud Sign: *Veronica Juverud*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Jennifer Macintosh Sign: *Jennifer Macintosh*	
Scope:			
The purpose of the test report is to see whether the log in the GUI reports attempts of firing on non-hostile objects.			
Test Acceptance Criteria:			
Attempts to fire on non-hostile targets will appear in the GUI, deemed successful if the attempts appear in GUI 10 out of 10 times.			
References:			
User manual.			
Relations to other documents/ procedures:			
User manual.			
Document Change procedures and history			
Draft – 21.05.2023 Ongoing – 21.05.2023 Final – 21.05.2023			
Inputs:		Outputs:	
Serial status from Arduino Nano		GUI displaying attempt to fire upon non-hostile target	
Software:		Hardware:	
Ubuntu 20.04 ROS Noetic Qt OpenCV		Computer Effector LiDAR AXIS camera Ethernet cables Power Supply	
Special Procedural requirements:			
None.			
Ordered description of the steps to be taken by each participant:			
– Log: Result of each attempt is written in a table in this test document. – Setup: Follow the user manual - “Forberedelser før bruk av Mini Guardian med avfyring”. – Start: Follow the user manual - “Forberedelser før bruk av Mini Guardian med			

avfyring”.
– Proceed: The operator needs to press the “Fire” buttons while the camera has identified an object as Friendly or Unknown.
– Measurement: No measurements will be made.
– Shut down: If the test is temporarily suspended due to unscheduled events, restart the GUI.
– Restart: For the test to be restarted at certain points, restart the GUI. If needed, follow the user manual, section “Forberedelser før bruk av Mini Guardian med avfyring” again.
– Stop: To stop the test in an ordely fashion, follow the user manual, section “Avslutningsprosedyre for systemet”.
– Wrap-up: After the test is finished, the results are evaluated.
– Contingencies: If unexpected events occur during the test execution, restart the GUI if necessary or follow the user manual, section “Avslutningsprosedyre for systemet” if shutdown is required.

Procedure result:

Attempt to fire upon non-hostile target	**Action displayed in GUI?**
1	No
2	No
3	No
4	No
5	No
6	No
7	No
8	No
9	No
10	No

As the buttons for firing projectiles are deactivated when Friendly or Unknown objects are detected, there is no option to press them. This results in no way of knowing when the buttons were pressed. The test is therefore deemed failed.

Environmental Information:

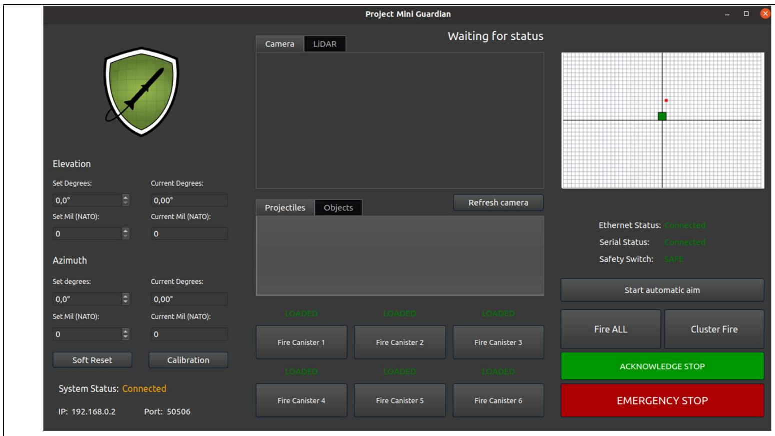
The test was done at Innovasjonsloftet in room I-214, 14:00, sunshine outside, approximately 21 degrees.

Anomalous events:

None.

M Test report: TI-06-01, TI-06-02, TI-06-03, TI-06-04

Test ID: TI-06-01, TI-06-02, TI-06-03, TI-06-04 System Requirement: SR-06-01, SR-06-02, SR-06-03, SR-06-04			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
20.05.2023	Final	Jennifer Macintosh	Jennifer Macintosh
Test Accepted:		Method:	
Yes		Demonstration	
Approval Signature:			
Project Leader: Stefan Hansen Sign: *Stefan Hansen* Author: Jennifer Macintosh Sign: *Jennifer Macintosh*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Veronica Juverud Sign: *Veronica Juverud*	
Scope:			
The purpose of this test is to make sure that the different statuses show in the GUI.			
Test Acceptance Criteria:			
Demonstrate that status displayed on the GUI corresponds with the actual status of the effector.			
References:			
None.			
Relations to other documents/ procedures:			
None.			
Document Change procedures and history			
Draft – 20.05.2023. Ongoing – 20.05.2023. Final – 20.05.2023.			
Inputs:		Outputs:	
Serial, and Ethernet signals.		Display status text in the GUI.	
Software:		Hardware:	
Ubuntu 20.04 or Windows Qt		Computer Effector	
Special Procedural requirements:			
None.			
Ordered description of the steps to be taken by each participant:			
– Setup: Start the computer. – Start: Start Qt. – Proceed: Compile the code. Read the status in the GUI. – Shut down: Compile the code again, restart the test. – Restart: Compile the code again to restart the GUI. – Stop: Shut down the GUI.			
Procedure result:			



Project Mini Guardian

Waiting for status

Elevation

Set Degrees: 0,0°

Current Degrees: 0,00°

Set Mil (NATO): 0

Current Mil (NATO): 0

Azimuth

Set degrees: 0,0°

Current Degrees: 0,00°

Set Mil (NATO): 0

Current Mil (NATO): 0

System Status: Connected

IP: 192.168.0.2 Port: 50506

Ethernet Status: Connected

Serial Status: Connected

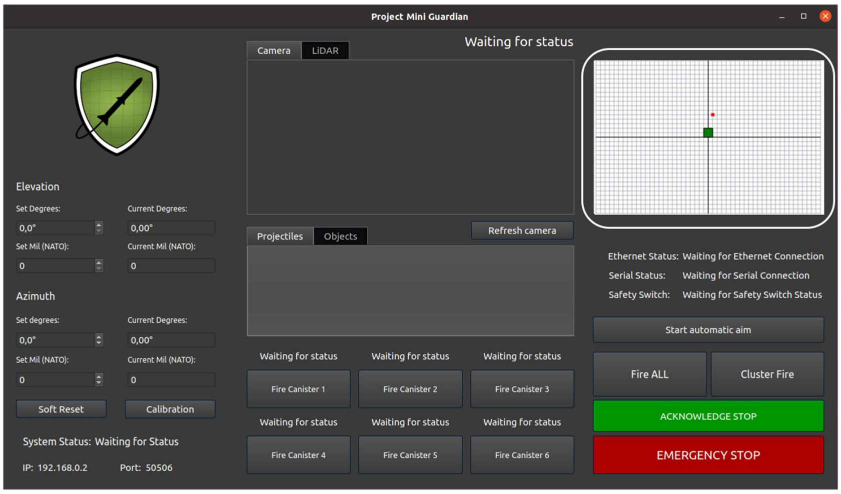
Safety Switch: SAFE

The test was deemed successful, all the statuses show the correct information.

Environmental Information:
Grouproom I-114, 23 degrees, no daylight.
Anomalous events:
The Arduino mega has to be reset sometimes and the GUI restarted.

N Test report: TI-07-01

Test ID: TI-07-01 System Requirement: SR-07-01			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
21.05.2023	Final	Jennifer Macintosh	Karine Christensen
Test Accepted:		Method:	
Yes		Demonstration	
Approval Signature:			
Project Leader: Stefan Hansen Sign: *Stefan Hansen* Author: Jennifer Macintosh Sign: *Jennifer Macintosh*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Karine Christensen Sign: *Karine Christensen*	
Scope:			
The purpose of this test is to verify that the Effector knows where the sensor is located.			
Test Acceptance Criteria:			
If information about locations appears in the GUI the test is deemed successful.			
References:			
None.			
Relations to other documents/ procedures:			
User Manual.			
Document Change procedures and history			
Draft – 21.05.2023. Ongoing – 21.05.2023. Final – 21.05.2023.			
Inputs:		Outputs:	
None.		Widget that displays the location.	
Software:		Hardware:	
Ubuntu or Windows OS Qt		Computer	
Special Procedural requirements:			
None.			
Ordered description of the steps to be taken by each participant:			
– Log: The test is done once, result written in test document. – Setup: Start the computer, start Qt. – Start: Compile the code in Qt. – Proceed: Visually inspect the widget with the placement. – Shut down: Compile the code again. – Restart: Compile the code again. – Stop: Shut down the GUI, exit Qt.			
Procedure result:			



System Status: Waiting for Status

IP: 192.168.0.2 Port: 50506

Ethernet Status: Waiting for Ethernet Connection

Serial Status: Waiting for Serial Connection

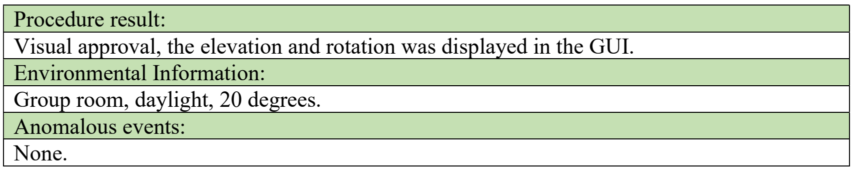
Safety Switch: Waiting for Safety Switch Status

Visual approval, the location is displayed in the GUI. The Effector, LiDAR and Camera is place on the same z-axis, the 2D coordinate system shows a green rectangle where the Effector is placed. The LiDAR and Camera is placed on top of it.

Environmental Information:
Group room, daylight, 20 degrees.
Anomalous events:
None.

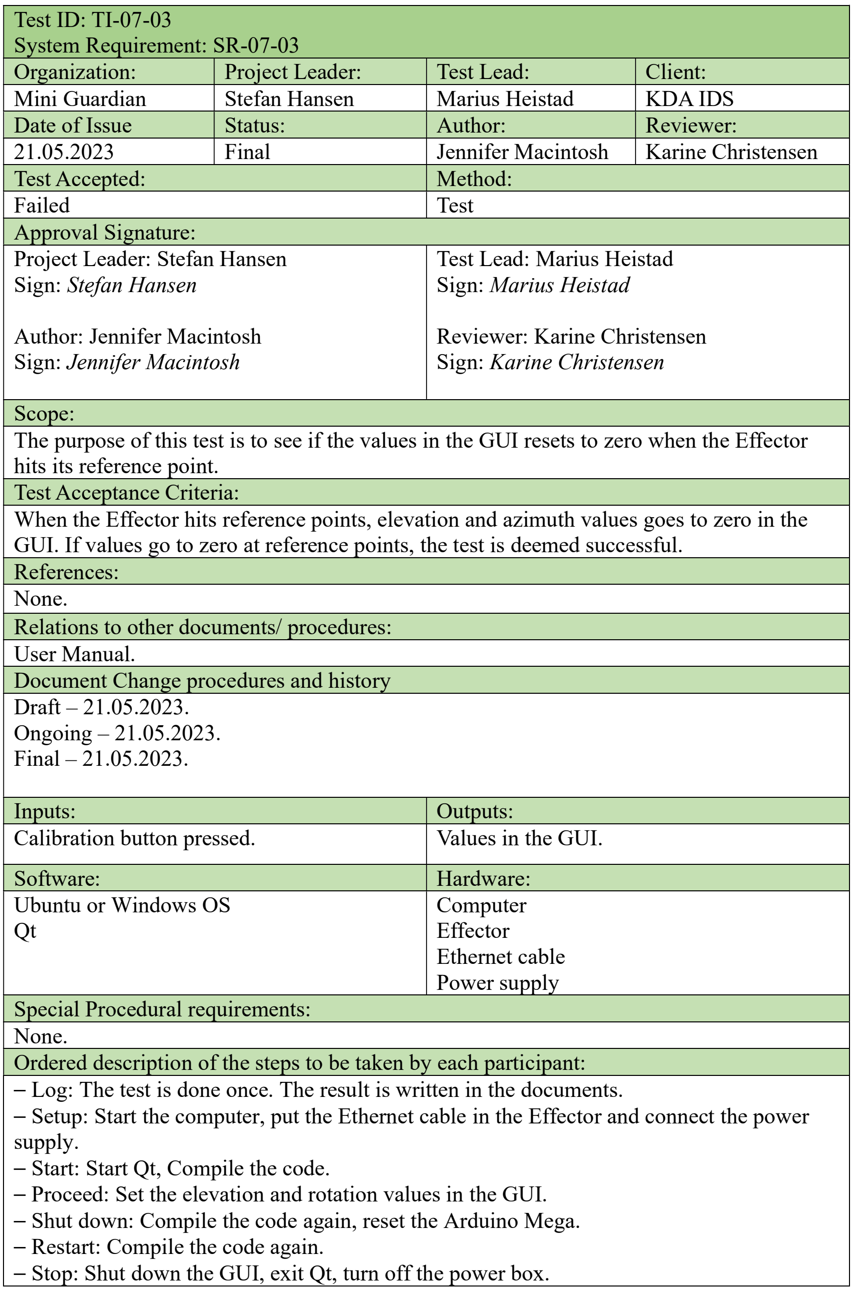
O Test report: TI-07-02

Test ID: TI-07-02 System Requirement: SR-07-02			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
20.05.2023	Final	Jennifer Macintosh	Veronica Juverud
Test Accepted:		Method:	
Yes		Test	
Approval Signature:			
Project Leader: Stefan Hansen Sign: *Stefan Hansen* Author: Jennifer Macintosh Sign: *Jennifer Macintosh*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Veronica Juverud Sign: *Veronica Juverud*	
Scope:			
The purpose of this test is to verify that the GUI displays azimuth and elevation values.			
Test Acceptance Criteria:			
The GUI shall display elevation and azimuth values. If value appears, the test is deemed successful.			
References:			
None.			
Relations to other documents/ procedures:			
None.			
Document Change procedures and history			
Draft – 20.05.2023. Ongoing – 20.05.2023. Final – 20.05.2023.			
Inputs:		Outputs:	
Values from Arduino Mega.		Azimuth and elevation values in GUI.	
Software:		Hardware:	
Ubuntu or Windows OS Qt		Computer Ethernet cable Effector Power supply	
Special Procedural requirements:			
None.			
Ordered description of the steps to be taken by each participant:			
– Log: The test is done one, result written in test document. – Setup: Start the computer, start Qt, put the Ethernet cable in the Effector and turn the power on. – Start: Compile the code in Qt. – Proceed: Type in elevation and rotation values in the GUI. – Shut down: Compile the code again. – Restart: Compile the code again. – Stop: Shut down the GUI, exit Qt. – Contingencies: It can be necessary to reset the Arduino Mega if the Effector is acting in a way it should not.			

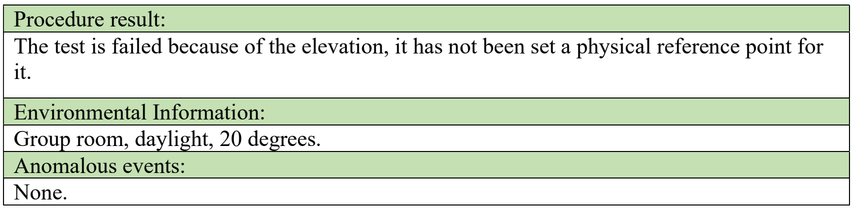


Procedure result:
Visual approval, the elevation and rotation was displayed in the GUI.
Environmental Information:
Group room, daylight, 20 degrees.
Anomalous events:
None.

P Test report: TI-07-03



Test ID: TI-07-03 System Requirement: SR-07-03			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
21.05.2023	Final	Jennifer Macintosh	Karine Christensen
Test Accepted:		Method:	
Failed		Test	
Approval Signature:			
Project Leader: Stefan Hansen Sign: *Stefan Hansen* Author: Jennifer Macintosh Sign: *Jennifer Macintosh*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Karine Christensen Sign: *Karine Christensen*	
Scope:			
The purpose of this test is to see if the values in the GUI resets to zero when the Effector hits its reference point.			
Test Acceptance Criteria:			
When the Effector hits reference points, elevation and azimuth values goes to zero in the GUI. If values go to zero at reference points, the test is deemed successful.			
References:			
None.			
Relations to other documents/ procedures:			
User Manual.			
Document Change procedures and history			
Draft – 21.05.2023. Ongoing – 21.05.2023. Final – 21.05.2023.			
Inputs:		Outputs:	
Calibration button pressed.		Values in the GUI.	
Software:		Hardware:	
Ubuntu or Windows OS Qt		Computer Effector Ethernet cable Power supply	
Special Procedural requirements:			
None.			
Ordered description of the steps to be taken by each participant:			
– Log: The test is done once. The result is written in the documents. – Setup: Start the computer, put the Ethernet cable in the Effector and connect the power supply. – Start: Start Qt, Compile the code. – Proceed: Set the elevation and rotation values in the GUI. – Shut down: Compile the code again, reset the Arduino Mega. – Restart: Compile the code again. – Stop: Shut down the GUI, exit Qt, turn off the power box.			



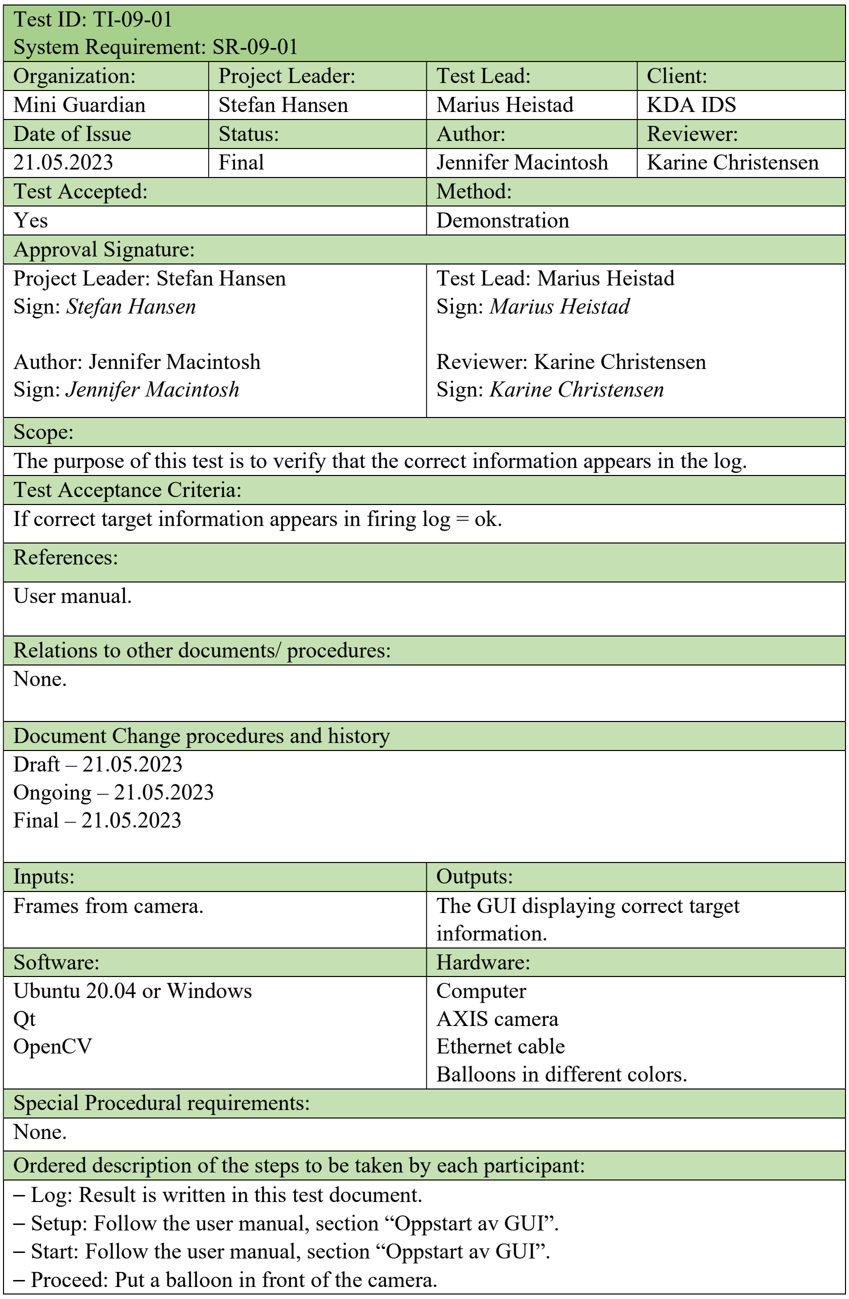
Procedure result:
The test is failed because of the elevation, it has not been set a physical reference point for it.
Environmental Information:
Group room, daylight, 20 degrees.
Anomalous events:
None.

Q Test report: TI-08-01

Test ID: TI-08-01 System Requirement: SR-08-01			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
21.05.2023	Final	Veronica Juverud	Jennifer Macintosh
Test Accepted:		Method:	
Yes		Demonstration	
Approval Signature:			
Project Leader: Stefan Hansen Sign: *Stefan Hansen* Author: Veronica Juverud Sign: *Veronica Juverud*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Jennifer Macintosh Sign: *Jennifer Macintosh*	
Scope:			
The purpose of this test report is to assess whether HUF information appears in the GUI when objects are detected.			
Test Acceptance Criteria:			
If correct HUF information appears on identified targets in the GUI according to MIL-STD-1477, the test is deemed successful.			
References:			
User manual.			
Relations to other documents/ procedures:			
None.			
Document Change procedures and history			
Draft – 21.05.2023 Ongoing – 21.05.2023 Final – 21.05.2023			
Inputs:		Outputs:	
Image from AXIS camera.		The GUI displaying HUF information of the detected objects.	
Software:		Hardware:	
Ubuntu 20.04 ROS Noetic Qt OpenCV		Computer AXIS camera Ethernet cable Balloon	
Special Procedural requirements:			
None.			
Ordered description of the steps to be taken by each participant:			
– Log: Result is written in this test document. – Setup: Follow the user manual, section “Oppstart av GUI”.			

– Start: Follow the user manual, section “Oppstart av GUI”. – Proceed: Put a balloon in front of the camera and see whether the HUF information appears. – Measurement: No measurements are made. – Shut down: If the test is temporarily suspended due to unscheduled events, restart the GUI. – Restart: To restart the test at certain points, follow the user manual, section “Oppstart av GUI”. – Stop: To stop the test, close the GUI window and Qt. – Wrap-up: When the test is finished, results are written in test document. – Contingencies: Unexpected events that can occur during the test execution is dealt with by restarting the test if necessary.
Procedure result:
The GUI does successfully display correct HUF information on identified targets in the GUI, therefore the test is deemed successful.
Environmental Information:
The test was done at Innovasjonsloftet in room I-214, 17:00, sunshine outside, approximately 20 degrees.
Anomalous events:
None.

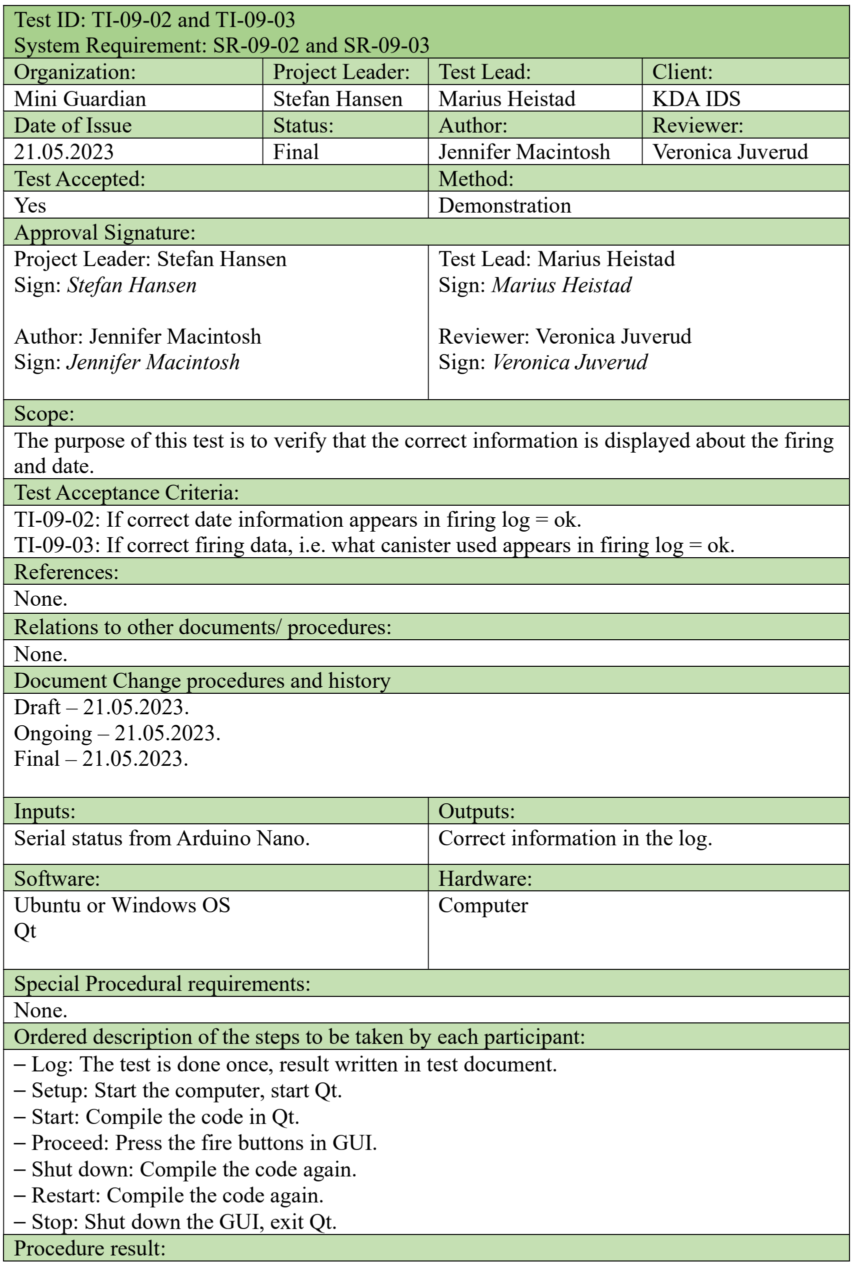
R Test report: TI-09-01



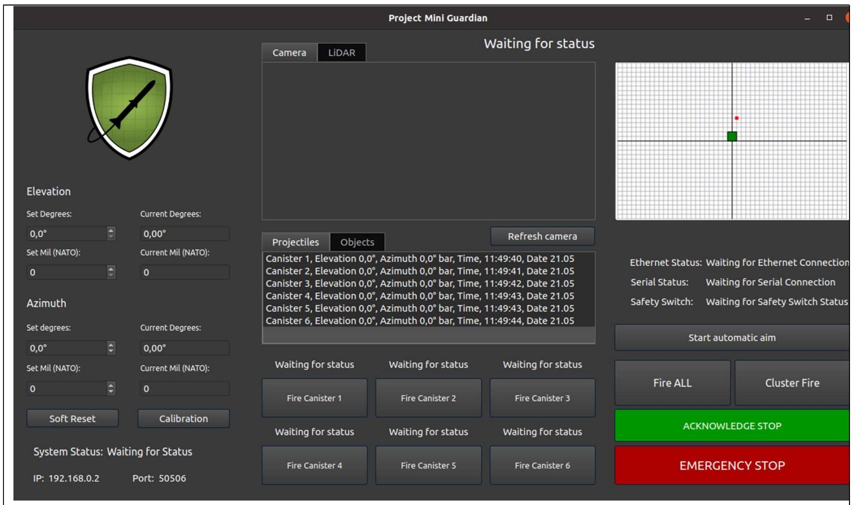
Test ID: TI-09-01 System Requirement: SR-09-01			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
21.05.2023	Final	Jennifer Macintosh	Karine Christensen
Test Accepted:		Method:	
Yes		Demonstration	
Approval Signature:			
Project Leader: Stefan Hansen Sign: *Stefan Hansen* Author: Jennifer Macintosh Sign: *Jennifer Macintosh*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Karine Christensen Sign: *Karine Christensen*	
Scope:			
The purpose of this test is to verify that the correct information appears in the log.			
Test Acceptance Criteria:			
If correct target information appears in firing log = ok.			
References:			
User manual.			
Relations to other documents/ procedures:			
None.			
Document Change procedures and history			
Draft – 21.05.2023 Ongoing – 21.05.2023 Final – 21.05.2023			
Inputs:		Outputs:	
Frames from camera.		The GUI displaying correct target information.	
Software:		Hardware:	
Ubuntu 20.04 or Windows Qt OpenCV		Computer AXIS camera Ethernet cable Balloons in different colors.	
Special Procedural requirements:			
None.			
Ordered description of the steps to be taken by each participant:			
– Log: Result is written in this test document. – Setup: Follow the user manual, section “Oppstart av GUI”. – Start: Follow the user manual, section “Oppstart av GUI”. – Proceed: Put a balloon in front of the camera.			

– Measurement: No measurements are made. – Shut down: If the test is temporarily suspended due to unscheduled events, restart the GUI. – Restart: To restart the test at certain points, follow the user manual, section “Oppstart av GUI”. – Stop: To stop the test, close the GUI window and Qt. – Wrap-up: When the test is finished, results are written in test document. – Contingencies: Unexpected events that can occur during the test execution is dealt with by restarting the test if necessary.
Procedure result:
The GUI does display correct information on identified targets, the test is deemed accepted.
Environmental Information:
The test was done at Innovasjonsloftet in room I-214, sunshine outside, approximately 20 degrees
Anomalous events:
None.

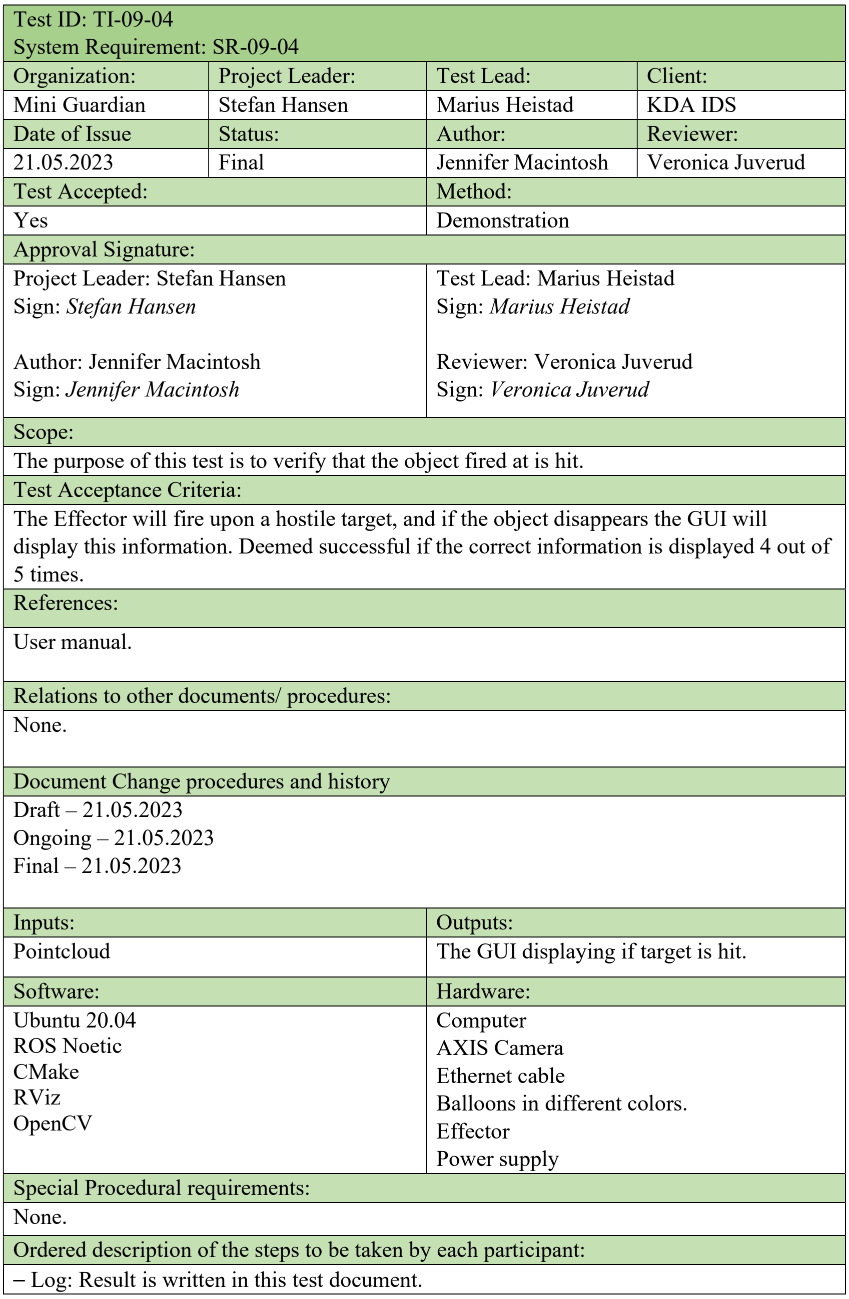
S Test report: TI-09-02 and TI-09-03



Test ID: TI-09-02 and TI-09-03 System Requirement: SR-09-02 and SR-09-03			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
21.05.2023	Final	Jennifer Macintosh	Veronica Juverud
Test Accepted:		Method:	
Yes		Demonstration	
Approval Signature:			
Project Leader: Stefan Hansen Sign: *Stefan Hansen* Author: Jennifer Macintosh Sign: *Jennifer Macintosh*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Veronica Juverud Sign: *Veronica Juverud*	
Scope:			
The purpose of this test is to verify that the correct information is displayed about the firing and date.			
Test Acceptance Criteria:			
TI-09-02: If correct date information appears in firing log = ok. TI-09-03: If correct firing data, i.e. what canister used appears in firing log = ok.			
References:			
None.			
Relations to other documents/ procedures:			
None.			
Document Change procedures and history			
Draft – 21.05.2023. Ongoing – 21.05.2023. Final – 21.05.2023.			
Inputs:		Outputs:	
Serial status from Arduino Nano.		Correct information in the log.	
Software:		Hardware:	
Ubuntu or Windows OS Qt		Computer	
Special Procedural requirements:			
None.			
Ordered description of the steps to be taken by each participant:			
– Log: The test is done once, result written in test document. – Setup: Start the computer, start Qt. – Start: Compile the code in Qt. – Proceed: Press the fire buttons in GUI. – Shut down: Compile the code again. – Restart: Compile the code again. – Stop: Shut down the GUI, exit Qt.			
Procedure result:			

 Project Mini Guardian Waiting for status Elevation Set Degrees: 0,0° Current Degrees: 0,00° Set Mil (NATO): 0 Current Mil (NATO): 0 Azimuth Set degrees: 0,0° Current Degrees: 0,00° Set Mil (NATO): 0 Current Mil (NATO): 0 System Status: Waiting for Status IP: 192.168.0.2 Port: 50506 Canister 1, Elevation 0,0°, Azimuth 0,0° bar, Time, 11:49:40, Date 21.05 Canister 2, Elevation 0,0°, Azimuth 0,0° bar, Time, 11:49:41, Date 21.05 Canister 3, Elevation 0,0°, Azimuth 0,0° bar, Time, 11:49:42, Date 21.05 Canister 4, Elevation 0,0°, Azimuth 0,0° bar, Time, 11:49:43, Date 21.05 Canister 5, Elevation 0,0°, Azimuth 0,0° bar, Time, 11:49:43, Date 21.05 Canister 6, Elevation 0,0°, Azimuth 0,0° bar, Time, 11:49:44, Date 21.05 Ethernet Status: Waiting for Ethernet Connection Serial Status: Waiting for Serial Connection Safety Switch: Waiting for Safety Switch Status Waiting for status Waiting for status Waiting for status Waiting for status Waiting for status Waiting for status Camera LiDAR Projectiles Objects Refresh camera Soft Reset Calibration Fire Canister 1 Fire Canister 2 Fire Canister 3 Fire Canister 4 Fire Canister 5 Fire Canister 6 Start automatic aim Fire ALL Cluster Fire ACKNOWLEDGE STOP EMERGENCY STOP The test is accepted, the correct information is shown in the log.
Environmental Information:
Group room, daylight, 20 degrees.
Anomalous events:
None.

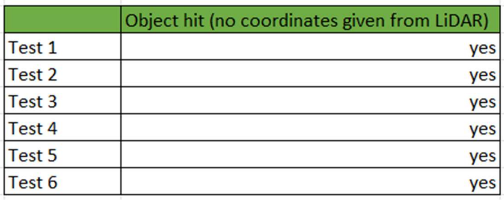
T Test report: TI-09-04



Test ID: TI-09-04 System Requirement: SR-09-04			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
21.05.2023	Final	Jennifer Macintosh	Veronica Juverud
Test Accepted:		Method:	
Yes		Demonstration	
Approval Signature:			
Project Leader: Stefan Hansen Sign: *Stefan Hansen* Author: Jennifer Macintosh Sign: *Jennifer Macintosh*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Veronica Juverud Sign: *Veronica Juverud*	
Scope:			
The purpose of this test is to verify that the object fired at is hit.			
Test Acceptance Criteria:			
The Effector will fire upon a hostile target, and if the object disappears the GUI will display this information. Deemed successful if the correct information is displayed 4 out of 5 times.			
References:			
User manual.			
Relations to other documents/ procedures:			
None.			
Document Change procedures and history			
Draft – 21.05.2023 Ongoing – 21.05.2023 Final – 21.05.2023			
Inputs:		Outputs:	
Pointcloud		The GUI displaying if target is hit.	
Software:		Hardware:	
Ubuntu 20.04 ROS Noetic CMake RViz OpenCV		Computer AXIS Camera Ethernet cable Balloons in different colors. Effector Power supply	
Special Procedural requirements:			
None.			
Ordered description of the steps to be taken by each participant:			
– Log: Result is written in this test document.			

– Setup: Follow the user manual, section “Oppstart av GUI”.
– Start: Follow the user manual, section “Oppstart av GUI”.
– Proceed: Put a balloon in front of the camera. Press the “start automatic aim” button in the GUI, press any fire canister button, press the “start automatic aim” button again to get a new set of coordinates from the GUI.
– Measurement: No measurements are made.
– Shut down: If the test is temporarily suspended due to unscheduled events, refer to the User manual.
– Restart: To restart the test at certain points, follow the user manual.
– Stop: To stop the test, close the GUI window and Qt, close ROS.
– Wrap-up: When the test is finished, results are written in test document.
– Contingencies: Unexpected events that can occur during the test execution is dealt with by restarting the test if necessary.

Procedure result:



	Object hit (no coordinates given from LiDAR)
Test 1	yes
Test 2	yes
Test 3	yes
Test 4	yes
Test 5	yes
Test 6	yes

The test is accepted, the LiDAR sent empty coordinates after every firing. The camera was used to confirm that the target disappeared.

Environmental Information:

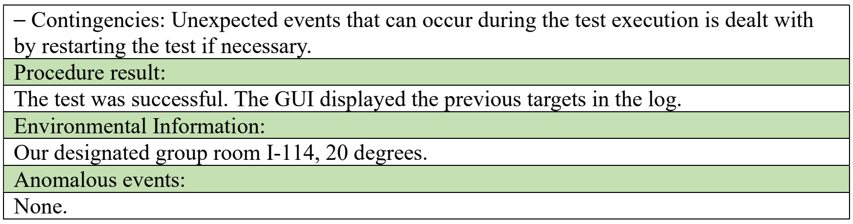
The test was done at Dronesonen, sunshine outside, approximately 20 degrees

Anomalous events:

None.

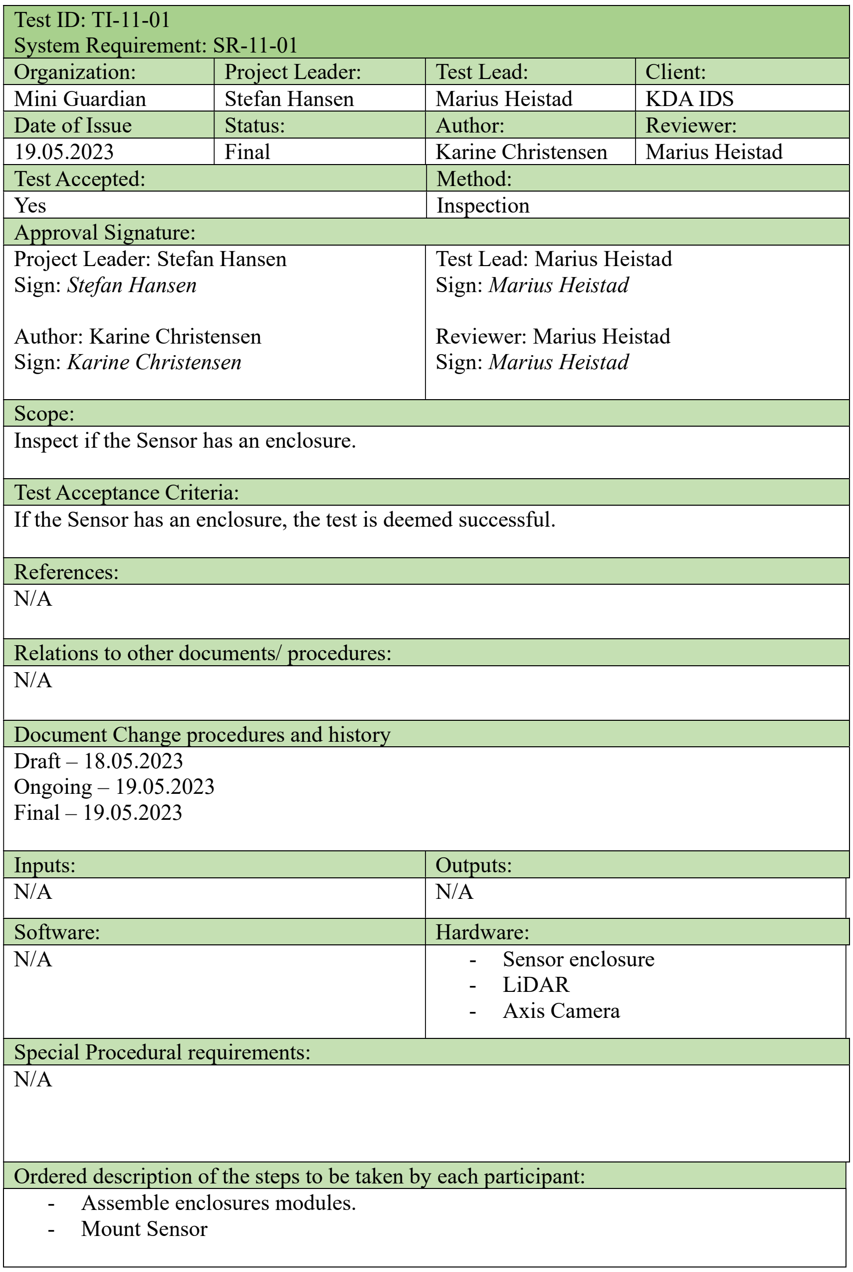
U Test report: TI-10-01

Test ID: TI-10-01 and TI-10-02 System Requirement: SR-10-01 and SR-10-01			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
21.05.2023	Final	Jennifer Macintosh	Veronica Juverud
Test Accepted:		Method:	
Yes		Demonstration	
Approval Signature:			
Project Leader: Stefan Hansen Sign: *Stefan Hansen* Author: Jennifer Macintosh Sign: *Jennifer Macintosh*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Veronica Juverud Sign: *Veronica Juverud*	
Scope:			
The purpose of the test is to see if the information about the objects appears in the GUI.			
Test Acceptance Criteria:			
Information about previous target assessments is shown in the GUI. If information is shown, the test is deemed successful.			
References:			
None.			
Relations to other documents/ procedures:			
None.			
Document Change procedures and history			
Draft – 21.05.2023. Ongoing – 21.05.2023. Final – 21.05.2023.			
Inputs:		Outputs:	
Video frames from camera.		Information in the log.	
Software:		Hardware:	
Qt OpenCV Windows or Ubuntu OS		Computer Camera Balloons Ethernet cable	
Special Procedural requirements:			
None.			
Ordered description of the steps to be taken by each participant:			
– Log: The test is done once, the result is written in the test document. – Setup: Follow the user manual, section “Oppstart av GUI”. – Start Follow the user manual, section “Oppstart av GUI”. Place a balloon in the frame of the camera and visually confirm the result in the GUI. – Measurement: Visual confirmation that the right information is shown. – Shut down: Shut down the GUI application, compile the code again. – Restart: Shut down the GUI application, compile the code again. – Stop: Shut down the GUI application.			

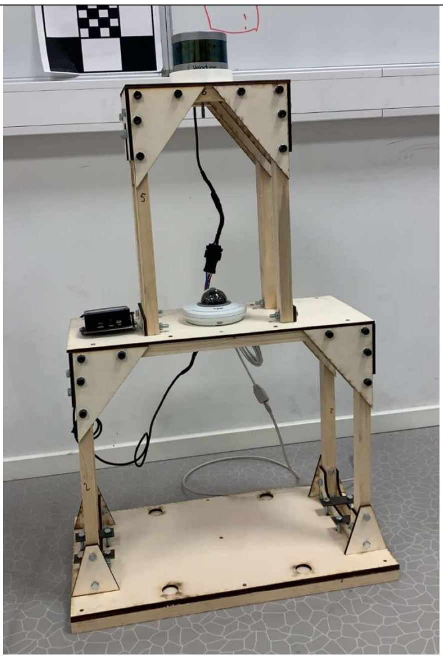


– Contingencies: Unexpected events that can occur during the test execution is dealt with by restarting the test if necessary.
Procedure result:
The test was successful. The GUI displayed the previous targets in the log.
Environmental Information:
Our designated group room I-114, 20 degrees.
Anomalous events:
None.

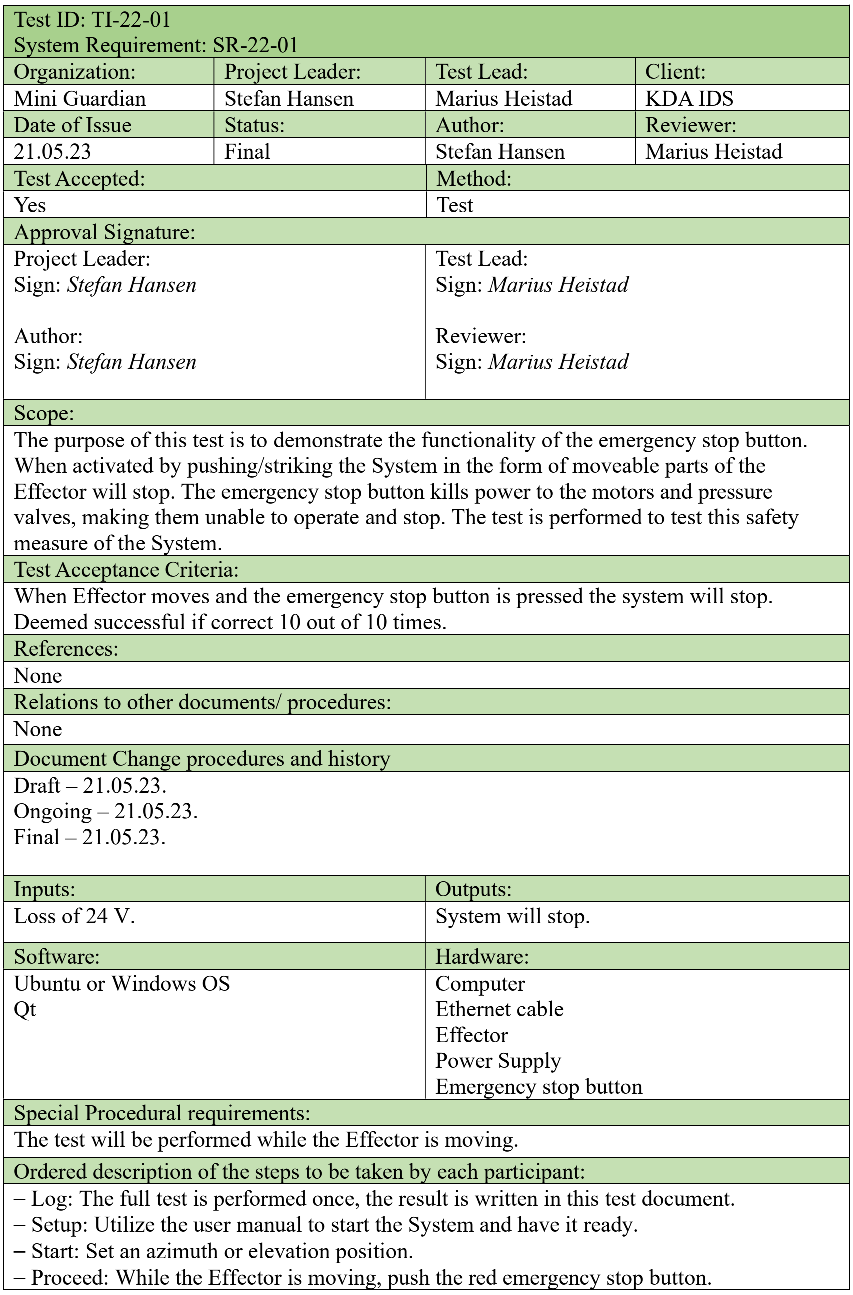
V Test report: TI-11-01



Test ID: TI-11-01 System Requirement: SR-11-01			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
19.05.2023	Final	Karine Christensen	Marius Heistad
Test Accepted:		Method:	
Yes		Inspection	
Approval Signature:			
Project Leader: Stefan Hansen Sign: *Stefan Hansen* Author: Karine Christensen Sign: *Karine Christensen*		Test Lead: Marius Heistad Sign: *Marius Heistad* Reviewer: Marius Heistad Sign: *Marius Heistad*	
Scope:			
Inspect if the Sensor has an enclosure.			
Test Acceptance Criteria:			
If the Sensor has an enclosure, the test is deemed successful.			
References:			
N/A			
Relations to other documents/ procedures:			
N/A			
Document Change procedures and history			
Draft – 18.05.2023 Ongoing – 19.05.2023 Final – 19.05.2023			
Inputs:		Outputs:	
N/A		N/A	
Software:		Hardware:	
N/A		- Sensor enclosure - LiDAR - Axis Camera	
Special Procedural requirements:			
N/A			
Ordered description of the steps to be taken by each participant:			
- Assemble enclosures modules. - Mount Sensor			


Procedure result:
The Sensor has an enclosure, the test is therefore deemed successful.
Environmental Information:
Must be placed on flat surface. Is portable.
Anomalous events:
N/A

W Test report: TI-22-01



Test ID: TI-22-01 System Requirement: SR-22-01			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
21.05.23	Final	Stefan Hansen	Marius Heistad
Test Accepted:		Method:	
Yes		Test	
Approval Signature:			
Project Leader: Sign: *Stefan Hansen* Author: Sign: *Stefan Hansen*		Test Lead: Sign: *Marius Heistad* Reviewer: Sign: *Marius Heistad*	
Scope:			
The purpose of this test is to demonstrate the functionality of the emergency stop button. When activated by pushing/striking the System in the form of moveable parts of the Effector will stop. The emergency stop button kills power to the motors and pressure valves, making them unable to operate and stop. The test is performed to test this safety measure of the System.			
Test Acceptance Criteria:			
When Effector moves and the emergency stop button is pressed the system will stop. Deemed successful if correct 10 out of 10 times.			
References:			
None			
Relations to other documents/ procedures:			
None			
Document Change procedures and history			
Draft – 21.05.23. Ongoing – 21.05.23. Final – 21.05.23.			
Inputs:		Outputs:	
Loss of 24 V.		System will stop.	
Software:		Hardware:	
Ubuntu or Windows OS Qt		Computer Ethernet cable Effector Power Supply Emergency stop button	
Special Procedural requirements:			
The test will be performed while the Effector is moving.			
Ordered description of the steps to be taken by each participant:			
– Log: The full test is performed once, the result is written in this test document. – Setup: Utilize the user manual to start the System and have it ready. – Start: Set an azimuth or elevation position. – Proceed: While the Effector is moving, push the red emergency stop button.			

– Measurement: Visually inspect that the Effector has stopped. Try to fire projectiles and audibly inspect that the solenoid valves do not open. – Shut down: Shut down GUI, power off System, wait 30 seconds, power on System, open GUI. – Restart: Set Effector back to zero-point position. – Stop: Performing the test will stop the Effector. – Wrap-up: The results are written down during the tests. – Contingencies: In the case of any interruptions during the test execution will be restarted.
Procedure result:
The test was performed 10 times. Both azimuth and elevation positions were set. The Effector stopped 10 times. The Effector was not able to fire projectiles 10 times.
Environmental Information:
Group office, daylight, 22 degrees Celsius
Anomalous events:
None

X Test report: TI-22-02

Test ID: TI-22-02 System Requirement: NSR-22-02			
Organization:	Project Leader:	Test Lead:	Client:
Mini Guardian	Stefan Hansen	Marius Heistad	KDA IDS
Date of Issue	Status:	Author:	Reviewer:
21.05.2023	Final	Karine Christensen	Marius Heistad
Test Accepted:		Method	
Yes		Demonstration	
Approval Signature:			
Project Leader: Sign: *Stefan Hansen* Author: Sign: *Karine Christensen*		Test Lead: Sign: *Marius Heistad* Reviewer: Sign: *Marius Heistad*	
Scope:			
The goal of this test is to ensure that both the mechanical safety barriers for firing projectiles function properly. The purpose is to make sure the projectiles can not be fired when the mechanical barriers are on. The safety barriers consist of a physical protection cover covering the canisters and a safety switch that cuts the 24 V supply to the pneumatic valves.			
Test Acceptance Criteria:			
The test is deemed successful if the safety barriers are sufficient.			
References:			
N/A			
Relations to other documents/ procedures:			
See user manual for start-up.			
Document Change procedures and history			
Draft – 21.05.2023 Ongoing – 21.05.2023 Final – 21.05.2023			
Inputs:		Outputs:	
- Mounting the protection cover. - Pushing the safety switch.		- When the barriers are on, the Effector can not fire upon/hit targets.	
Software:		Hardware:	
- Windows or Ubuntu OS. - QT.		- Power Supply. - Effector. - Ethernet cables. - Power cable. - Protective cover.	
Special Procedural requirements:			
N/A			
Ordered description of the steps to be taken by each participant:			

- Power up the system. - Push the safety switch. - After pushing the safety switch, the GUI displays that the Effector is unable to fire. An attempt to fire is conducted and the Effector does not respond. - Mount the protective cover. - An attempt to fire is conducted while the protective cover is mounted. The Effector fires the projectiles, but they do not leave their canisters because of the blockage from the protective plate.
Procedure result:
The test is deemed successful since: - The Effector is unable to fire the projectiles when the safety switch is turned off. - The Effector is able to fire, but the projectiles do not leave their canisters when the protective cover is mounted. -
Environmental Information:
The test has been conducted in room i-114 in room temperature.
Anomalous events:
N/A

Y Tension test report

KC

Test type	
Performed by	Karine Christensen - MiniGuardian Martin Haugerud - MINRAD
Date	27.04.2023
Goal	Research yield strength and E-modulus of Nylon 12, Tough PLA and ABS
Standard	Based on ASTM D 638-02A
Test equipment	LLOYD Instruments LR10K Tension test machine Caliper

When designing parts and products to be produced by 3D-printing, it is important to have knowledge about the materials properties. It is possible to find material data sheets from the producers but keeping all the factors weighing in on the final product's quality in mind, these data sheets may not be realistic for the chosen product. Because of this, the tension test is necessary, giving an overview over for instance the materials yield strength and E-modulus. The materials used in this test are Nylon 12, Tough Polylactic acid (PLA) and Acrylonitrile Butadiene Styrene (ABS).

A total of 46 test-specimen were printed, of which ten are made of Nylon 12, 18 is made of Tough PLA and 18 is made of ABS. For the Nylon 12 material, the specimen was printed in two different orientations: five lying flat and five rotated 45 degrees in both x and y direction as seen in fig. 1. The Tough PLA and ABS specimen was printed in three orientations, where six specimens were printed lying flat, six was printed standing and six was printed in a 45-degree angle with the base plate.

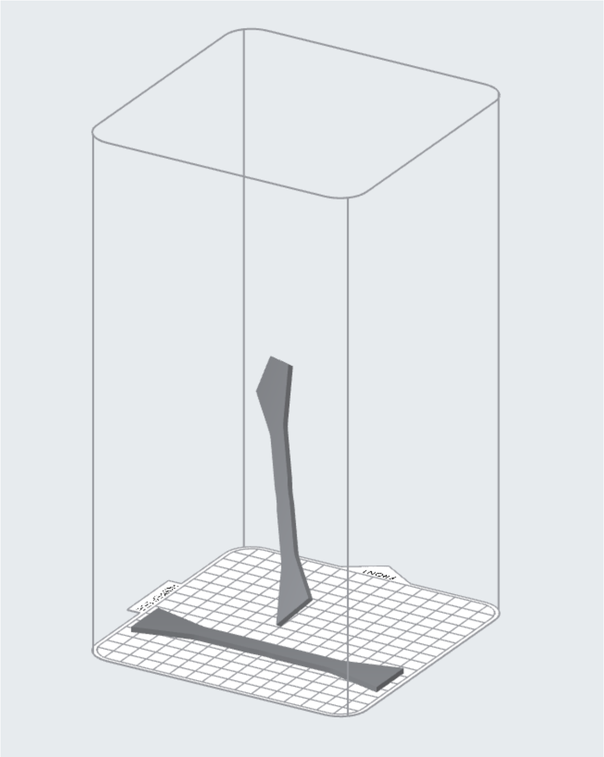


Figure 1: Illustration of the orientations used when printing with Nylon 12.

To produce the test specimen, the printers used were the Formlabs Fuse 1+ [114] and the Ultimaker S5 [115]. The former is an Selective Laser Sintering (SLS) printer using nylon powder. The Ultimaker S5 is an Fused Deposition Modeling (FDM) printer using filament of different polymers, in this case Tough PLA and ABS.

An important factor to remember regarding 3D-printing, is that the settings used in the printers slicer-software will affect the properties of the final product. These settings can for instance be the layer thickness, percentage of infill and the print temperature.

When printing with the Formlabs Fusion 1+ SLS-printer, the slicer-software PreForm [116] was used. The material needed in these types of printers comes in the shape of fine powder and the printer uses a 30W Ytterbium fiber laser to sinter it. The layer thickness therefore becomes approx. 110 microns. Products produced in an SLS-printer will have 100% infill [114].

The printing process regarding the FDM printer is different from the SLS-printer. With the FDM printer, a polymer thread is melted and extruded in layers to create the desired shape. The most important settings when printing using the FDM-printer are listed in Table. 1. This table also shows the values used for the different settings when printing for this test.

Table 1: Table of settings used in FDM-printers.

	PLA	ABS
Layer thickness	0.1 mm	0.1 mm
Infill	100 %	100 %
Print temperature	210 ° C	240 ° C
Baseplate temperature	60 ° C	85 ° C
Print speed	30.0 mm/s	30.0 mm/s
Fan speed	100 %	2.0 %

Every test specimen is printed using the same 3D-model and should therefore have approx. equal cross sections. Every specimen has a gage area of 80 mm. The machine used to perform the tests is the LLOYD Instruments LR10K tension test machine located in USNs Materialteknisk laboratory.

From the producer of the materials data sheet, the theoretical yield strength and E-modulus is found. These values can be seen in Table 2.

Table 2: The theoretical values from the producer.

	Nylon 12 [117]	**Tough PLA [118]**	**ABS** [119]
Strekkfasthet	50 MPa	33.4 – 47.5 MPa	38.1 MPa
E-modul	1850 MPa	2696 – 2797 MPa	1699-1962 MPa

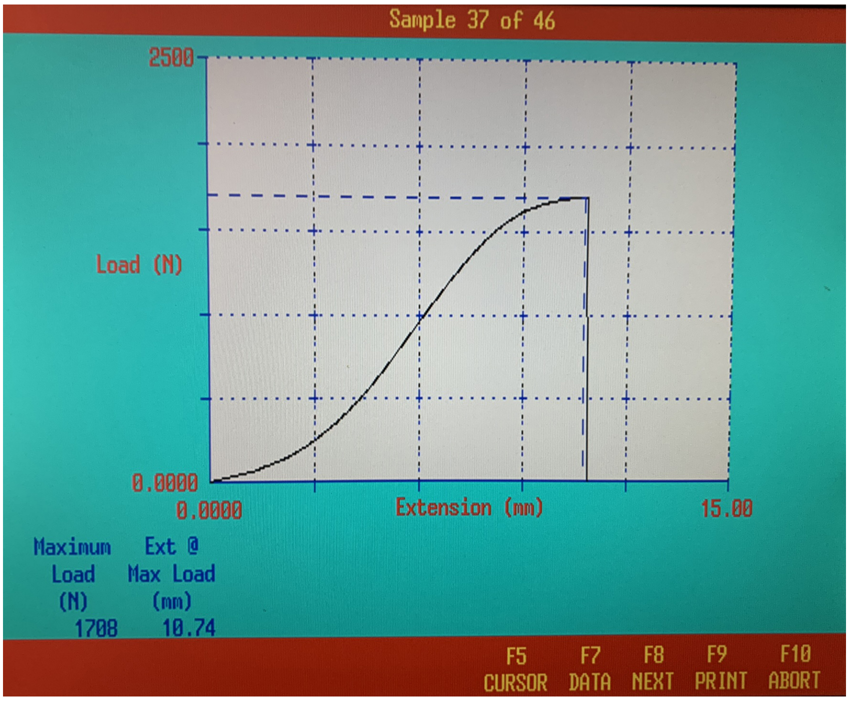
Because of ambiguities regarding the producers test methods, the tension test will be performed to confirm the given values. Therefore, the given values represents the tests hypothesis.
Before the tests can begin, the specimen must be marked with numbers and measured to get the most accurate value for the cross section as possible. When the measurements are logged, the specimen can be placed in the machine and the test can proceed. The test is conducted by the machine pulling the test specimen with an increasing force until the specimen fractures. In this case, the machine pulls the specimen at a speed of 25 mm/s. Information about the force used and the extension before fracture is logged by the machine and is displayed as a curve after the test is done. These values are used to calculate the maximum tension the material can take and the material's E-modulus. After the test is finished, the information and curve are saved, and the fractures direction and placement are examined. Every one of the 46 specimens was tested this way and the results can be seen in Tables 3, 4 and 5.

At the tables left side, the material used and the orientation the specimen is printed in is visible. As mentioned earlier, six test specimens were produced in each of the three orientations for Tough PLA and ABS, and five test specimens were produced in both orientations for Nylon 12. Every specimen has, as seen in the column second to the left, been assigned its own test number. Further, the logging of the width, thickness and cross section follows, before the values of force and extension given by the machine are seen. After this, the yield strength and the E-modulus are calculated. At the far right, the fracture type is described. This is a description of how the present fracture looks and where it is located. The definition of the different labels is explained in Table 6.

Table 6: Explanation of different fracture types.

Fracture direction	Area	Placement
L (Lateral)	I (In grip)	T (Top)
A (Angular)	T (Transition)	M (Middle)
X (Not Lateral or angular)	G (Gage)	B (Bottom)

This means that fractures marked with “LGM” are a lateral fracture in the gage and placed on the middle of the specimen.



```
Maximum   Ext @
  Load   Max Load
  (N)     (mm)
  1708    10.74
```

Figure 2: The information given by the machine shown as a curve on the screen.

Fig. 2 shows the information given by the tension-test machine after finishing a tension test. The values for the maximum force used and the maximum elongation before fracture are seen in blue writing. An approximation of the E-modulus of every one of the 46 test specimens is calculated in the same, simplified way the E-modulus for specimen N0_1 is calculated here:

Since the E-modulus is the slope of the curve in the tension-elongation graph and $E = \frac{\sigma}{\epsilon}$, from Hookes law [106], the values for the tension σ and the elongation ϵ must be determined.

When finding the E-modulus of the curve, an approx. linear part of the curve is focused on. Two points on the curve are chosen in opposite ends of the most linear area, and the force and elongation in these points are read from the graph. The load axis has a maximum value of 2500N and is divided into five parts. This means that each line in the vertical direction will represent 500N. Since the most linear part of the curve is between 500N and 1000N, these will be the forces used in the calculations. The extension axis is also divided into five parts, with a maximum value of 15 mm. this means that every line in the horizontal direction represents three mm. The chosen points give these values for force and elongation, as seen in Table 7:

Table 7: Table showing values used to calculate E-module.

F_1	500 N
F_2	100 N
X_1	4.30 mm
X_2	6.10 mm
L	80 mm
L_1	$L - X_1$
A	$31.66\,\mathrm{mm}^2$

The E-modulus is calculated by using (4). This equation Is found by implementing the equations for elongation ϵ (2) and tension σ (3) in the original E-modulus equation (1):

$$E = \frac{\sigma}{\epsilon} \tag{1}$$

$$\epsilon = \frac{X}{L} \tag{2}$$

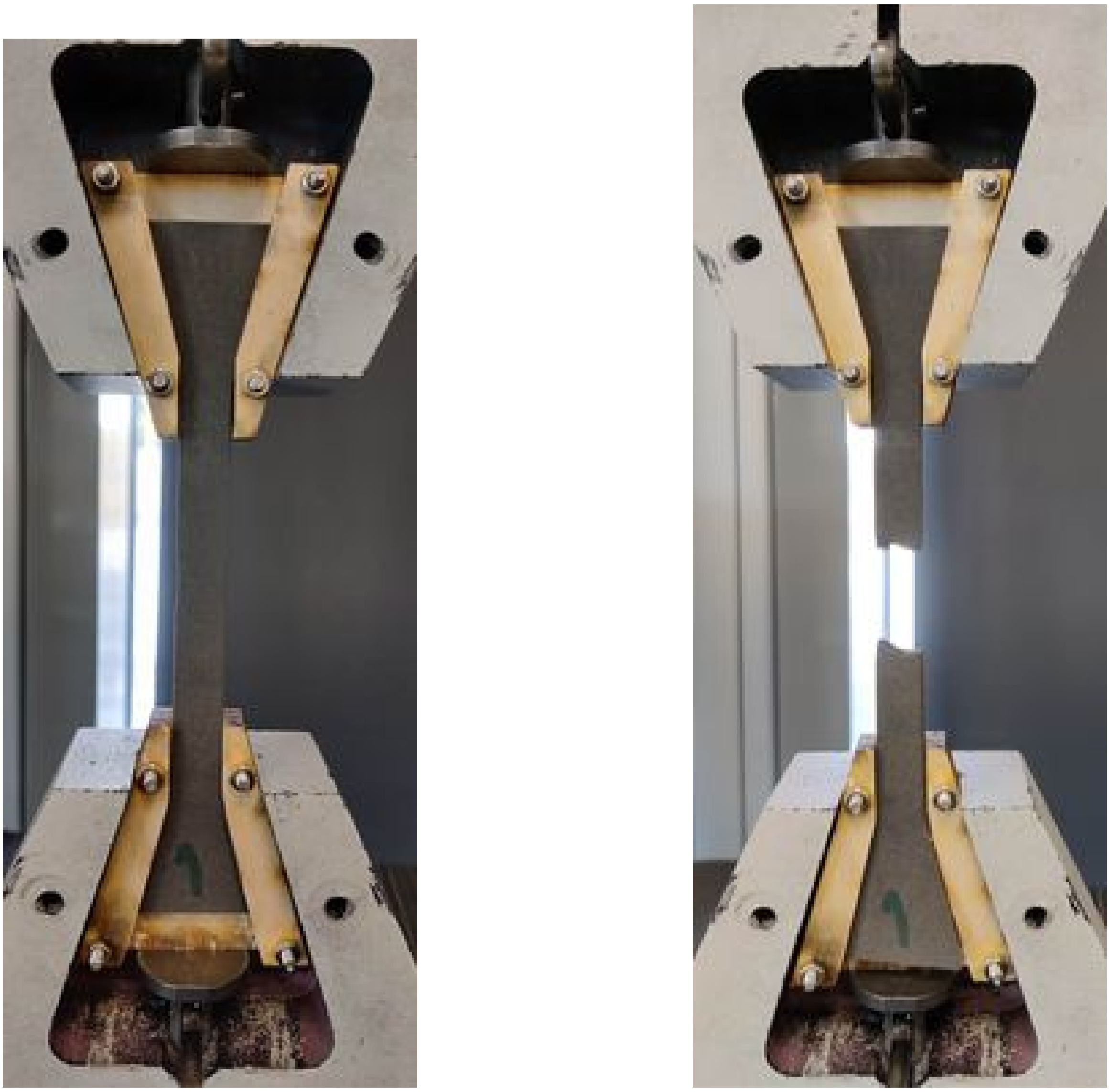
$$\sigma = \frac{F}{A} \tag{3}$$

$$E = \frac{\sigma_2 - \sigma_1}{\epsilon_2 - \epsilon_1}, \tag{4}$$

where E is the E-module, X is the extension of the test specimen, F is the applied force and A is the specimens cross section. Using (2), (3) and (4), the E-modulus is calculated:

$$E = \frac{\frac{F_2 - F_1}{A}}{\frac{X_2 - X_1}{L_1}} = \frac{\frac{1000\,\mathrm{N} - 500\,\mathrm{N}}{31.66\,\mathrm{mm}^2}}{\frac{6.10\,\mathrm{mm} - 4.30\,\mathrm{mm}}{80.00\,\mathrm{mm} + 4.30\,\mathrm{mm}}} = 739.63\,\mathrm{MPa}. \tag{5}$$

Every E-modulus is calculated this way and the results can be seen in Tables 3, 4 and 5. Fig. 3a shows how the test specimen are attached to the tension-test machine and fig. 3b shows how the fracture of N0_1 turned out.



(a) The test specimen N0_1 placed in the tension test machine.

(b) The test specimen N0_1 after executed test.

Figure 3: Test specimen N0_1 before and after fracture.

As shown in fig. 3b, the fracture on N0_1 is an “XGM” fracture, meaning it is neither a lateral nor an angular fracture in the gage-area placed in the middle of the specimen.

In conclusion, it has been seen that the yield strength of polymers differs according to what material and which orientation is used.

For Nylon 12, the flat print shows the greatest strength. This material and orientation achieve an average value of 51,86 MPa, while the oblique orientation reaches an average of 48,9 MPa. This matches the given value from the data sheet, at 50 MPa. The difference in strength in the two tested orientations is not very significant, only approx. three MPa. This means that Nylon 12 has a more equal strength in different directions than both the ABS and the Tough PLA.

For the Tough PLA, the flat print also shows the greatest strength with an average of

52,36 MPa. The variation between the different orientations is clearly more significant than for the Nylon 12, but not as significant as for the ABS. Oblique and standing orientations achieved respectively 43,12 MPa and 32,67 MPa. These values also match the values given in the data sheet at 33.4 - 47.5 MPa quite good. The variation in yield strength is due to the print orientation.

For ABS, again the flat orientation shows the greatest strength with an average of 37,19 MPa. Both oblique and standing orientations were significantly weaker, with an average yield of respectively 12,59 MPa and 13,55 MPa. exept for the flat orientation, these values does not match the given value from the data sheet at 38.1 MPa. Since almost every fracture on the oblique and the standing specimens in ABS were categorized as “AIT” or “LIT”, meaning angular or lateral fractures in the top grip, there is a potential error here. This means that these results are not to be trusted completely. The most probable error source here is poor adhesion between the top layers and possible printing errors.

The results for the E-modulus is evenly significantly lower than the given value. This is because the calculations is simplified due to the stress-strain curve not being linear. Other possible factors of error is the print orientation and the adhesion between layers.

From these results, it is clear that Nylon 12 has the highest and most evenly distributed yield strength of the materials tested. ABS gave the weakest results. Tough PLA has the greatest end most even E-modulus, which indicates that this material has the best resistance against elastic deformation. ABS has the lowest E-modulus in this test and is therefore the material with the least resistance against elastic deformation. The materials yield strength is matching the producers given values quite good, but the E-modulus has greater deviations.

ABS:	**Test number**	**Width (mm)**	**Thickness (mm)**	**Cross section (mm^2)**	**Force (N)**	**Extension (mm)**	**Tension (Mpa)**	**E-modulus (Mpa)**	**Fracture type**
Flat (0)	ABS0_1	10.16	2.94	29.87	1110.00	7.05	37.16	562.06	XGM
	ABS0_2	10.15	2.95	29.94	1089.00	7.36	36.37	582.00	AGM
	ABS0_3	10.12	2.92	29.55	1112.00	7.27	37.63	587.44	LGM
	ABS0_4	10.07	2.95	29.71	1110.00	7.58	37.37	563.72	XGM
	ABS0_5	10.00	2.95	29.50	1131.00	7.63	38.34	692.62	LGM
	ABS0_6	10.02	2.96	29.66	1075.00	6.98	36.25	661.90	AGM
	Average	10.09	2.95	29.70	1104.50	7.31	37.19	608.29	
Oblique (45)	ABS45_1	9.93	3.02	29.99	463.90	5.13	15.47	408.84	AIT
	ABS45_2	9.99	3.03	30.27	481.40	3.50	15.90	578.87	AIT
	ABS45_3	9.99	3.01	30.07	437.20	3.64	14.54	438.04	AIT
	ABS45_4	9.94	2.96	29.42	199.90	1.86	6.79	291.59	AGB
	ABS45_5	9.92	3.00	29.76	363.20	3.37	12.20	327.99	AIT
	ABS45_6	9.97	3.03	30.21	495.90	4.67	16.42	462.70	AIT
	Average	9.96	3.01	29.95	406.92	3.70	13.55	418.01	
Standing (90)	ABS90_1	9.90	2.99	29.60	498.20	4.75	16.83	478.89	LIT
	ABS90_2	10.00	3.02	30.20	267.00	3.93	8.84	296.48	LIT
	ABS90_3	9.99	3.02	30.17	359.30	3.83	11.91	395.91	LIT
	ABS90_4	9.98	3.00	29.94	389.90	4.26	13.02	352.93	LIT
	ABS90_5	10.01	2.98	29.83	401.30	4.08	13.45	416.34	LIT
	ABS90_6	10.02	3.01	30.16	345.60	3.74	11.46	357.46	LIT
	Average	9.98	3.00	29.98	376.88	4.10	12.59	383.00	

Table 3: Results from testing ABS.

Tough PLA:	**Test number**	**Width (mm)**	**Thickness (mm)**	**Cross section (mm^2)**	**Force (N)**	**Extension (mm)**	**Tension (Mpa)**	**E-modulus (Mpa)**	**Fracture type**
Flat (0)	PLA0_1	10.20	2.92	29.78	1587.00	8.04	53.28	870.43	LGM
	PLA0_2	10.10	3.01	30.40	1632.00	8.30	53.68	851.62	LGM
	PLA0_3	10.10	3.00	30.30	1631.00	7.67	53.83	933.17	LGM
	PLA0_4	10.15	3.03	30.75	1608.00	8.06	52.29	907.11	LGM
	PLA0_5	10.16	3.01	30.58	1535.00	7.70	50.19	893.69	LGM
	PLA0_6	10.19	3.04	30.98	1576.00	7.02	50.88	966.45	LGM
	Average	10.15	3.00	30.47	1594.83	7.80	52.36	903.74	
Oblique (45)	PLA45_1	10.11	2.93	29.62	1164.00	6.16	39.29	916.90	AGT
	PLA45_2	10.13	3.08	31.20	1365.00	7.45	43.75	794.94	AIT
	PLA45_3	10.13	3.04	30.80	1398.00	7.24	45.40	898.96	AGM
	PLA45_4	10.12	3.01	30.46	1351.00	7.70	44.35	848.00	AGB
	PLA45_5	10.12	3.02	30.56	1367.00	6.49	44.73	884.72	AGM
	PLA45_6	10.12	3.09	31.27	1289.00	6.56	41.22	864.34	AGT
	Average	10.12	3.03	30.65	1322.33	6.93	43.12	867.98	
Standing (90)	PLA90_1	10.17	2.92	29.70	1003.00	4.93	33.78	815.07	LGM
	PLA90_2	10.04	2.78	27.91	736.20	3.36	26.38	627.83	LGM
	PLA90_3	10.13	2.94	29.78	960.50	4.64	32.25	781.59	LGM
	PLA90_4	10.15	2.99	30.35	1094.00	5.44	36.05	801.35	LGM
	PLA90_5	10.06	2.95	29.68	937.70	5.62	31.60	876.39	LGM
	PLA90_6	10.08	2.95	29.74	1070.00	5.78	35.98	856.36	LTT
	Average	10.11	2.92	29.53	966.90	4.96	32.67	793.10	

Table 4: Results from testing Tough PLA.

Nylon 12:	**Test number**	**Width (mm)**	**Thickness (mm)**	Cross section (mm^2)	Force (N)	Extension (mm)	Tension (Mpa)	E-modulus (Mpa)	Fracture type
Flat (0)	N0_1	10.18	3.11	31.66	1708.00	10.74	53.95	739.63	XGM
	N0_2	10.16	3.13	31.80	1607.00	10.21	50.53	738.10	LGM
	N0_3	10.13	2.97	30.09	1551.00	10.24	51.55	753.63	XGM
	N0_4	10.18	3.16	32.17	1675.00	10.21	52.07	772.58	XGM
	N0_5	10.19	3.11	31.69	1622.00	10.44	51.18	678.35	LGM
	Average	10.17	3.10	31.48	1632.60	10.37	51.86	736.46	
Oblique (45)	N45_1	10.11	2.97	30.03	1398.00	8.43	46.56	441.94	LGM
	N45_2	10.13	2.97	30.09	1478.00	10.26	49.13	383.05	AGM
	N45_3	10.13	2.89	29.28	1399.00	8.57	47.79	446.30	LGM
	N45_4	10.16	2.92	29.67	1507.00	10.49	50.80	387.39	XTB
	N45_5	10.10	2.93	29.59	1493.00	9.75	50.45	414.00	XGM
	Average	10.13	2.94	29.73	1455.00	9.50	48.94	414.54	

Table 5: Results from testing Nylon 12.

Appendix C

Matlab scripts

A Limit switch location Matlab

This is the Matlab script to plot the location of the limit switch sensors around the Effector gear wheel.

```matlab
% Date created: 17.04.2023
% Author: Marius Heistad
% Scope: The purpose of this script is to plot the location of the
% limit swithces around the Effectors gear wheel. This is in order
% to get the correct positions for a zero-point, and overtravel

% Clear command window
clc;
clear all;

% Variables
rotasjon_grader = 360;

% Calculate relationship between motor and Effector gear wheel
dia_motor_hjul = 30.94; %mm
dia_effektor_hjul = 99; %mm

forhold_motor_effektor = dia_effektor_hjul / dia_motor_hjul;
forhold_andre_vei = dia_motor_hjul / dia_effektor_hjul;

% Number of rotations on effector with regards to motor
rot_eff_forhold = rotasjon_grader / forhold_motor_effektor;

% Positions for over-travel limit switches;
posisjon_lim_ende_pluss = (4 * rot_eff_forhold);
posisjon_lim_ende_minus = (4* (-rot_eff_forhold));
posisjon_zero = 0;

% Converting to radians for plotting.
posisjon_lim_ende_pluss_rad = deg2rad(posisjon_lim_ende_pluss);
posisjon_lim_ende_minus_rad = deg2rad(posisjon_lim_ende_minus);
posisjon_zero_rad = deg2rad(posisjon_zero);

% Values used when plotting:
pos2plot = [posisjon_lim_ende_pluss_rad posisjon_zero_rad posisjon_lim_ende_minus_rad];

% Plotting of the limit switch positions.
```

```
figure(1)
polarplot(pos2plot , 1 , "o")
set(gca,'ThetaZeroLocation','top')
rlim([0 1.5])
title('Position of limit switch')
```

B Cam profile script Matlab

This is the Matlab script to plot the cam profile and the resulting movement for the follower.

```
% Date Created: 13.04.2023
% Author: Marius Heistad
% Original script by Xdynemo:
% https://xdynemo.wordpress.com/design-of-cam-profile-in-matlab/
% The purpose of this matlab script is to create a cam profile
% by using a set of input data. The resulting profile is then plotted
% and a follower motion movement is also plotted.


% Clear command window
clc;
clear all;

% Lift and Base Circle radius in mm.
height = input('Lift of Follower (mm) \n');
radius = input('Base Radius (mm) \n');

% Ascent and Descent Angles
ascent = input('Ascent Angle \n');
dwell_1 = input('Dwell Angle \n');
descent = input('Descent Angle \n');
dwell_2 = 360-(ascent + dwell_1 + descent);

% Additional angles for our convenience
after_ascent = ascent + dwell_1;
after_descent = after_ascent + descent;

% Cam angles
theta = linspace(0, 360, 361);

% Formula for SHM - Simple Harmonic Motion. Uniform Velocity, Cycloidal
% Adding Motion Selector Window
motion_list = {'Simple Harmonic Motion','Uniform Velocity','Cycloidal'};
  [ascent_motion] = listdlg('ListSize',[270,100],'Name','Select Ascent Motion','SelectionMode','single','ListString',motion_list);
  [descent_motion] = listdlg('ListSize',[270,100],'Name','Select Descent Motion','SelectionMode','single','ListString',motion_list);
```

```

% Ascent Motion Conditions
if ascent_motion == 1 % SHM
    h_ascent = (0.5*height).*(1-cosd((180/ascent).*theta(
       theta<ascent)));
elseif ascent_motion == 2 % Uniform Velocity
    h_ascent = (height/ascent).*theta(theta<ascent);
elseif ascent_motion == 3 % Cycloidal
    h_ascent = (height/pi)*(((pi/ascent).*theta(theta<ascent)
       ) - 0.5*sind((2*180/ascent).*theta(theta<ascent)));
end


% Descent Motion Conditions
if descent_motion == 1 % SHM
    h_descent = height - ((0.5*height).*(1 - cosd((180/
       descent).*theta(theta<=descent))));
elseif descent_motion == 2 % Uniform Velocity
    h_descent = height - (height/descent).*theta(theta<=
       descent);
elseif descent_motion == 3 % Cycloidal
    h_descent = height - ((height/pi)*(((pi/descent).*theta(
       theta<=descent)) - 0.5*sind((2*180/descent).*theta(
       theta<=descent))));
end

% Plotting Cam Angle vs Lift
plot(theta(theta<ascent),h_ascent,theta(theta>=after_ascent &
    theta<=after_descent),h_descent);
set(gca,'XTick',(0:10:after_descent))
set(gca,'YTick',(0:5:height))
set(gcf,'position',[0,0,6000,400])
title('Cam Angle Vs Lift')
xlabel('Cam Angle (degrees)');
ylabel('Lift of Follower (mm)');

% Lift during dwell
h_dwell_1 = ones(1, dwell_1).*height;
h_dwell_2 = zeros(1, dwell_2);

% Defining radii during different phases of Cam
r1 = radius + h_ascent;
r2 = radius + h_dwell_1;
r3 = radius + h_descent;
r4 = radius + h_dwell_2;

% Joining all radii
radius = [r1 r2 r3 r4];
```

```

% Convert theta to radians
theta_radians = deg2rad(theta);

% Plotting Cam Profile
figure
polarplot(theta_radians,radius);
set(gca,'ThetaZeroLocation','top')

% Converting Polar to Cartesian Coordinate System
[x,y] = pol2cart(theta_radians,radius);
x_cord = transpose(x);
y_cord = transpose(y);
z_cord = zeros(361,1);

cam_profile = [x_cord, y_cord, z_cord];

% Export Cam Profile to Excel as XYZ Coordinates
writematrix(cam_profile,'
    cam_profile_big_51_dia_3_lift_harmonic_test.xls')
```

C Matlab script for vibrational analysis

This is the Matlab script to perform the spectral analysis for the vibrational analysis.

```
clear
clc
close all;

% Date created: 11.05.2023
% Author: Stefan Theodor Zurbuchen Hansen
% Scope: The purpose of this script is to perform a spectral analysis for
% the vibrational analysis performed by project Mini Guardian

% Specify the path to the .wav file
vibration_test = 'C:\Users\Stefan\Desktop\VibrationAnalysis\Pos1\Pos1_0-30.wav';

% Read the .wav file, y = audio data and fs = sampling frequency
[y, fs] = audioread(vibration_test);

% View only channel 1
y = y(:,1);

% See max value of y, see if clipped signal
max_y = max(abs(y));

% Perform spectral analysis
N = length(y);      % Length of the signal
Nsec = N / fs;      % Duration of the signal in seconds
t = (0:N-1)/fs;     % Time vector

% Window
Win = 4*512;
Noverlap = fix(Win/2);

% Reduce datapoints for more efficient runtime (avoid killing my pc)
fsRed = 0:10:fs/2;

% Compute the FFT
Y = fft(y);

% Compute the single-sided spectrum
P = abs(Y/N);
P = P(1:N/2+1);
P(2:end-1) = 2*P(2:end-1);

```

```
% Frequency vector
f = fs*(0:(N/2))/N;

% Prepare the spectral analysis variables
[y0G,f0G,t0G,p0G] = spectrogram(y, Win, Noverlap, fsRed, N, 'yaxis');

%% Plot the spectrum
figure(1)
plot(f, abs(P)/max(abs(P)))
title('Single-Sided Amplitude Spectrum', 'FontSize', 14)
xlabel('Frequency (Hz)', 'FontSize', 12)
ylabel('Normalized Amplitude', 'FontSize', 12)
% Zoom into the interesting frequency range
min_freq = 0;
max_freq = 4000;
xlim([min_freq max_freq])

%% Plotting the spectral analysis:
figure(2)
subplot(2,1,1)
xaxisP1 = linspace(0, Nsec, N);
plot(xaxisP1, y)
title('Time-domain signal', 'FontSize', 14)
grid on
xlabel('Time (s)', 'FontSize', 12)
ylabel('Amplitude', 'FontSize', 12)

figure(2)
subplot(2,1,2)
surf(t0G*10, f0G/1e4, 20*log10(abs(p0G)), 'EdgeColor', 'none');
title('(c)', 'FontSize', 14)
axis xy; axis tight; colormap(jet); view(0, 90);
title('Spectral analysis', 'FontSize', 14)
xlabel('Time (s)', 'FontSize', 12);
ylabel('Frequency (kHz)', 'FontSize', 12);

%% Save figures
% saveas(figure(1), 'AmplitudeSpectrum_Pos1_0-30.jpg')
% saveas(figure(2), 'SpectralAnalysis_Pos1_0-30.jpg')
```

D Azimuth plotting script

This is the Matlab script to plot recorded measurements of the accuracy on the azimuth axis on the Effector

```matlab
% Date Created: 15.03.2023
% Author: Daniel Solbrekke
% The purpose of this matlab script is to plot the measured accuracy of the
% azimuth axis on the Effector. The data is extracted from the excel sheets
% used for data recording.


clear; close all; clc;

%% ORIGINAL
%%% Negative rotation to positive rotation (Negative to positive)
% Goal
raw_N2P = readtable('..\Measurement Azimuth Effector.xlsx','sheet','Before', 'Range', 'K12:K52');
N2P = table2array(raw_N2P);

% GUI reading
raw_OrgN2PGUI = readtable('..\Measurement Azimuth Effector.xlsx','sheet','Before', 'Range', 'J12:J52');
OrgN2PGUI = table2array(raw_OrgN2PGUI);

% Actual position
raw_OrgN2PActual = readtable('..\Measurement Azimuth Effector.xlsx','sheet','Before', 'Range', 'L12:L52');
OrgN2PActual = table2array(raw_OrgN2PActual);

% Error between goal and actual
raw_OrgN2PError = readtable('..\Measurement Azimuth Effector.xlsx', 'sheet', 'Before', 'Range', 'M12:M52');
OrgN2PError = table2array(raw_OrgN2PError);

%%%% Positive rotation to negative rotation (Positive to negative)
% Goal
raw_P2N = readtable('..\Measurement Azimuth Effector.xlsx','sheet','Before', 'Range', 'Q12:Q52');
P2N = table2array(raw_P2N);

% GUI reading
raw_OrgP2NGUI = readtable('..\Measurement Azimuth Effector.xlsx','sheet','Before', 'Range', 'P12:P52');
```

```
OrgP2NGUI = table2array(raw_OrgP2NGUI);

% Actual position
raw_OrgP2NActual = readtable('..\Measurement Azimuth Effector.xlsx','sheet','Before', 'Range', 'R12:R52');
OrgP2NActual = table2array(raw_OrgP2NActual);

% Error between goal and actual
raw_OrgP2NError = readtable('..\Measurement Azimuth Effector.xlsx', 'sheet', 'Before', 'Range', 'S12:S52');
OrgP2NError = table2array(raw_OrgP2NError);

%% Improved
%%% Negative rotation to positive rotation (Negative to positive)
% Goal
raw_N2P = readtable('..\Measurement Azimuth Effector.xlsx','sheet','After', 'Range', 'K12:K52');
N2P = table2array(raw_N2P);

% GUI reading
raw_ImpN2PGUI = readtable('..\Measurement Azimuth Effector.xlsx','sheet','After', 'Range', 'J12:J52');
ImpN2PGUI = table2array(raw_ImpN2PGUI);

% Actual position
raw_ImpN2PActual = readtable('..\Measurement Azimuth Effector.xlsx','sheet','After', 'Range', 'L12:L52');
ImpN2PActual = table2array(raw_ImpN2PActual);

% Error between goal and actual
raw_ImpN2PError = readtable('..\Measurement Azimuth Effector.xlsx', 'sheet', 'After', 'Range', 'M12:M52');
ImpN2PError = table2array(raw_ImpN2PError);

%%%% Positive rotation to negative rotation (Positive to negative)
% Goal
raw_P2N = readtable('..\Measurement Azimuth Effector.xlsx','sheet','After', 'Range', 'Q12:Q52');
P2N = table2array(raw_P2N);

% GUI reading
raw_ImpP2NGUI = readtable('..\Measurement Azimuth Effector.xlsx','sheet','After', 'Range', 'P12:P52');
ImpP2NGUI = table2array(raw_ImpP2NGUI);

% Actual position
```

```
raw_ImpP2NActual = readtable('..\Measurement Azimuth Effector.xlsx','sheet','After', 'Range', 'R12:R52');
ImpP2NActual = table2array(raw_ImpP2NActual);

% Error between goal and actual
raw_ImpP2NError = readtable('..\Measurement Azimuth Effector.xlsx', 'sheet', 'After', 'Range', 'S12:S52');
ImpP2NError = table2array(raw_ImpP2NError);

clearvars raw*

%% Plot setup

x_label = 'Desired position [deg]';
y_label = 'Measured position [deg]';
y_label_error = 'Error [deg]';

font_size = 15;

%% Plot original

% Plot original negative to positive rotation
figure(1)
plot(N2P, OrgN2PActual, '-b', 'LineWidth', 1.5)
hold on
plot(N2P, OrgN2PGUI, '-r', 'LineWidth', 1.5)
xlabel(x_label, 'Interpreter', 'latex', 'FontSize', font_size)
ylabel(y_label, 'Interpreter', 'latex', 'FontSize', font_size)
legend({'Actual', 'GUI'}, 'Interpreter','latex', 'Location', 'northwest')
grid on
saveas(gcf,'fig\Azimuth\Original\AZ_OrgN2P.png')

% Plot original positive to negative rotation
figure(2)
plot(P2N, OrgP2NActual, '-b', 'LineWidth', 1.5)
hold on
plot(P2N, OrgP2NGUI, '-r', 'LineWidth', 1.5)
xlabel(x_label, 'Interpreter', 'latex', 'FontSize', font_size)
ylabel(y_label, 'Interpreter', 'latex', 'FontSize', font_size)
legend({'Actual', 'GUI'}, 'Interpreter','latex', 'Location', 'northwest')
grid on
saveas(gcf,'fig\Azimuth\Original\AZ_OrgP2N.png')
```

```

% Plot error of original negative to positive rotation
figure(3)
area(N2P, OrgN2PError)
xlabel(x_label, 'Interpreter', 'latex', 'FontSize', font_size)
ylabel(y_label_error, 'Interpreter', 'latex', 'FontSize', font_size)
saveas(gcf,'fig\Azimuth\Original\AZ_OrgN2P_Error.png')

% Plot error of original positive to negative rotation
figure(4)
area(P2N, OrgP2NError)
xlabel(x_label, 'Interpreter', 'latex', 'FontSize', font_size)
ylabel(y_label_error, 'Interpreter', 'latex', 'FontSize', font_size)
saveas(gcf,'fig\Azimuth\Original\AZ_OrgP2N_Error.png')

%% Plot improved

% Plot original negative to positive rotation
figure(5)
plot(N2P, ImpN2PActual, '-b', 'LineWidth', 1.5)
hold on
plot(N2P, ImpN2PGUI, '-r', 'LineWidth', 1.5)
xlabel(x_label, 'Interpreter', 'latex', 'FontSize', font_size)
ylabel(y_label, 'Interpreter', 'latex', 'FontSize', font_size)
legend({'Actual', 'GUI'}, 'Interpreter','latex', 'Location', 'northwest')
grid on
saveas(gcf,'fig\Azimuth\Improved\AZ_ImpN2P.png')

% Plot original positive to negative rotation
figure(6)
plot(P2N, ImpP2NActual, '-b', 'LineWidth', 1.5)
hold on
plot(P2N, ImpP2NGUI, '-r', 'LineWidth', 1.5)
xlabel(x_label, 'Interpreter', 'latex', 'FontSize', font_size)
ylabel(y_label, 'Interpreter', 'latex', 'FontSize', font_size)
legend({'Actual', 'GUI'}, 'Interpreter','latex', 'Location', 'northwest')
grid on
saveas(gcf,'fig\Azimuth\Improved\AZ_ImpP2N.png')
```

```

% Plot error of original negative to positive rotation
figure(7)
area(N2P, ImpN2PError)
xlabel(x_label, 'Interpreter', 'latex', 'FontSize', font_size)
ylabel(y_label_error, 'Interpreter', 'latex', 'FontSize', font_size)
saveas(gcf,'fig\Azimuth\Improved\AZ_ImpN2P_Error.png')

% Plot error of original positive to negative rotation
figure(8)
area(P2N, ImpP2NError)
xlabel(x_label, 'Interpreter', 'latex', 'FontSize', font_size)
ylabel(y_label_error, 'Interpreter', 'latex', 'FontSize', font_size)
saveas(gcf,'fig\Azimuth\Improved\AZ_ImpP2N_Error.png')
```

E Elevation plotting script

This is the Matlab script to plot recorded measurements of the accuracy on the elevation axis on the Effector

```
% Date Created: 15.03.2023
% Author: Daniel Solbrekke
% Modified: Marius Heistad 19.05.2023
% The purpose of this matlab script is to plot the measured accuracy of the
% elevation axis on the Effector. The data is extracted from the excel sheets
% used for data recording.

clear; close all; clc;

%% Original
%%% Zero elevation to top elevation (Zero to top)
% Goal
raw_Z2T = readtable('..\Measurement Elevation Effector.xlsx','sheet','Before', 'Range', 'M17:M46');
Z2T = table2array(raw_Z2T);

% GUI reading
raw_OrgZ2TGUI = readtable('..\Measurement Elevation Effector.xlsx','sheet','Before', 'Range', 'L17:L46');
OrgZ2TGUI = table2array(raw_OrgZ2TGUI);

% Actual position
raw_OrgZ2TActual = readtable('..\Measurement Elevation Effector.xlsx','sheet','Before', 'Range', 'N17:N46');
OrgZ2TActual = table2array(raw_OrgZ2TActual);

% Error between goal and actual
raw_OrgZ2TError = readtable('..\Measurement Elevation Effector.xlsx', 'sheet', 'Before', 'Range', 'O17:O46');
OrgZ2TError = table2array(raw_OrgZ2TError);

%%%% Top elevation to zero elevation (Top to zero)
% Goal
raw_T2Z = readtable('..\Measurement Elevation Effector.xlsx','sheet','Before', 'Range', 'S17:S46');
T2Z = table2array(raw_T2Z);

% GUI reading
raw_OrgT2ZGUI = readtable('..\Measurement Elevation Effector.xlsx','sheet','Before', 'Range', 'R17:R46');
OrgT2ZGUI = table2array(raw_OrgT2ZGUI);

```

```
% Actual position
raw_OrgT2ZActual = readtable('..\Measurement Elevation Effector.xlsx','sheet','Before', 'Range', 'T17:T46');
OrgT2ZActual = table2array(raw_OrgT2ZActual);

% Error between goal and actual
raw_OrgT2ZError = readtable('..\Measurement Elevation Effector.xlsx', 'sheet', 'Before', 'Range', 'U17:U46');
OrgT2ZError = table2array(raw_OrgT2ZError);


%% Improved
% Goal
raw_Z2T = readtable('..\Measurement Elevation Effector.xlsx','sheet','After', 'Range', 'M17:M46');
Z2T = table2array(raw_Z2T);

% GUI reading
raw_ImpZ2TGUI = readtable('..\Measurement Elevation Effector.xlsx','sheet','After', 'Range', 'L17:L46');
ImpZ2TGUI = table2array(raw_ImpZ2TGUI);

% Actual position
raw_ImpZ2TActual = readtable('..\Measurement Elevation Effector.xlsx','sheet','After', 'Range', 'N17:N46');
ImpZ2TActual = table2array(raw_ImpZ2TActual);

% Error between goal and actual
raw_ImpZ2TError = readtable('..\Measurement Elevation Effector.xlsx', 'sheet', 'After', 'Range', 'O17:O46');
ImpZ2TError = table2array(raw_ImpZ2TError);

%%%% Top elevation to zero elevation (Top to zero)
% Goal
raw_T2Z = readtable('..\Measurement Elevation Effector.xlsx','sheet','After', 'Range', 'S17:S46');
T2Z = table2array(raw_T2Z);

% GUI reading
raw_ImpT2ZGUI = readtable('..\Measurement Elevation Effector.xlsx','sheet','After', 'Range', 'R17:R46');
ImpT2ZGUI = table2array(raw_ImpT2ZGUI);

% Actual position
raw_ImpT2ZActual = readtable('..\Measurement Elevation Effector.xlsx','sheet','After', 'Range', 'T17:T46');
ImpT2ZActual = table2array(raw_ImpT2ZActual);

```

```
% Error between goal and actual
raw_ImpT2ZError = readtable('..\Measurement Elevation Effector.xlsx', 'sheet', 'After', 'Range', 'U17:U46');
ImpT2ZError = table2array(raw_ImpT2ZError);

clearvars raw*

%% Plot setup

x_label = 'Desired position [deg]';
y_label = 'Measured position [deg]';
y_label_error = 'Error [deg]';

font_size = 15;

%% Plot original

% Plot original zero to top elevation
figure(1)
plot(Z2T, OrgZ2TActual, '-b', 'LineWidth', 1.5)
hold on
plot(Z2T, OrgZ2TGUI, '-r', 'LineWidth', 1.5)
xlabel(x_label, 'Interpreter', 'latex', 'FontSize', font_size)
ylabel(y_label, 'Interpreter', 'latex', 'FontSize', font_size)
legend({'Actual', 'GUI'}, 'Interpreter','latex', 'Location', 'northwest')
grid on
saveas(gcf,'fig\Elevation\Original\EL_OrgZ2T.png')

% Plot original top to zero elevation
figure(2)
plot(T2Z, OrgT2ZActual, '-b', 'LineWidth', 1.5)
hold on
plot(T2Z, OrgT2ZGUI, '-r', 'LineWidth', 1.5)
xlabel(x_label, 'Interpreter', 'latex', 'FontSize', font_size)
ylabel(y_label, 'Interpreter', 'latex', 'FontSize', font_size)
legend({'Actual', 'GUI'}, 'Interpreter','latex', 'Location', 'northwest')
grid on
saveas(gcf,'fig\Elevation\Original\EL_OrgT2Z.png')

% Plot error of original zero to top elevation
figure(3)
area(Z2T, OrgZ2TError)
```

```
xlabel(x_label, 'Interpreter', 'latex', 'FontSize', font_size)
ylabel(y_label_error, 'Interpreter', 'latex', 'FontSize', font_size)
saveas(gcf,'fig\Elevation\Original\EL_OrgZ2T_Error.png')

% Plot error of original zero to top elevation
figure(4)
area(T2Z, OrgT2ZError)
xlabel(x_label, 'Interpreter', 'latex', 'FontSize', font_size)
ylabel(y_label_error, 'Interpreter', 'latex', 'FontSize', font_size)
saveas(gcf,'fig\Elevation\Original\EL_OrgT2Z_Error.png')

%% Plot improved

% Plot improved zero to top elevation
figure(5)
plot(Z2T, ImpZ2TActual, '-b', 'LineWidth', 1.5)
hold on
plot(Z2T, ImpZ2TGUI, '-r', 'LineWidth', 1.5)
xlabel(x_label, 'Interpreter', 'latex', 'FontSize', font_size)
ylabel(y_label, 'Interpreter', 'latex', 'FontSize', font_size)
legend({'Actual', 'GUI'}, 'Interpreter','latex', 'Location', 'northwest')
grid on
saveas(gcf,'fig\Elevation\Improved\EL_ImpZ2T.png')

% Plot improved top to zero elevation
figure(6)
plot(T2Z, ImpT2ZActual, '-b', 'LineWidth', 1.5)
hold on
plot(T2Z, ImpT2ZGUI, '-r', 'LineWidth', 1.5)
xlabel(x_label, 'Interpreter', 'latex', 'FontSize', font_size)
ylabel(y_label, 'Interpreter', 'latex', 'FontSize', font_size)
legend({'Actual', 'GUI'}, 'Interpreter','latex', 'Location', 'northwest')
grid on
saveas(gcf,'fig\Elevation\Improved\EL_ImpT2Z.png')

% Plot error of improved zero to top elevation
figure(7)
area(Z2T, ImpZ2TError)
```

```
xlabel(x_label, 'Interpreter', 'latex', 'FontSize', font_size
   )
ylabel(y_label_error, 'Interpreter', 'latex', 'FontSize',
   font_size)
saveas(gcf,'fig\Elevation\Improved\EL_ImpZ2T_Error.png')

% Plot error of improved zero to top elevation
figure(8)
area(T2Z, ImpT2ZError)
xlabel(x_label, 'Interpreter', 'latex', 'FontSize', font_size
   )
ylabel(y_label_error, 'Interpreter', 'latex', 'FontSize',
   font_size)
saveas(gcf,'fig\Elevation\Improved\EL_ImpT2Z_Error.png')
```

F Stepper

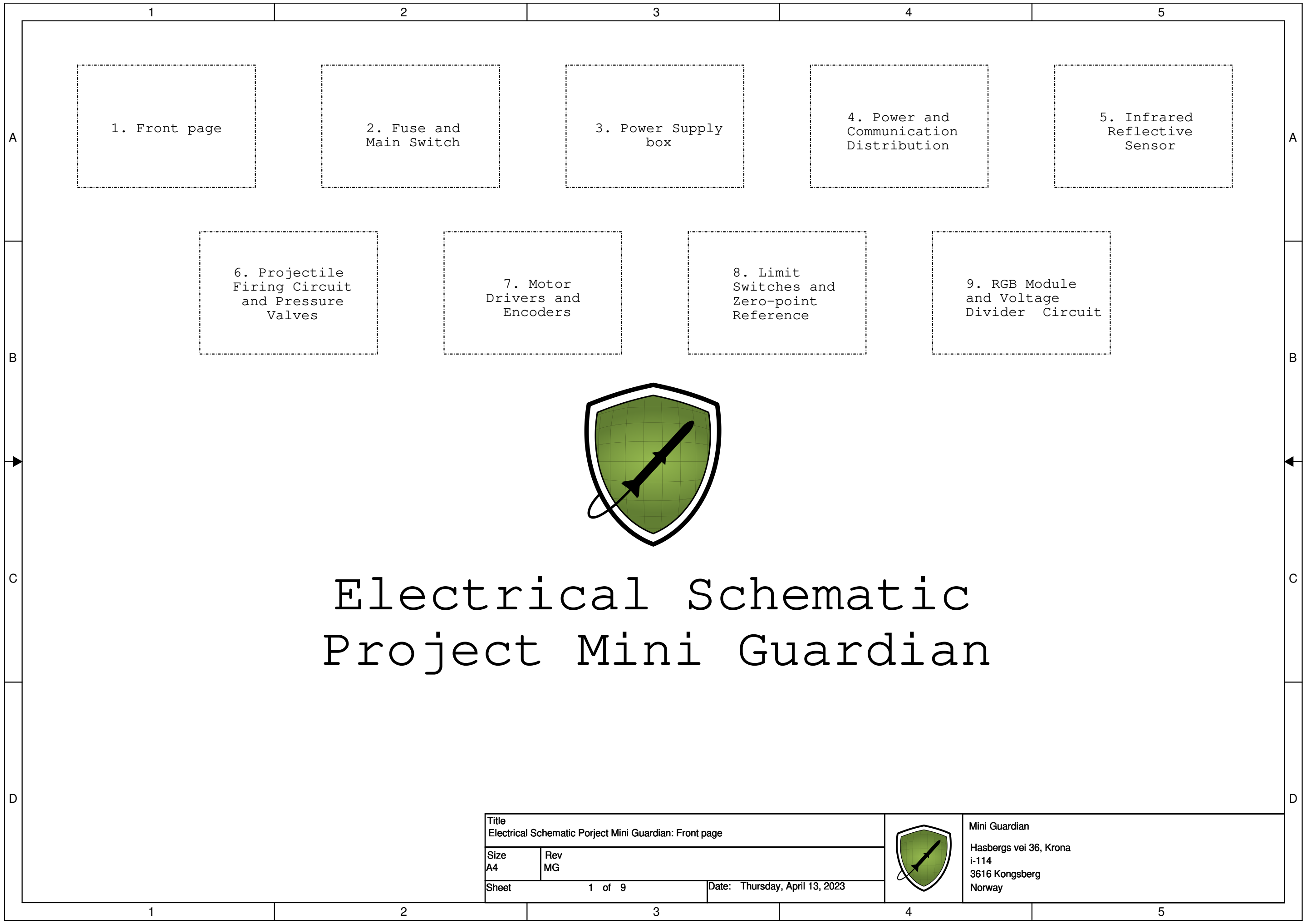
This is a script for implementing data to stepper motors.

```
% Date Created: 13.03.2023
% Author: Daniel Solbrekke

time    = [0 2 4 6 8];      % Seconds
pos     = [0 10 30 10 170];  % Degrees
% 10 -> 170[deg] = 160[deg] = 80[deg/s] > max deg per second

RA      = 3.1997; % Gear Ratio
D2S     = 8/1.8;  % Convert from degrees to steps required for effector to move desired amount of degrees


% Azimuth stepper constants
AZ_acc      = 700;  % Step/s^2
%AZ_speed    = 1500; % Step/s
% Measured to 72 degrees/second = 0.2 rotasjon/sekund = 12 rpm
% 72[degree/s]*(8/1.8)[steps/degree]*3.1997[Gear ratio] = 1024 steps/second
AZ_speed    = 1024; % Step/s

% Azimuth stepper constants
EL_acc      = 800;  % Step/s^2
EL_speed    = 800;  % Step/s


% https://stackoverflow.com/questions/44571660/is-it-possible-to-implement-a-pid-controller-in-simscape-branch-of-simulink
% https://se.mathworks.com/help/ident/gs/identify-linear-models-using-the-gui.html
```

Appendix D

Wiring Diagram

A Wiring diagram


1
2
3
4
5
A
B
C
D
1. Front page
2. Fuse and Main Switch
3. Power Supply box
4. Power and Communication Distribution
5. Infrared Reflective Sensor
6. Projectile Firing Circuit and Pressure Valves
7. Motor Drivers and Encoders
8. Limit Switches and Zero-point Reference
9. RGB Module and Voltage Divider Circuit

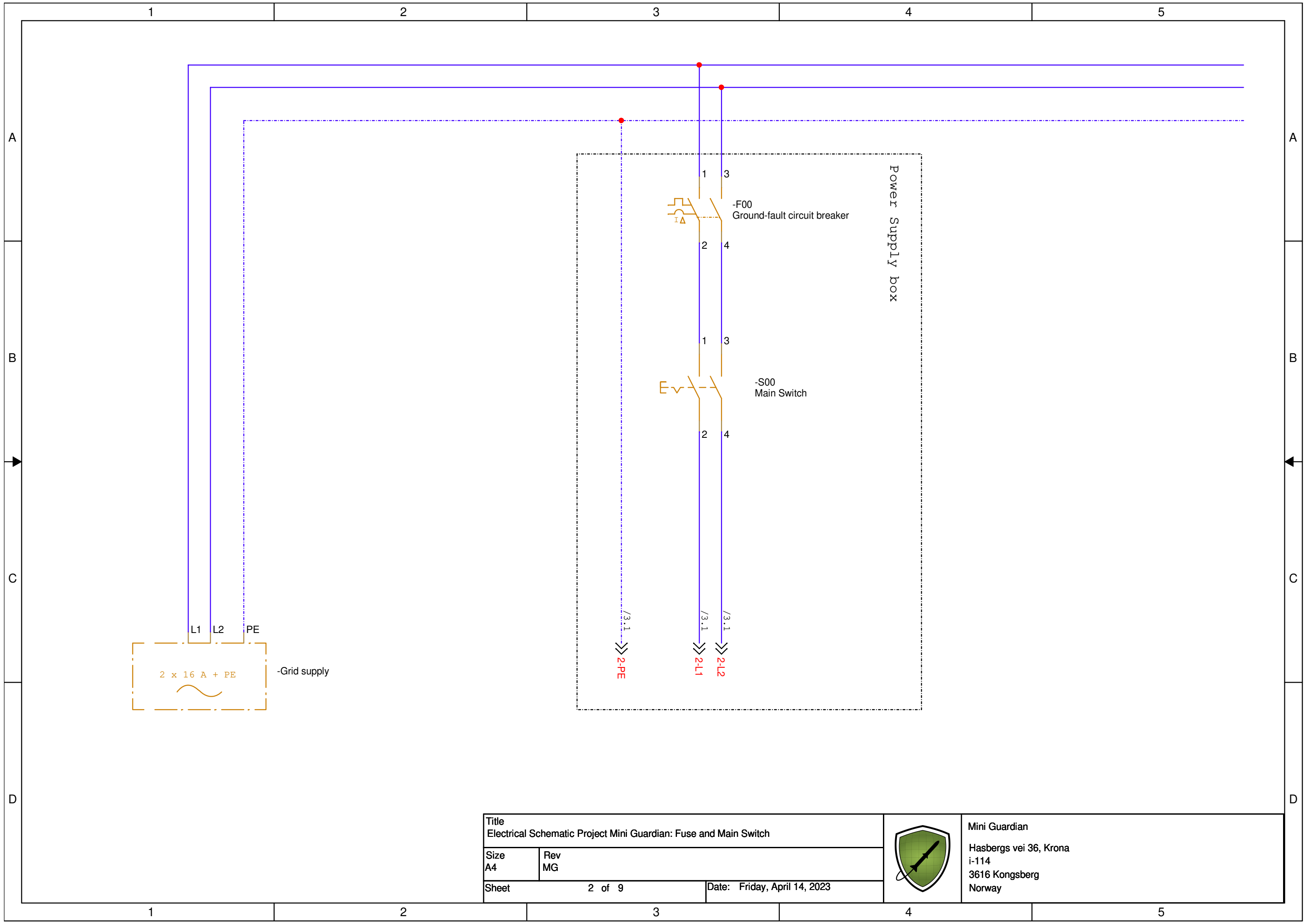
Electrical Schematic
Project Mini Guardian

Title
Electrical Schematic Porject Mini Guardian: Front page

Size A4
Rev MG

Sheet 1 of 9
Date: Thursday, April 13, 2023

Mini Guardian
Hasbergs vei 36, Krona
i-114
3616 Kongsberg
Norway



Title
Electrical Schematic Project Mini Guardian: Fuse and Main Switch

Size
A4

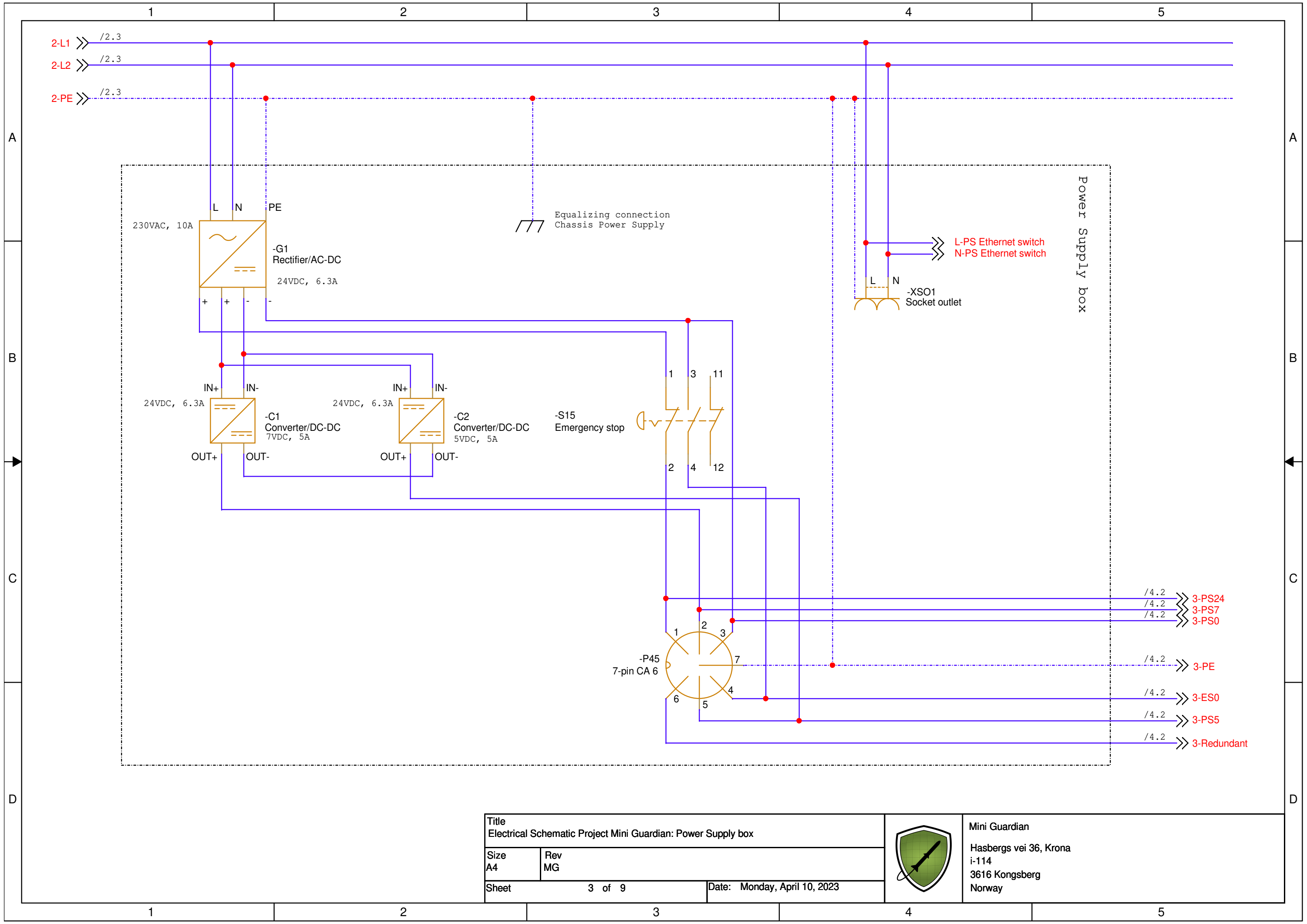
Rev
MG

Sheet 2 of 9

Date: Friday, April 14, 2023

Mini Guardian

Hasbergs vei 36, Krona
i-114
3616 Kongsberg
Norway



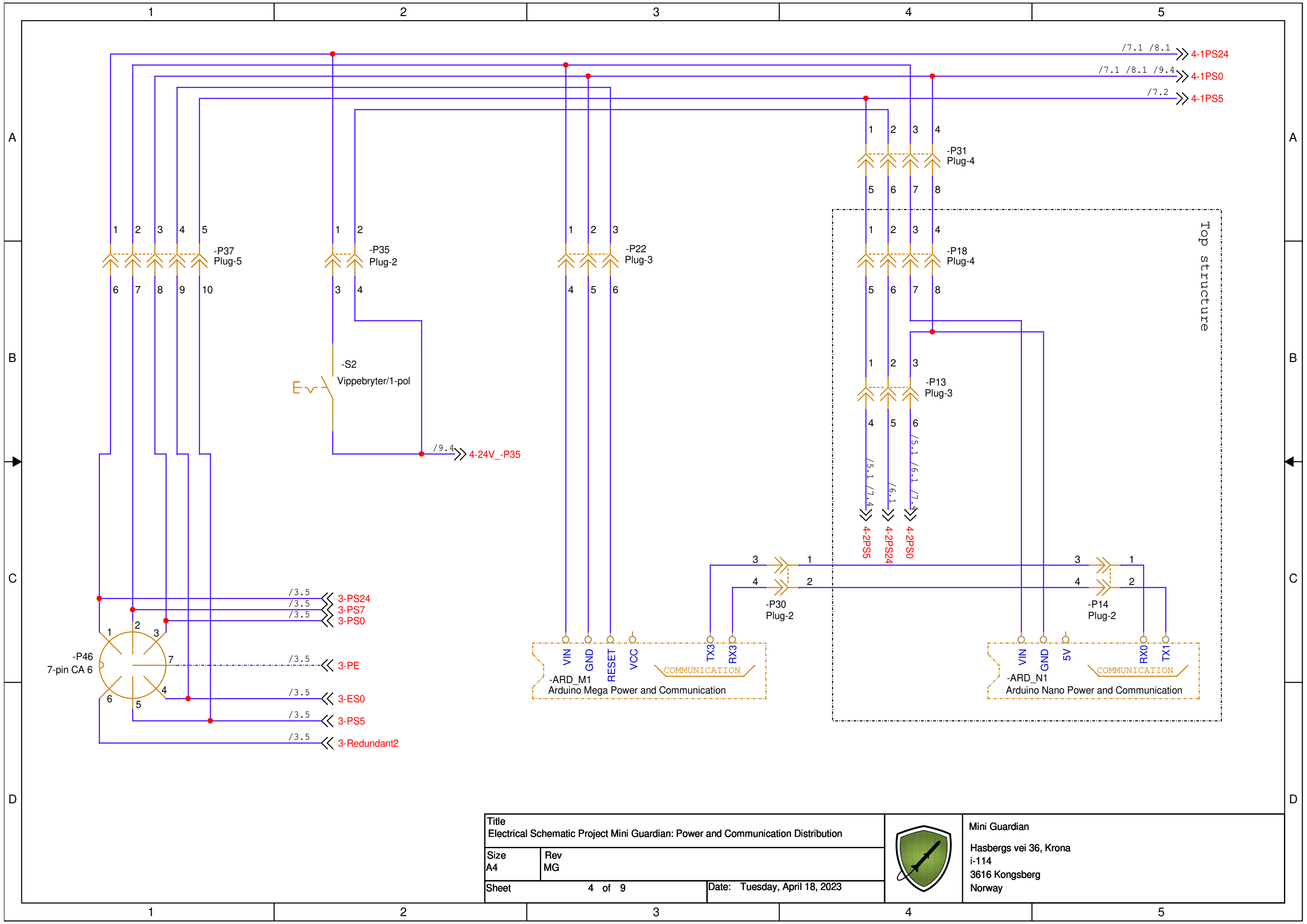
Title
Electrical Schematic Project Mini Guardian: Power Supply box

Size A4
Rev MG

Sheet 3 of 9
Date: Monday, April 10, 2023

Mini Guardian

Hasbergs vei 36, Krona
i-114
3616 Kongsberg
Norway



Title
Electrical Schematic Project Mini Guardian: Power and Communication Distribution

Size
A4

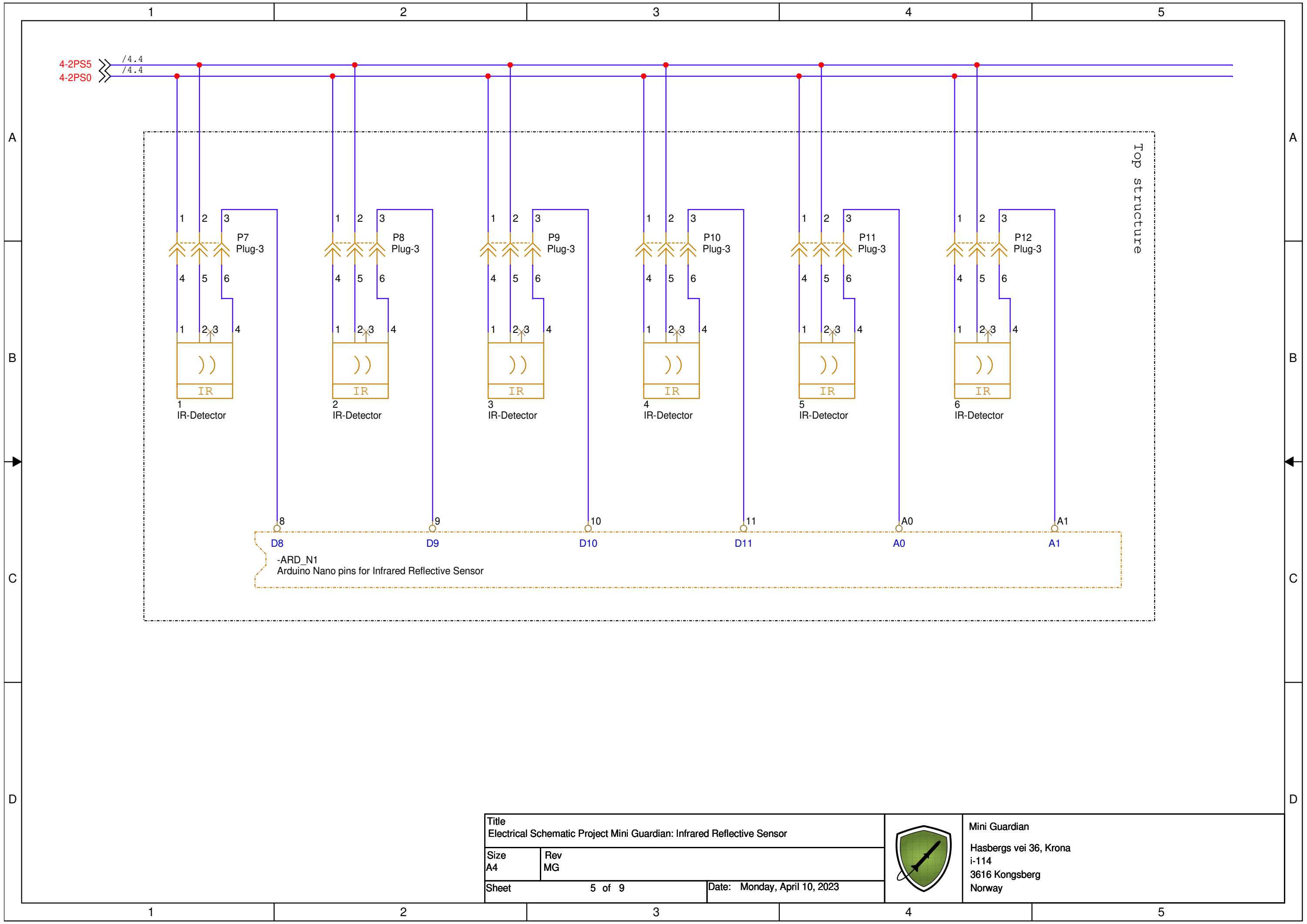
Rev
MG

Sheet 4 of 9

Date: Tuesday, April 18, 2023

Mini Guardian

Hasbergs vei 36, Krona
i-114
3616 Kongsberg
Norway



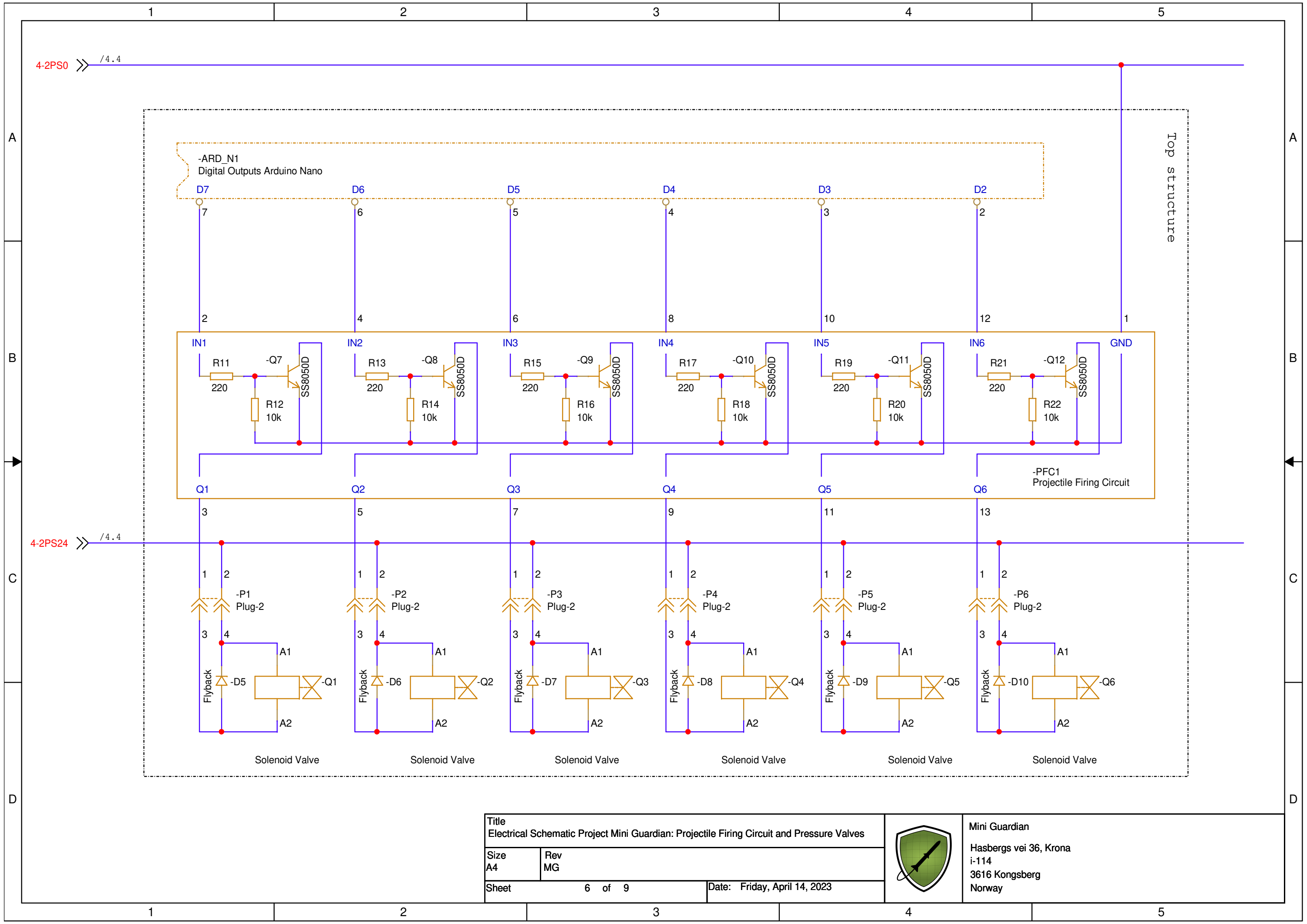
Title
Electrical Schematic Project Mini Guardian: Infrared Reflective Sensor

Size A4 | Rev MG

Sheet 5 of 9 | Date: Monday, April 10, 2023

Mini Guardian

Hasbergs vei 36, Krona
i-114
3616 Kongsberg
Norway



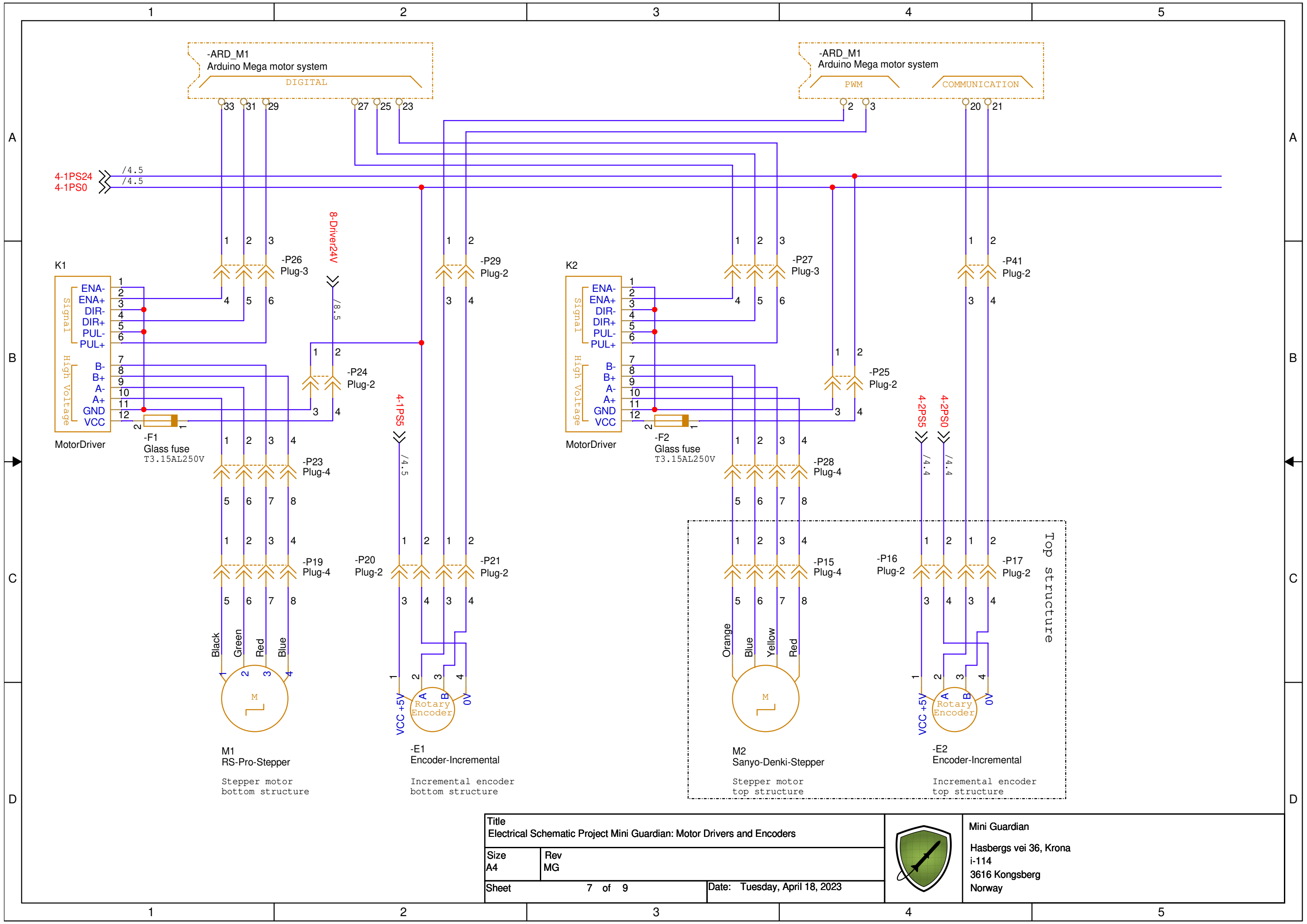
Title
Electrical Schematic Project Mini Guardian: Projectile Firing Circuit and Pressure Valves

Size A4 | Rev MG

Sheet 6 of 9 | Date: Friday, April 14, 2023

Mini Guardian

Hasbergs vei 36, Krona
i-114
3616 Kongsberg
Norway



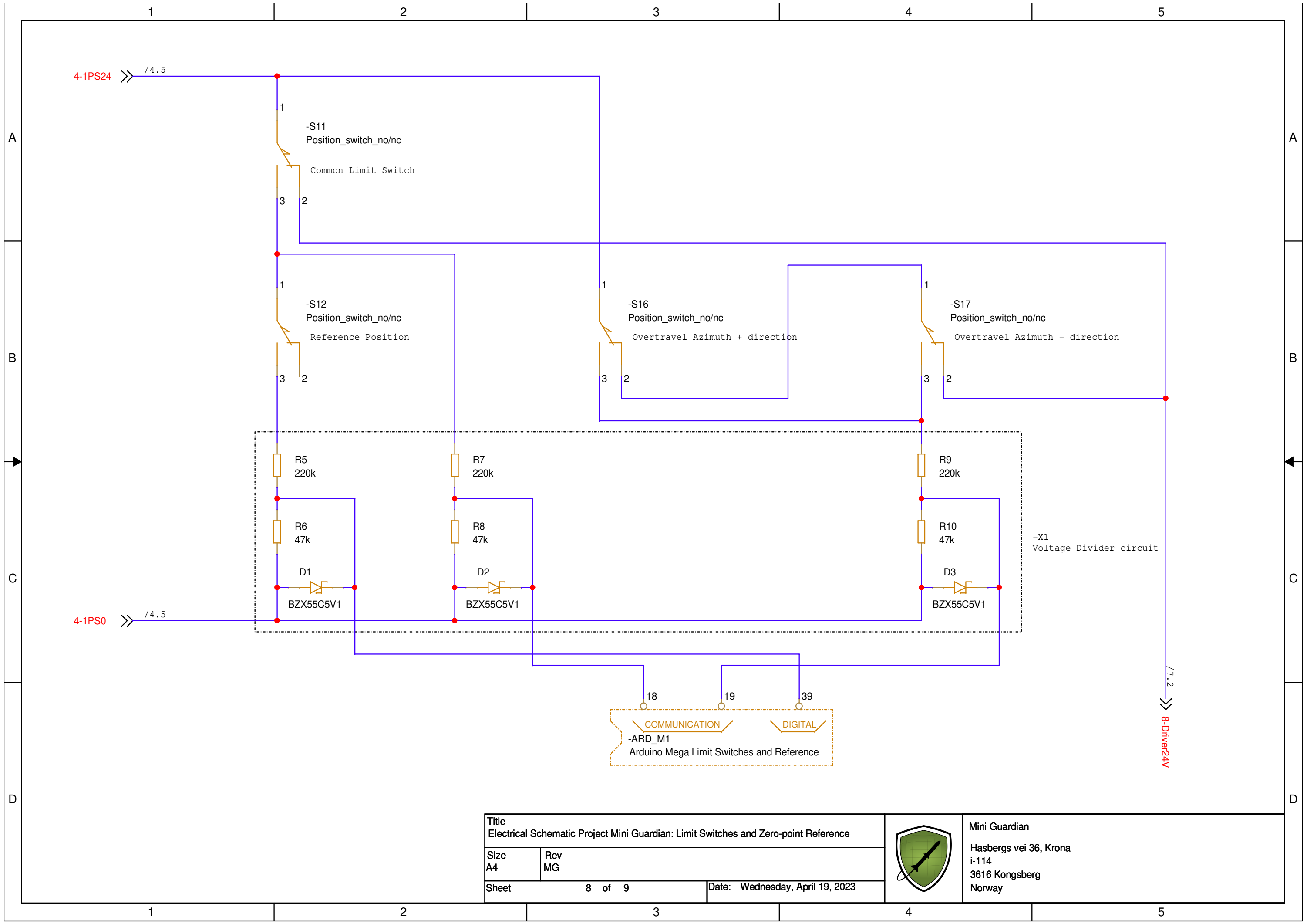
Title
Electrical Schematic Project Mini Guardian: Motor Drivers and Encoders

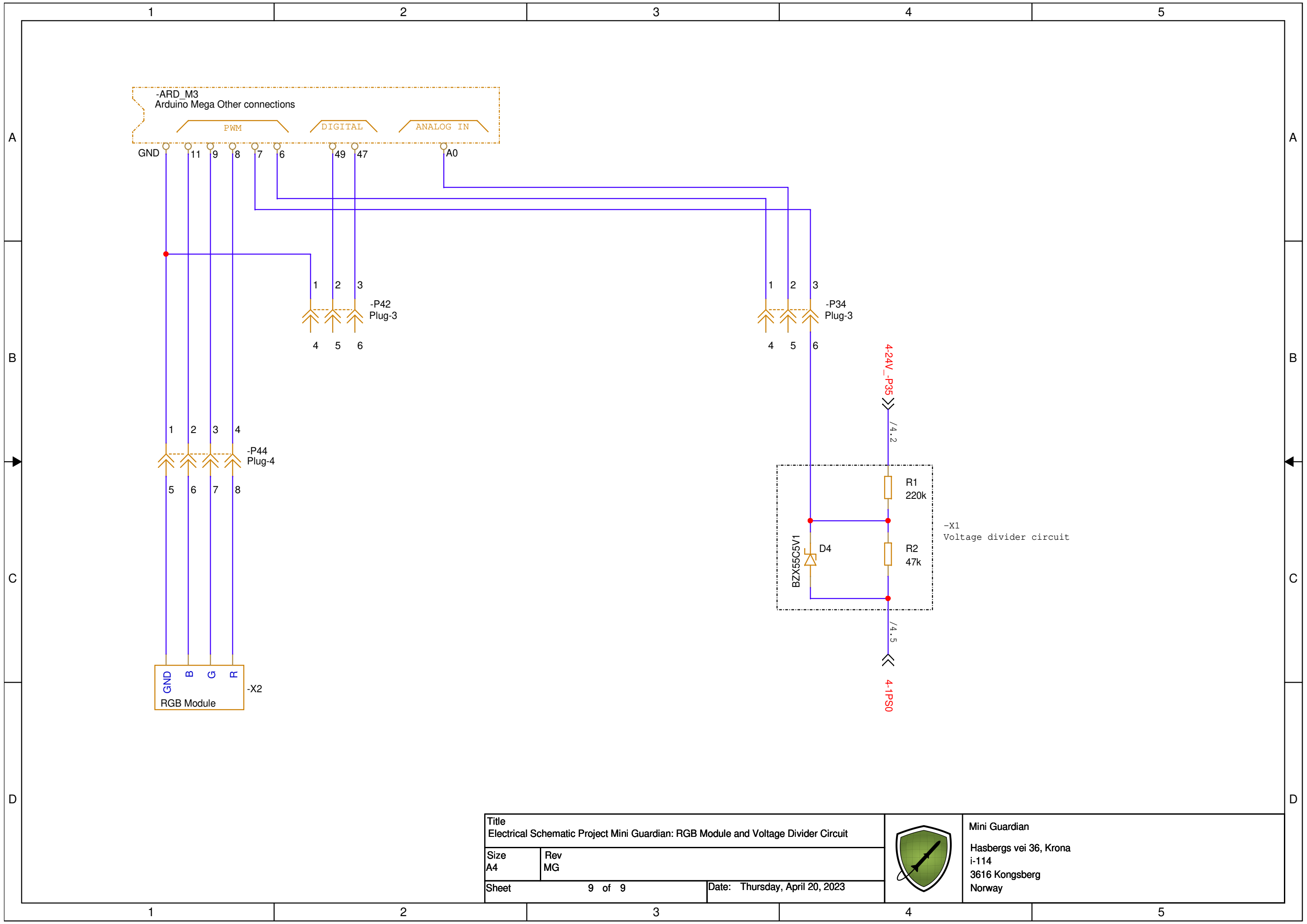
Size	Rev
A4	MG

Sheet 7 of 9 Date: Tuesday, April 18, 2023

Mini Guardian

Hasbergs vei 36, Krona
i-114
3616 Kongsberg
Norway


1
2
3
4
5
A
B
C
D
4-1PS24
/4.5
-S11
Position_switch_no/nc
Common Limit Switch
1
3
2
-S12
Position_switch_no/nc
Reference Position
-S16
Position_switch_no/nc
Overtravel Azimuth + direction
-S17
Position_switch_no/nc
Overtravel Azimuth - direction
R5
220k
R7
220k
R9
220k
R6
47k
R8
47k
R10
47k
D1
D2
D3
BZX55C5V1
-X1
Voltage Divider circuit
4-1PS0
/4.5
18
19
39
COMMUNICATION
DIGITAL
-ARD_M1
Arduino Mega Limit Switches and Reference
/7.2
8-Driver24V
Title
Electrical Schematic Project Mini Guardian: Limit Switches and Zero-point Reference
Size
A4
Rev
MG
Sheet
8 of 9
Date: Wednesday, April 19, 2023
Mini Guardian
Hasbergs vei 36, Krona
i-114
3616 Kongsberg
Norway



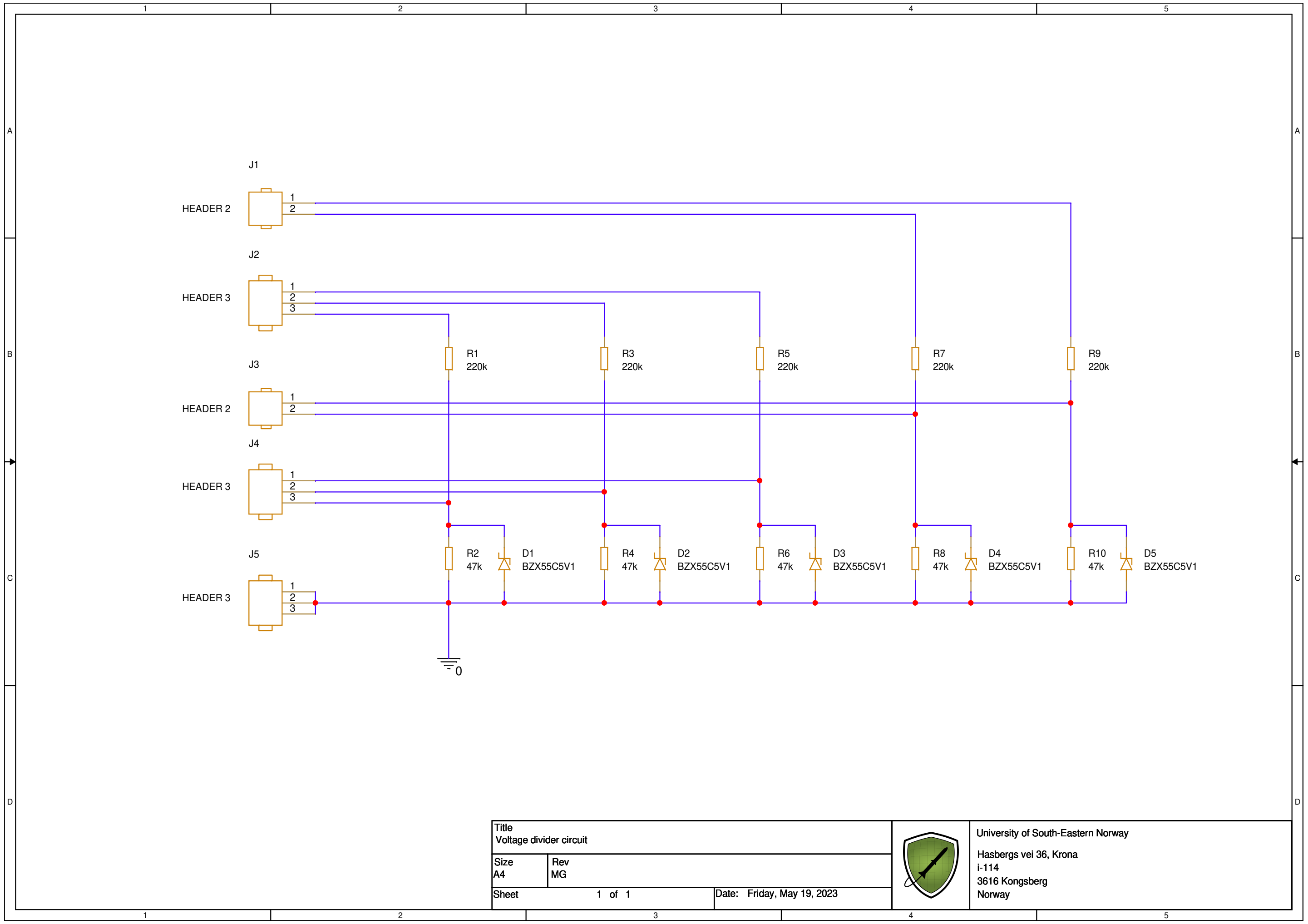
Title
Electrical Schematic Project Mini Guardian: RGB Module and Voltage Divider Circuit

Size A4
Rev MG
Sheet 9 of 9
Date: Thursday, April 20, 2023

Mini Guardian

Hasbergs vei 36, Krona
i-114
3616 Kongsberg
Norway

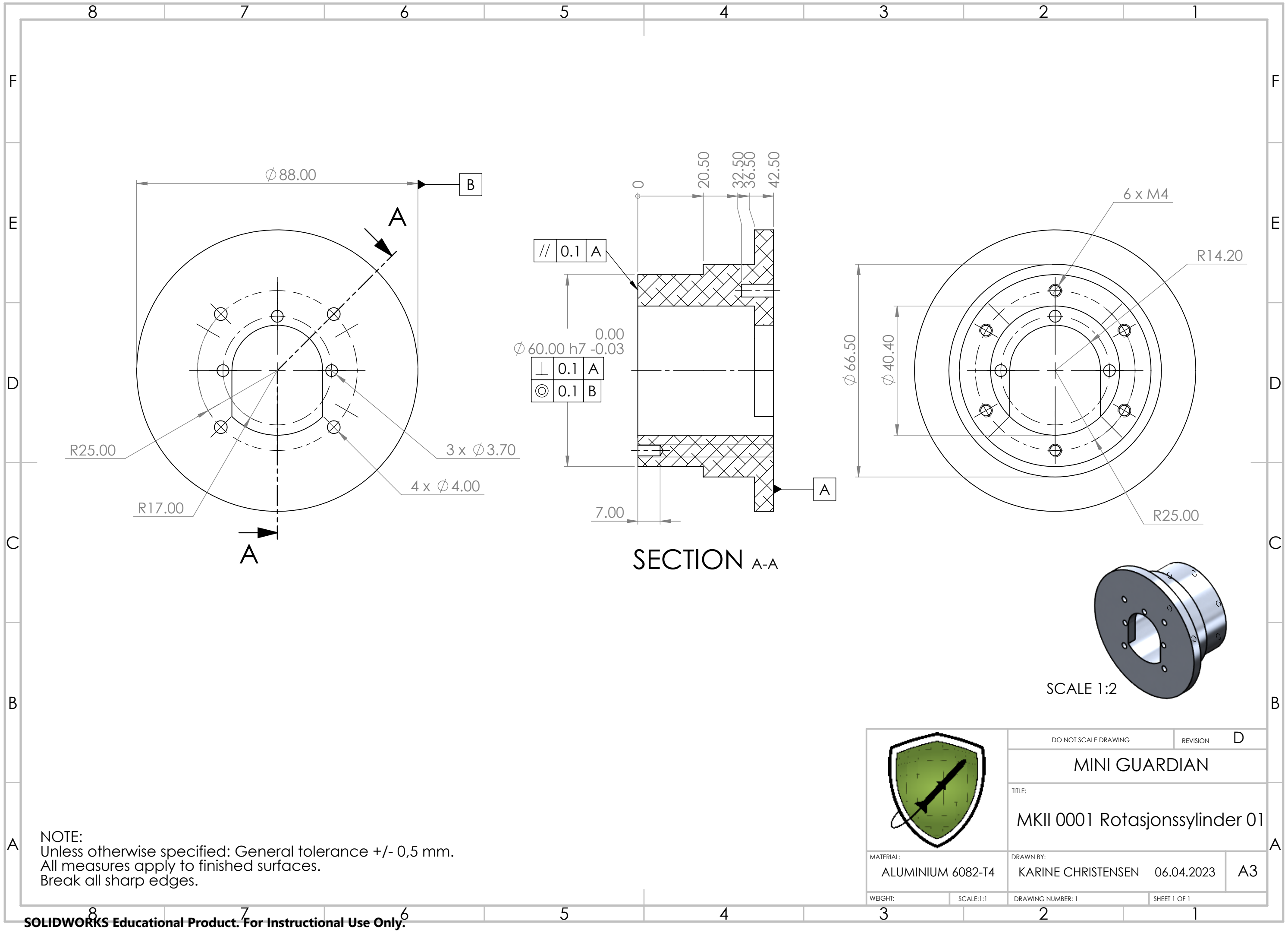
B Electrical schematic for voltage divider



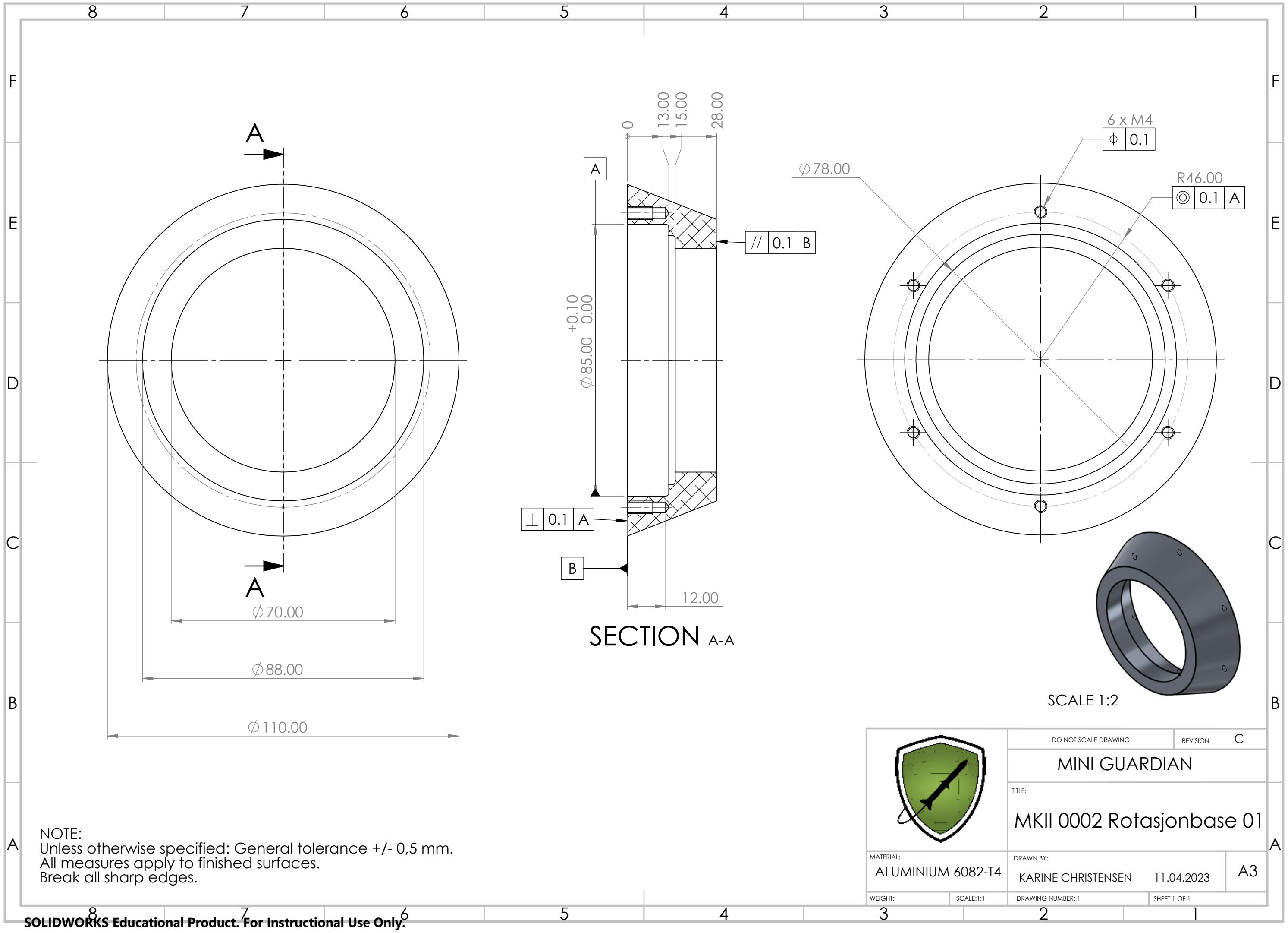
Appendix E

Mechanical drawings

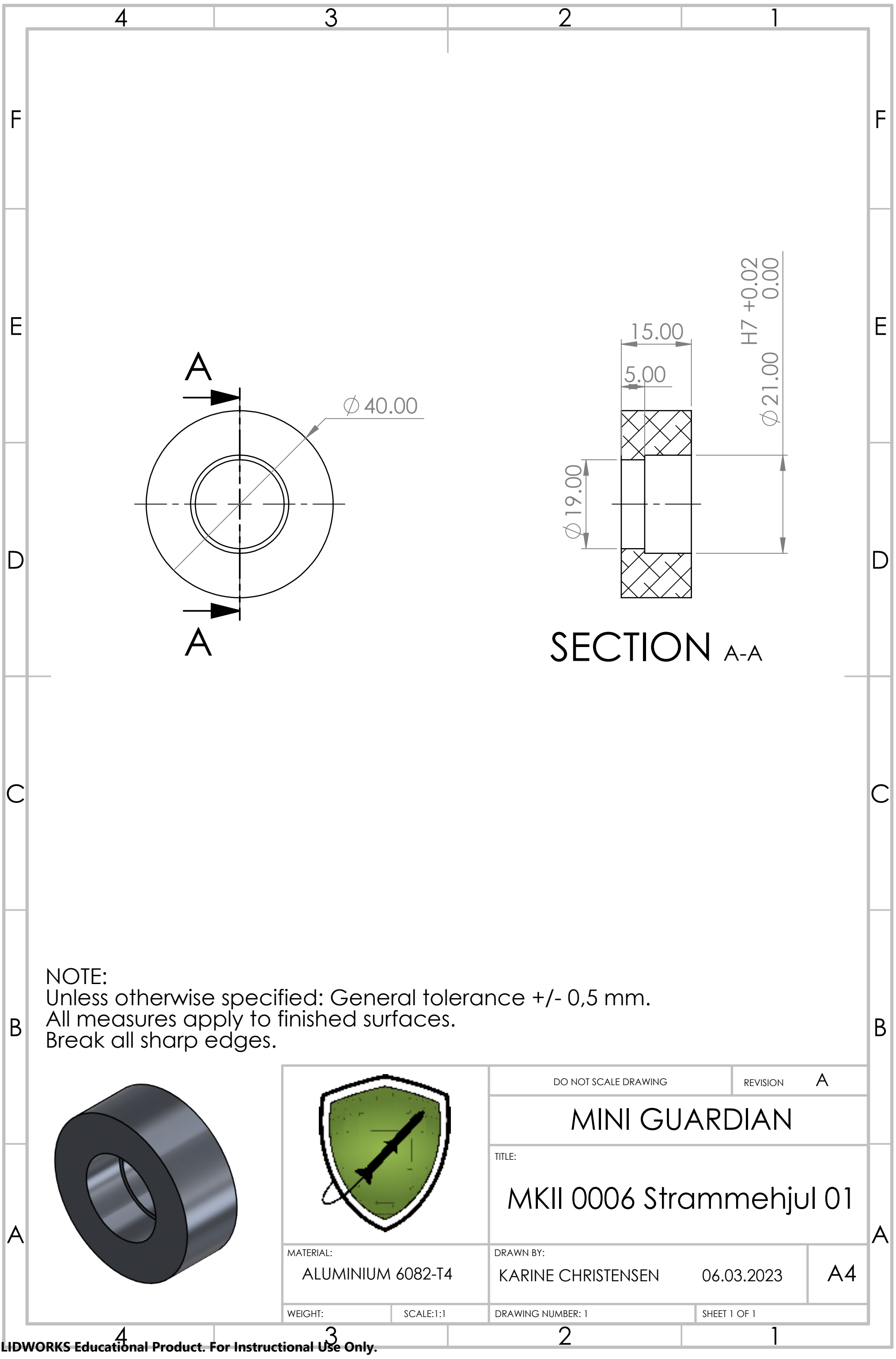
A Rotation Cylinder Production Drawing



B Rotation Base Production Drawing



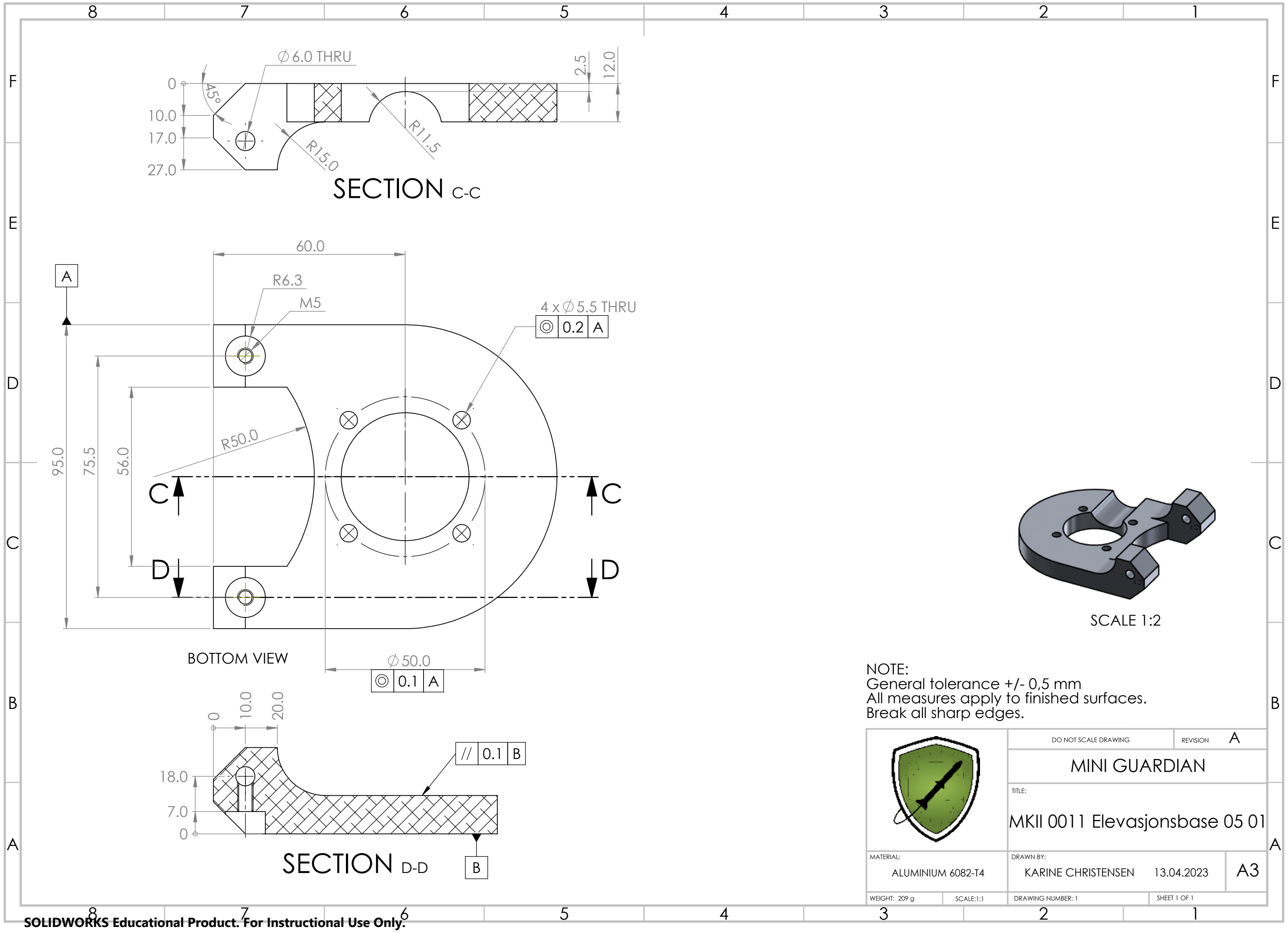
C Tension Wheel Production Drawing



NOTE:
Unless otherwise specified: General tolerance +/- 0,5 mm.
All measures apply to finished surfaces.
Break all sharp edges.

DO NOT SCALE DRAWING	REVISION	A
MINI GUARDIAN		
TITLE: MKII 0006 Strammehjul 01		
MATERIAL: ALUMINIUM 6082-T4	DRAWN BY: KARINE CHRISTENSEN 06.03.2023	A4
WEIGHT:	SCALE:1:1	DRAWING NUMBER: 1	SHEET 1 OF 1

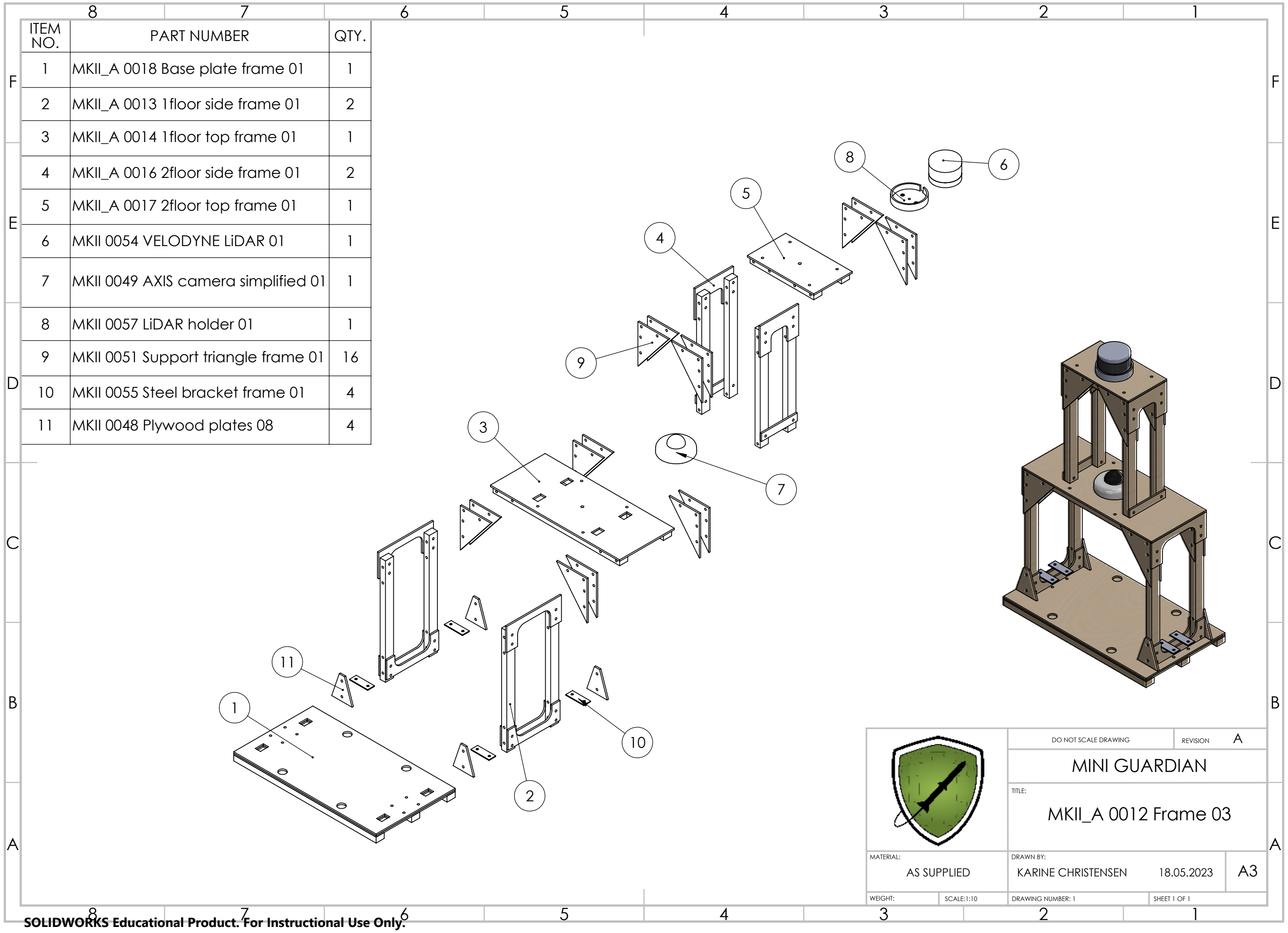
D Elevation Base Production Drawing



NOTE:
General tolerance +/- 0,5 mm
All measures apply to finished surfaces.
Break all sharp edges.

	DO NOT SCALE DRAWING	REVISION A
	MINI GUARDIAN	
	TITLE: MKII 0011 Elevasjonsbase 05 01	
MATERIAL: ALUMINIUM 6082-T4	DRAWN BY: KARINE CHRISTENSEN 13.04.2023	A3
WEIGHT: 209 g SCALE:1:1	DRAWING NUMBER: 1	SHEET 1 OF 1

E Exploded view of Sensor enclosure



ITEM NO.	PART NUMBER	QTY.
1	MKII_A 0018 Base plate frame 01	1
2	MKII_A 0013 1floor side frame 01	2
3	MKII_A 0014 1floor top frame 01	1
4	MKII_A 0016 2floor side frame 01	2
5	MKII_A 0017 2floor top frame 01	1
6	MKII 0054 VELODYNE LiDAR 01	1
7	MKII 0049 AXIS camera simplified 01	1
8	MKII 0057 LiDAR holder 01	1
9	MKII 0051 Support triangle frame 01	16
10	MKII 0055 Steel bracket frame 01	4
11	MKII 0048 Plywood plates 08	4

DO NOT SCALE DRAWING

REVISION A

MINI GUARDIAN

TITLE: MKII_A 0012 Frame 03

MATERIAL: AS SUPPLIED

DRAWN BY: KARINE CHRISTENSEN 18.05.2023

A3

WEIGHT:

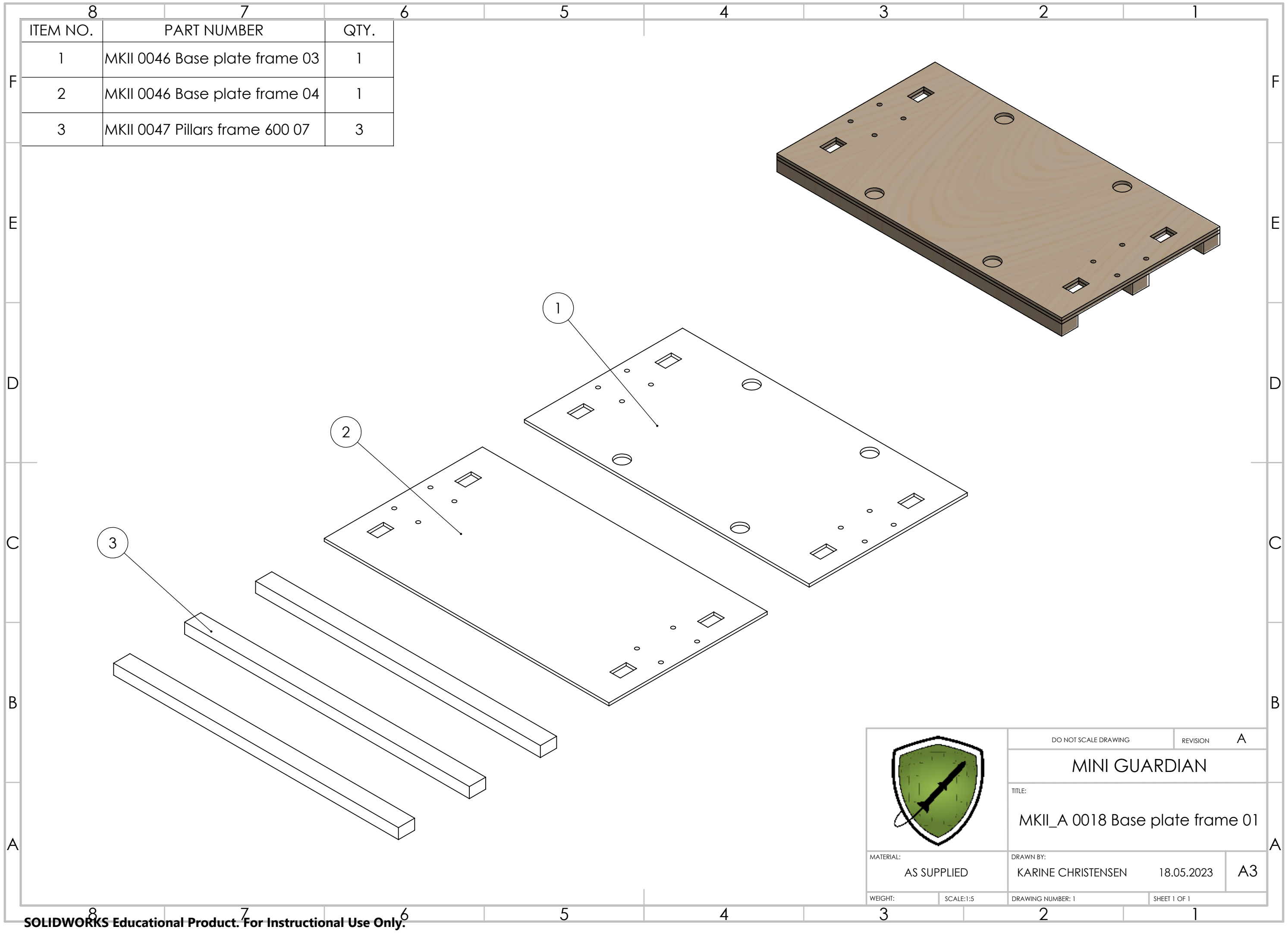
SCALE:1:10

DRAWING NUMBER: 1

SHEET 1 OF 1

SOLIDWORKS Educational Product. For Instructional Use Only.

F Exploded view of Sensor enclosure module 01

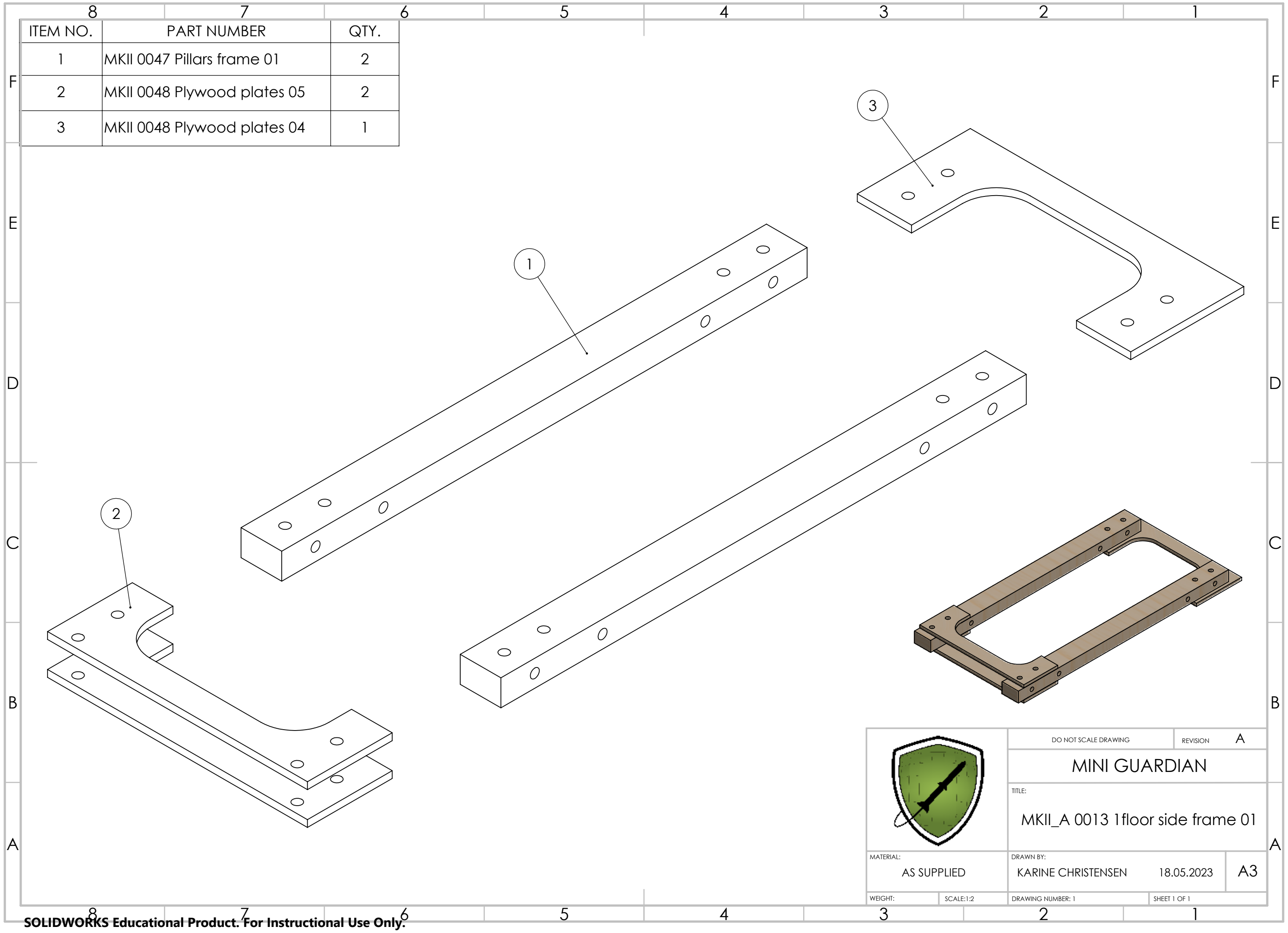


ITEM NO.	PART NUMBER	QTY.
1	MKII 0046 Base plate frame 03	1
2	MKII 0046 Base plate frame 04	1
3	MKII 0047 Pillars frame 600 07	3

DO NOT SCALE DRAWING		REVISION	A
MINI GUARDIAN			
TITLE: MKII_A 0018 Base plate frame 01			
MATERIAL: AS SUPPLIED	DRAWN BY: KARINE CHRISTENSEN	18.05.2023	A3
WEIGHT:	SCALE:1:5	DRAWING NUMBER: 1	SHEET 1 OF 1

SOLIDWORKS Educational Product. For Instructional Use Only.

G Exploded view of Sensor enclosure module 02

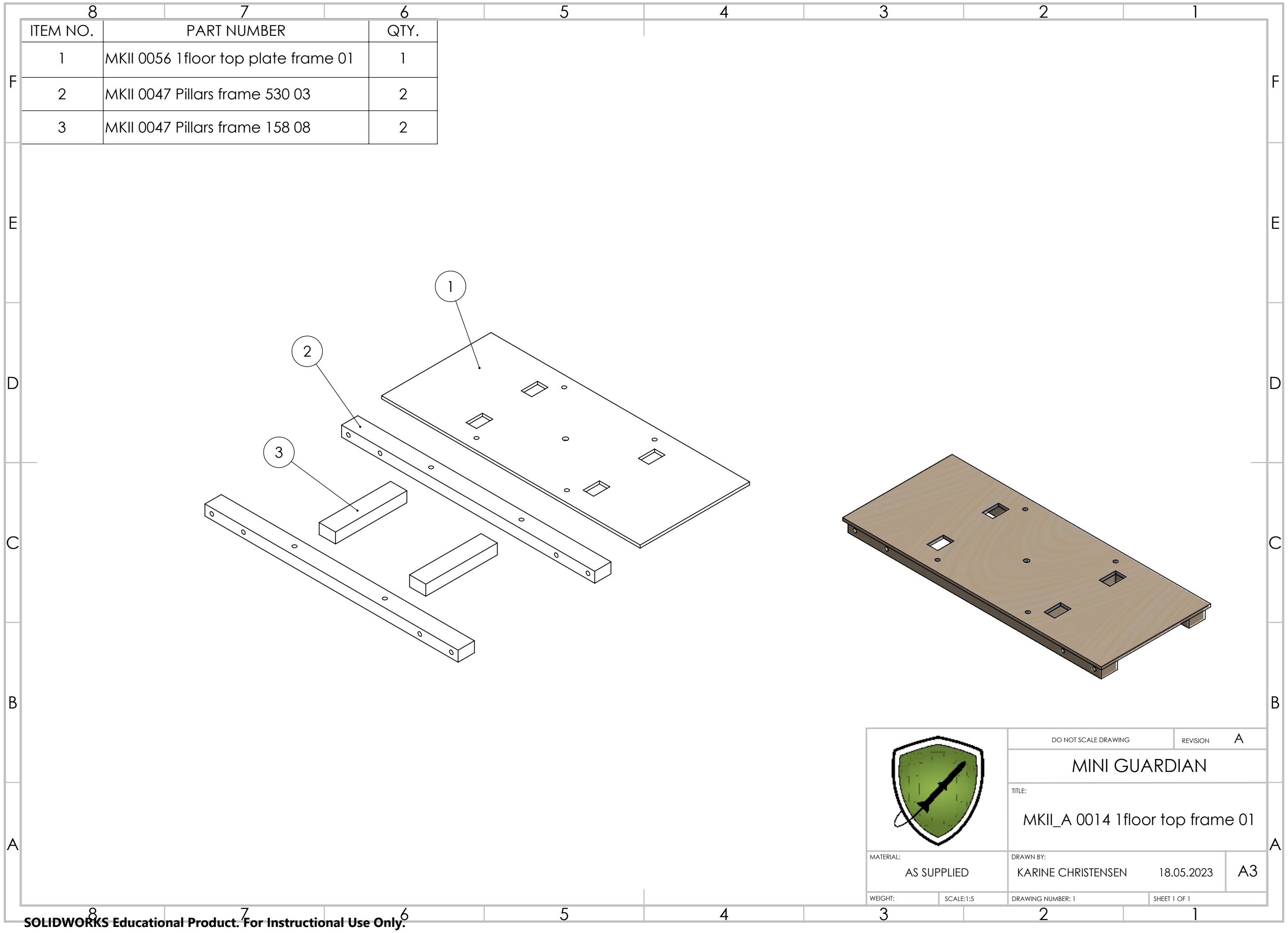


ITEM NO.	PART NUMBER	QTY.
1	MKII 0047 Pillars frame 01	2
2	MKII 0048 Plywood plates 05	2
3	MKII 0048 Plywood plates 04	1

	DO NOT SCALE DRAWING	REVISION	A
	MINI GUARDIAN		
	TITLE: MKII_A 0013 1floor side frame 01		
MATERIAL: AS SUPPLIED	DRAWN BY: KARINE CHRISTENSEN	18.05.2023	A3
WEIGHT: SCALE:1:2	DRAWING NUMBER: 1	SHEET 1 OF 1	

SOLIDWORKS Educational Product. For Instructional Use Only.

H Exploded view of Sensor enclosure module 03

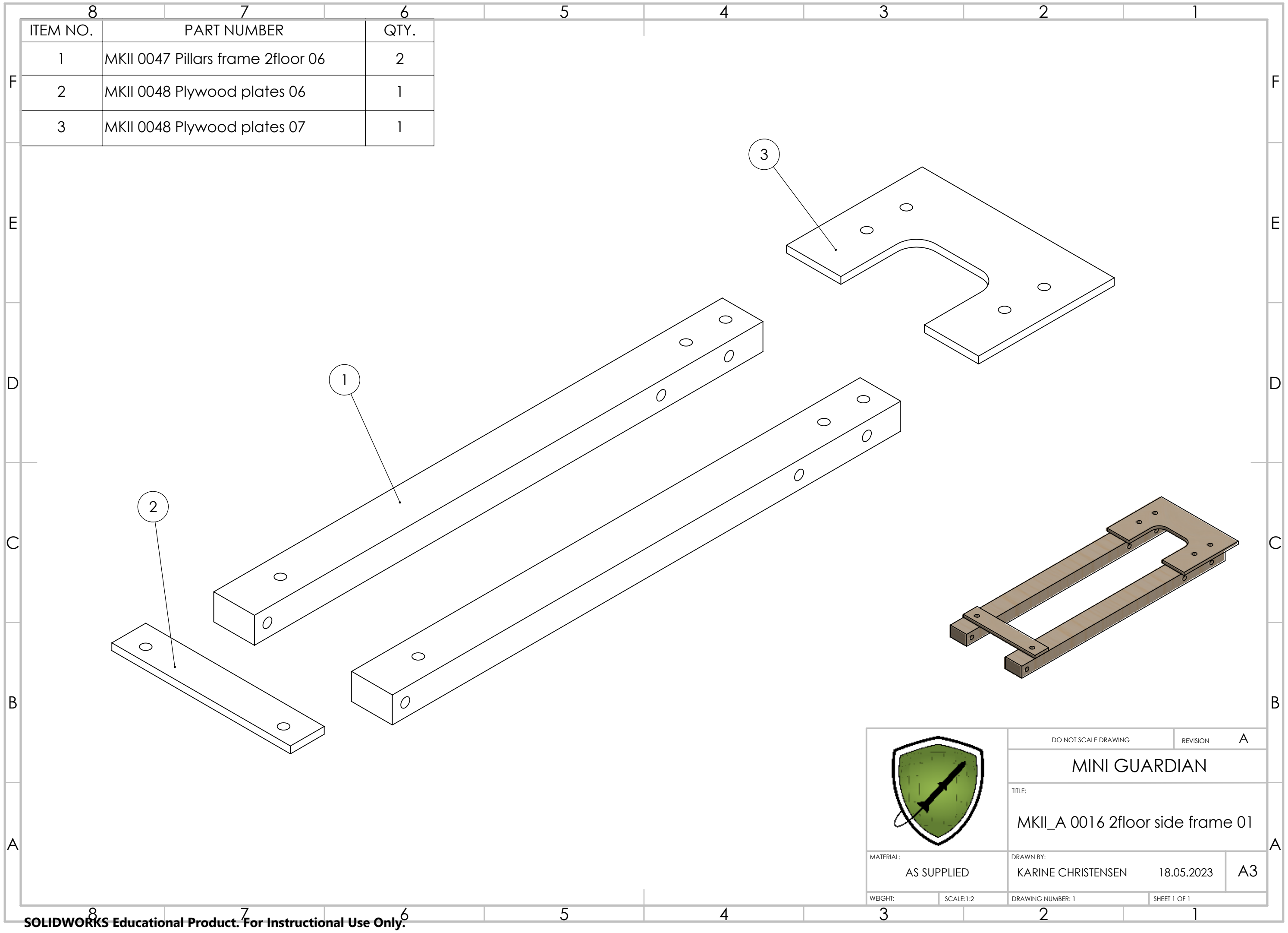


ITEM NO.	PART NUMBER	QTY.
1	MKII 0056 1floor top plate frame 01	1
2	MKII 0047 Pillars frame 530 03	2
3	MKII 0047 Pillars frame 158 08	2

DO NOT SCALE DRAWING		REVISION	A
MINI GUARDIAN			
TITLE: MKII_A 0014 1floor top frame 01			
MATERIAL: AS SUPPLIED	DRAWN BY: KARINE CHRISTENSEN	18.05.2023	A3
WEIGHT:	SCALE:1:5	DRAWING NUMBER: 1	SHEET 1 OF 1

SOLIDWORKS Educational Product. For Instructional Use Only.

I Exploded view of Sensor enclosure module 04

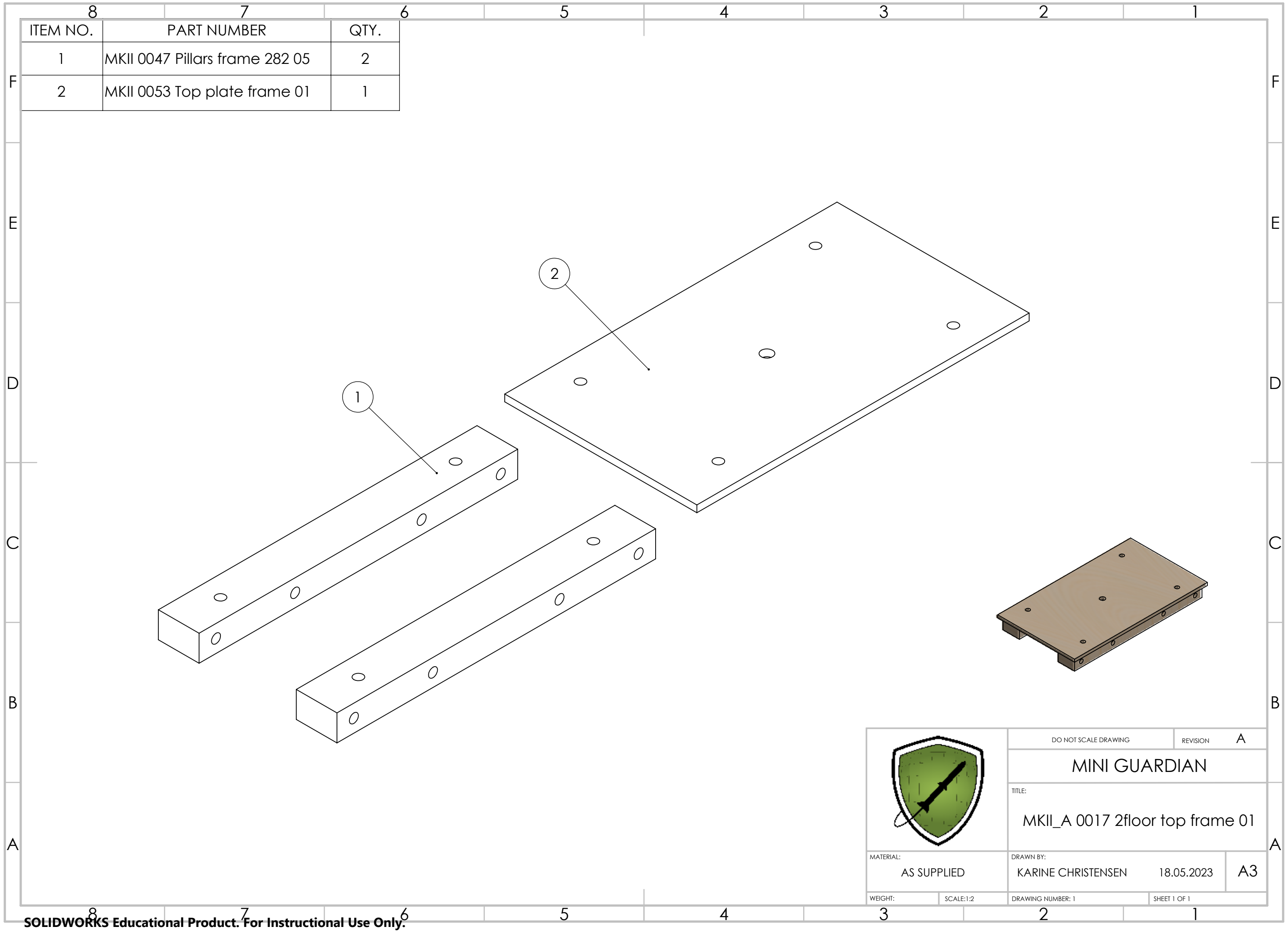


ITEM NO.	PART NUMBER	QTY.
1	MKII 0047 Pillars frame 2floor 06	2
2	MKII 0048 Plywood plates 06	1
3	MKII 0048 Plywood plates 07	1

	DO NOT SCALE DRAWING	REVISION	A
	MINI GUARDIAN		
	TITLE: MKII_A 0016 2floor side frame 01		
MATERIAL: AS SUPPLIED	DRAWN BY: KARINE CHRISTENSEN 18.05.2023		A3
WEIGHT: SCALE:1:2	DRAWING NUMBER: 1	SHEET 1 OF 1	

SOLIDWORKS Educational Product. For Instructional Use Only.

J Exploded view of Sensor enclosure module 05



ITEM NO.	PART NUMBER	QTY.
1	MKII 0047 Pillars frame 282 05	2
2	MKII 0053 Top plate frame 01	1

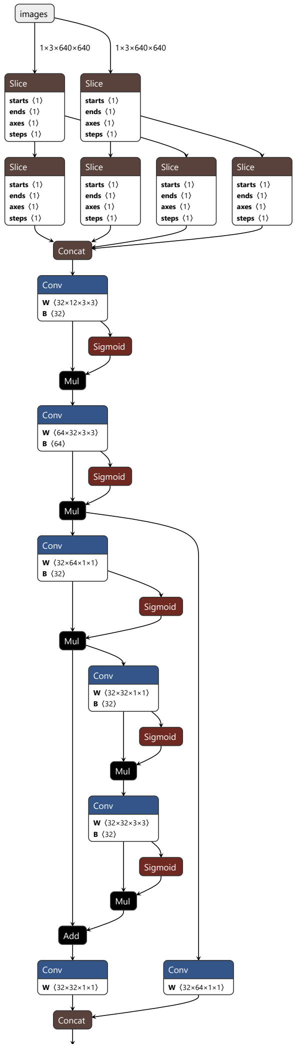
	DO NOT SCALE DRAWING	REVISION A
	MINI GUARDIAN	
	TITLE: MKII_A 0017 2floor top frame 01	
MATERIAL: AS SUPPLIED	DRAWN BY: KARINE CHRISTENSEN 18.05.2023	A3
WEIGHT: SCALE:1:2	DRAWING NUMBER: 1	SHEET 1 OF 1

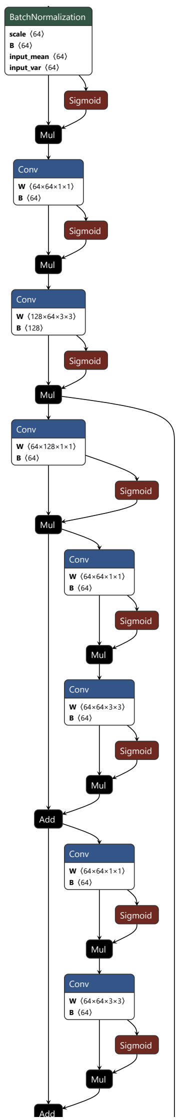
SOLIDWORKS Educational Product. For Instructional Use Only.

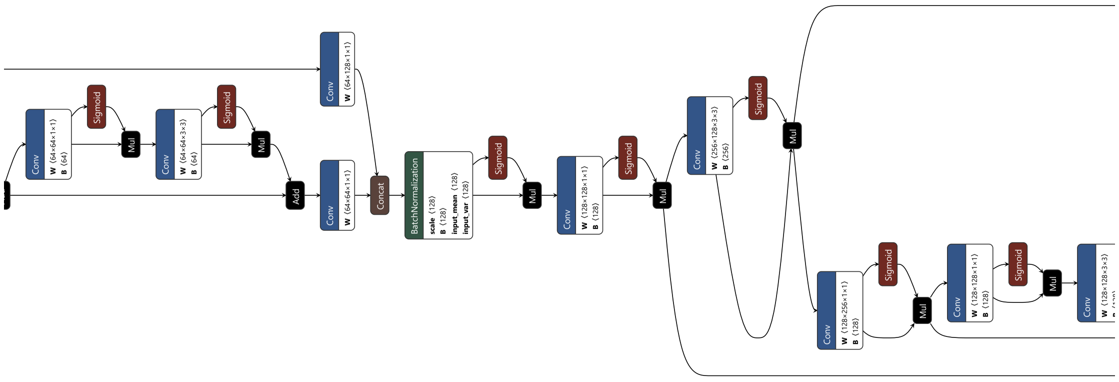
Appendix F

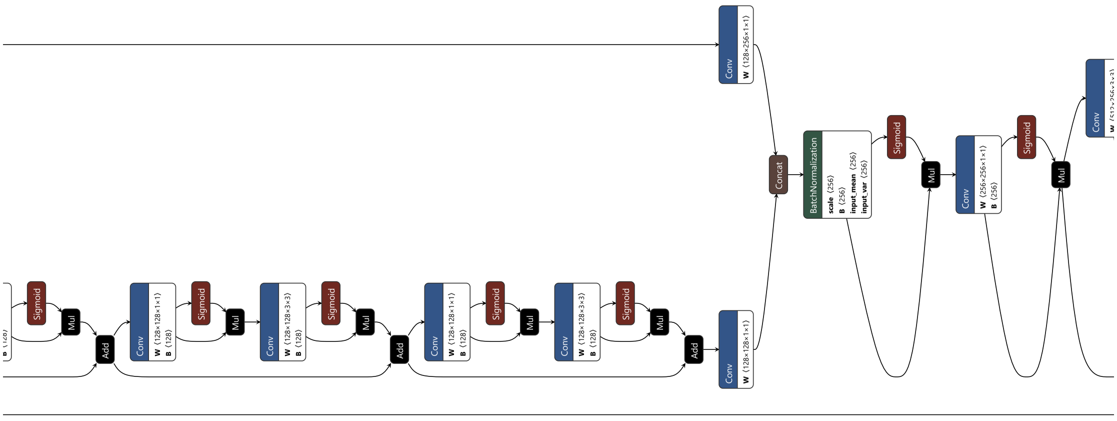
General

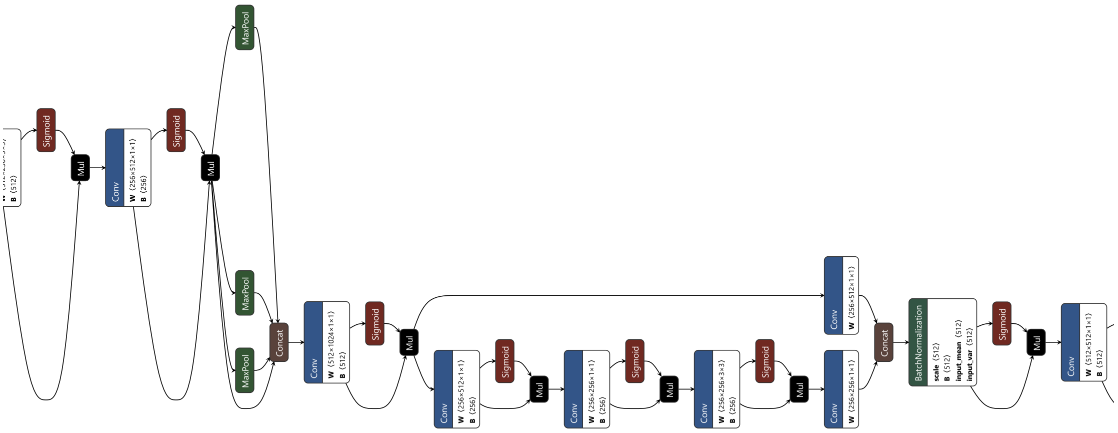
A Object detection model

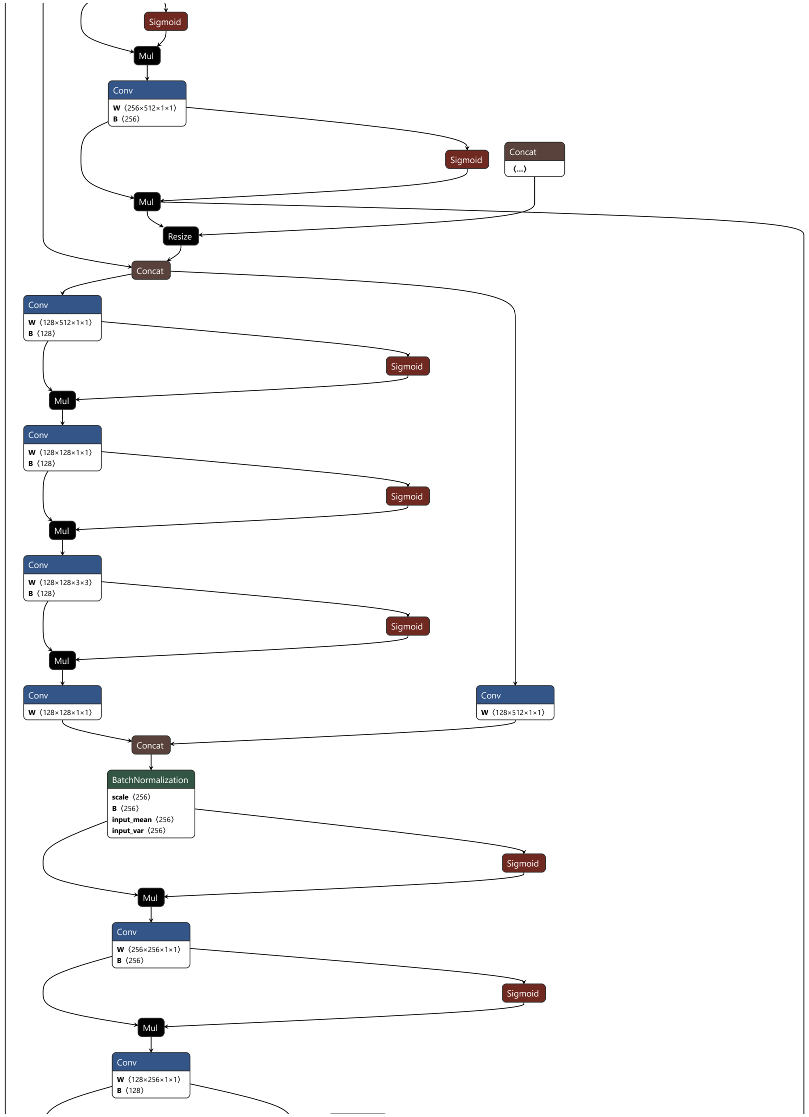


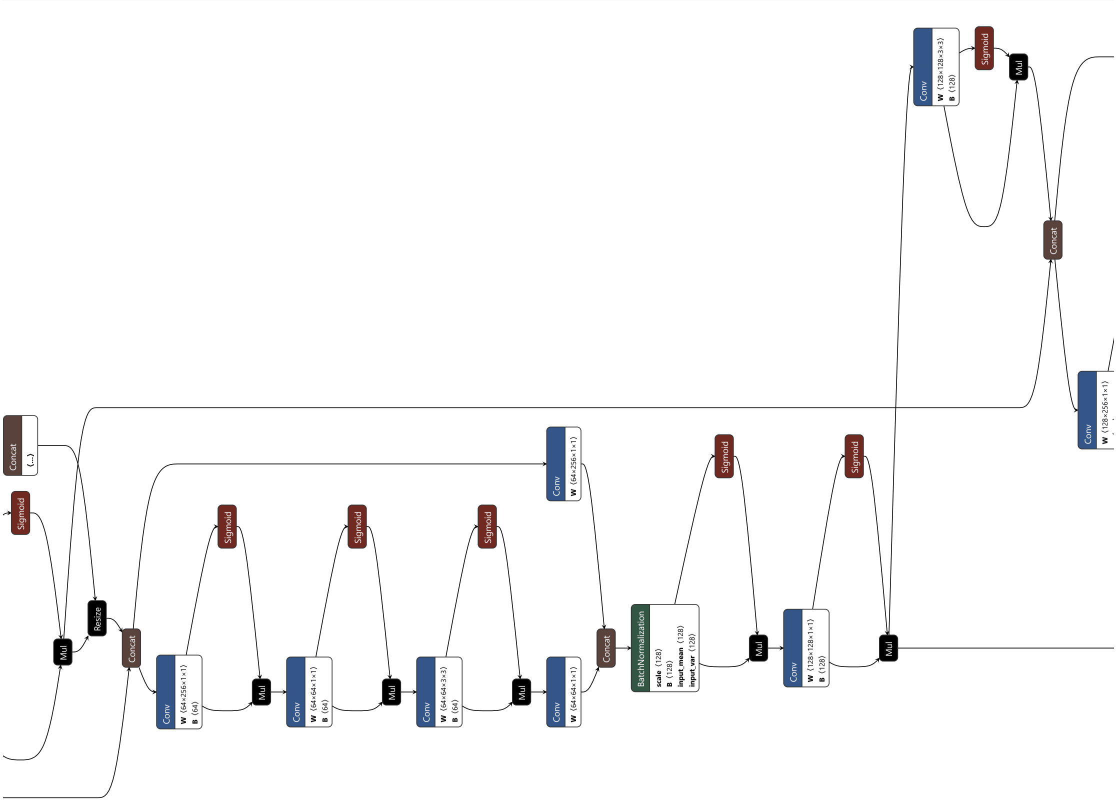

BatchNormalization
scale ⟨64⟩
B ⟨64⟩
input_mean ⟨64⟩
input_var ⟨64⟩
Sigmoid
Mul
Conv
W ⟨64×64×1×1⟩
B ⟨64⟩
Sigmoid
Mul
Conv
W ⟨128×64×3×3⟩
B ⟨128⟩
Sigmoid
Mul
Conv
W ⟨64×128×1×1⟩
B ⟨64⟩
Sigmoid
Mul
Conv
W ⟨64×64×1×1⟩
B ⟨64⟩
Sigmoid
Mul
Conv
W ⟨64×64×3×3⟩
B ⟨64⟩
Sigmoid
Mul
Add
Conv
W ⟨64×64×1×1⟩
B ⟨64⟩
Sigmoid
Mul
Conv
W ⟨64×64×3×3⟩
B ⟨64⟩
Sigmoid
Mul
Add

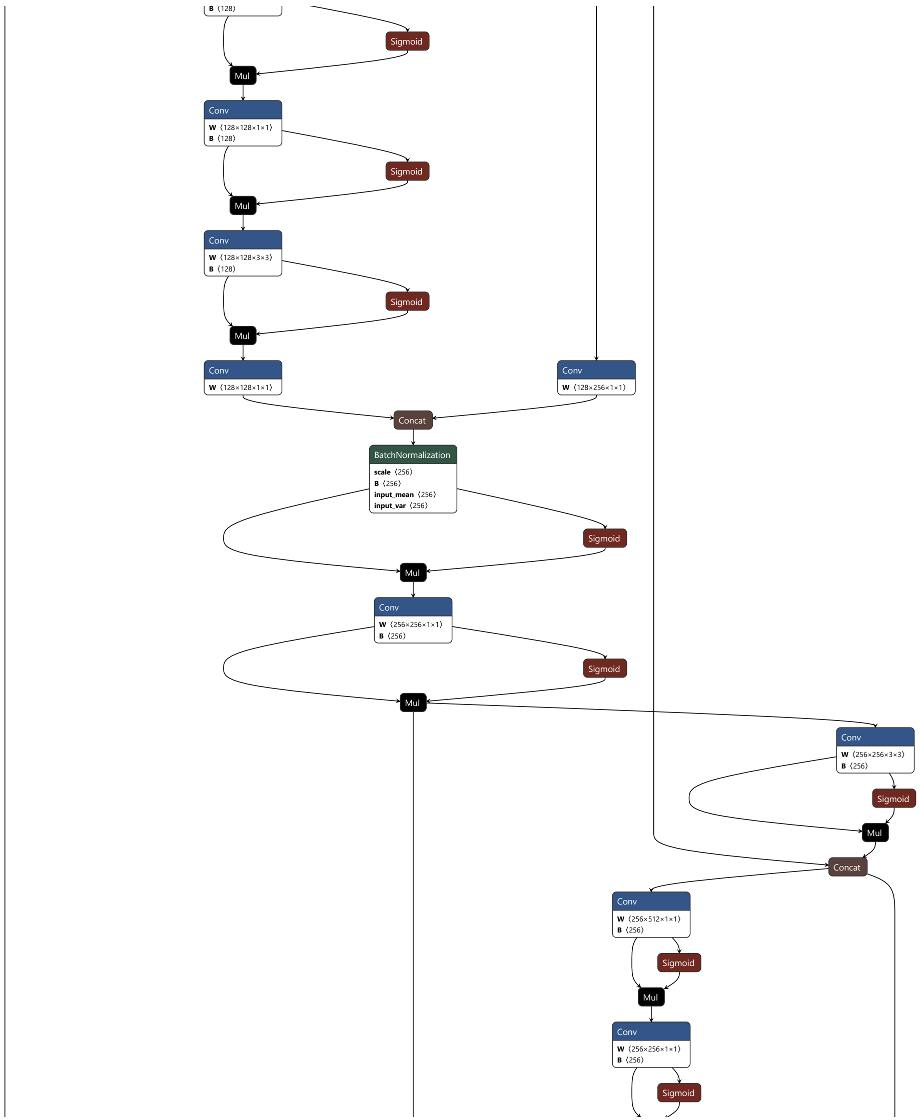

Conv
W ⟨64×64×1×1⟩
B ⟨64⟩
Sigmoid
Mul
Conv
W ⟨64×64×3×3⟩
B ⟨64⟩
Sigmoid
Mul
Add
Conv
W ⟨64×64×1×1⟩
Conv
W ⟨64×128×1×1⟩
Concat
BatchNormalization
scale ⟨128⟩
B ⟨128⟩
input_mean ⟨128⟩
input_var ⟨128⟩
Sigmoid
Mul
Conv
W ⟨128×128×1×1⟩
B ⟨128⟩
Sigmoid
Mul
Conv
W ⟨256×128×3×3⟩
B ⟨256⟩
Sigmoid
Mul
Conv
W ⟨128×256×1×1⟩
B ⟨128⟩
Sigmoid
Mul
Conv
W ⟨128×128×1×1⟩
B ⟨128⟩
Sigmoid
Mul
Conv
W ⟨128×128×3×3⟩

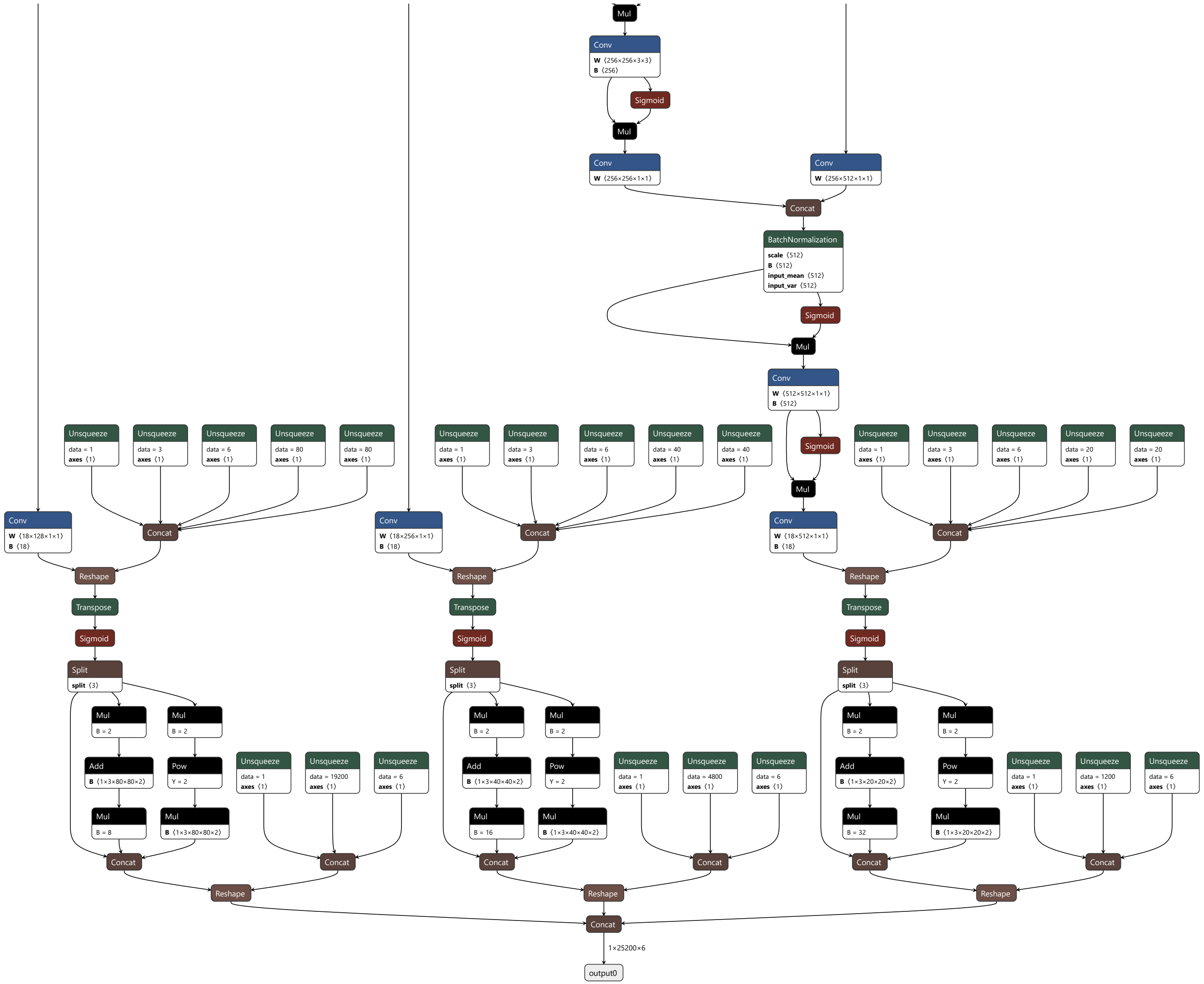

B ⟨128⟩
Sigmoid
Mul
Add
Conv
W ⟨128×128×1×1⟩
B ⟨128⟩
Sigmoid
Mul
Conv
W ⟨128×128×3×3⟩
B ⟨128⟩
Sigmoid
Mul
Add
Conv
W ⟨128×128×1×1⟩
B ⟨128⟩
Sigmoid
Mul
Conv
W ⟨128×128×3×3⟩
B ⟨128⟩
Sigmoid
Mul
Add
Conv
W ⟨128×128×1×1⟩
Conv
W ⟨128×256×1×1⟩
Concat
BatchNormalization
scale ⟨256⟩
B ⟨256⟩
input_mean ⟨256⟩
input_var ⟨256⟩
Sigmoid
Mul
Conv
W ⟨256×256×1×1⟩
B ⟨256⟩
Sigmoid
Mul
Conv
W ⟨512×256×3×3⟩


B ⟨512⟩
Sigmoid
Mul
Conv
W ⟨256×512×1×1⟩
B ⟨256⟩
Sigmoid
Mul
MaxPool
MaxPool
MaxPool
Concat
Conv
W ⟨512×1024×1×1⟩
B ⟨512⟩
Sigmoid
Mul
Conv
W ⟨256×512×1×1⟩
B ⟨256⟩
Sigmoid
Mul
Conv
W ⟨256×256×1×1⟩
B ⟨256⟩
Sigmoid
Mul
Conv
W ⟨256×256×3×3⟩
B ⟨256⟩
Sigmoid
Mul
Conv
W ⟨256×256×1×1⟩
Conv
W ⟨256×512×1×1⟩
Concat
BatchNormalization
scale ⟨512⟩
B ⟨512⟩
input_mean ⟨512⟩
input_var ⟨512⟩
Sigmoid
Mul
Conv
W ⟨512×512×1×1⟩
B ⟨512⟩


Sigmoid
Mul
Conv
W ⟨256×512×1×1⟩
B ⟨256⟩
Sigmoid
Concat
⟨...⟩
Mul
Resize
Concat
Conv
W ⟨128×512×1×1⟩
B ⟨128⟩
Sigmoid
Mul
Conv
W ⟨128×128×1×1⟩
B ⟨128⟩
Sigmoid
Mul
Conv
W ⟨128×128×3×3⟩
B ⟨128⟩
Sigmoid
Mul
Conv
W ⟨128×128×1×1⟩
Conv
W ⟨128×512×1×1⟩
Concat
BatchNormalization
scale ⟨256⟩
B ⟨256⟩
input_mean ⟨256⟩
input_var ⟨256⟩
Sigmoid
Mul
Conv
W ⟨256×256×1×1⟩
B ⟨256⟩
Sigmoid
Mul
Conv
W ⟨128×256×1×1⟩
B ⟨128⟩


Sigmoid
Concat
(...)
Mul
Resize
Concat
Conv
W ⟨64×256×1×1⟩
B ⟨64⟩
Sigmoid
Mul
Conv
W ⟨64×64×1×1⟩
B ⟨64⟩
Sigmoid
Mul
Conv
W ⟨64×64×3×3⟩
B ⟨64⟩
Sigmoid
Mul
Conv
W ⟨64×64×1×1⟩
Conv
W ⟨64×256×1×1⟩
Concat
BatchNormalization
scale ⟨128⟩
B ⟨128⟩
input_mean ⟨128⟩
input_var ⟨128⟩
Sigmoid
Mul
Conv
W ⟨128×128×1×1⟩
B ⟨128⟩
Sigmoid
Mul
Conv
W ⟨128×128×3×3⟩
B ⟨128⟩
Sigmoid
Mul
Concat
Conv
W ⟨128×256×1×1⟩


B 〈128〉
Sigmoid
Mul
Conv
W 〈128×128×1×1〉
B 〈128〉
Sigmoid
Mul
Conv
W 〈128×128×3×3〉
B 〈128〉
Sigmoid
Mul
Conv
W 〈128×128×1×1〉
Conv
W 〈128×256×1×1〉
Concat
BatchNormalization
scale 〈256〉
B 〈256〉
input_mean 〈256〉
input_var 〈256〉
Sigmoid
Mul
Conv
W 〈256×256×1×1〉
B 〈256〉
Sigmoid
Mul
Conv
W 〈256×256×3×3〉
B 〈256〉
Sigmoid
Mul
Concat
Conv
W 〈256×512×1×1〉
B 〈256〉
Sigmoid
Mul
Conv
W 〈256×256×1×1〉
B 〈256〉
Sigmoid


Mul
Conv
W ⟨256×256×3×3⟩
B ⟨256⟩
Sigmoid
Mul
Conv
W ⟨256×256×1×1⟩
Conv
W ⟨256×512×1×1⟩
Concat
BatchNormalization
scale ⟨512⟩
B ⟨512⟩
input_mean ⟨512⟩
input_var ⟨512⟩
Sigmoid
Mul
Conv
W ⟨512×512×1×1⟩
B ⟨512⟩
Sigmoid
Mul
Unsqueeze
data = 1
axes ⟨1⟩
Unsqueeze
data = 3
axes ⟨1⟩
Unsqueeze
data = 6
axes ⟨1⟩
Unsqueeze
data = 80
axes ⟨1⟩
Unsqueeze
data = 80
axes ⟨1⟩
Conv
W ⟨18×128×1×1⟩
B ⟨18⟩
Concat
Reshape
Transpose
Sigmoid
Split
split ⟨3⟩
Mul
B = 2
Mul
B = 2
Add
B ⟨1×3×80×80×2⟩
Pow
Y = 2
Mul
B = 8
Mul
B ⟨1×3×80×80×2⟩
Unsqueeze
data = 1
axes ⟨1⟩
Unsqueeze
data = 19200
axes ⟨1⟩
Unsqueeze
data = 6
axes ⟨1⟩
Concat
Concat
Reshape
Unsqueeze
data = 1
axes ⟨1⟩
Unsqueeze
data = 3
axes ⟨1⟩
Unsqueeze
data = 6
axes ⟨1⟩
Unsqueeze
data = 40
axes ⟨1⟩
Unsqueeze
data = 40
axes ⟨1⟩
Conv
W ⟨18×256×1×1⟩
B ⟨18⟩
Concat
Reshape
Transpose
Sigmoid
Split
split ⟨3⟩
Mul
B = 2
Mul
B = 2
Add
B ⟨1×3×40×40×2⟩
Pow
Y = 2
Mul
B = 16
Mul
B ⟨1×3×40×40×2⟩
Unsqueeze
data = 1
axes ⟨1⟩
Unsqueeze
data = 4800
axes ⟨1⟩
Unsqueeze
data = 6
axes ⟨1⟩
Concat
Concat
Reshape
Unsqueeze
data = 1
axes ⟨1⟩
Unsqueeze
data = 3
axes ⟨1⟩
Unsqueeze
data = 6
axes ⟨1⟩
Unsqueeze
data = 20
axes ⟨1⟩
Unsqueeze
data = 20
axes ⟨1⟩
Conv
W ⟨18×512×1×1⟩
B ⟨18⟩
Concat
Reshape
Transpose
Sigmoid
Split
split ⟨3⟩
Mul
B = 2
Mul
B = 2
Add
B ⟨1×3×20×20×2⟩
Pow
Y = 2
Mul
B = 32
Mul
B ⟨1×3×20×20×2⟩
Unsqueeze
data = 1
axes ⟨1⟩
Unsqueeze
data = 1200
axes ⟨1⟩
Unsqueeze
data = 6
axes ⟨1⟩
Concat
Concat
Reshape
Concat
1×25200×6
output0

B Script for plotting object detection data

This is the Matlab script to plot the resulting data given by the object detection training software.

```
% Author Marius Heistad
% Date Created: 18.05.2023
% The purpose if this script is to plot the data that has been recorded
% when the object detection model has been trained.

% Clear window:
clc;
clear all;

% Import xaxis data to be plotted
xaxis = readtable('results.xlsx', 'Range', 'A2:A523');
xaxis = table2array(xaxis);

% Import Precision data to be plotted
metricsPrecision = readtable('results.xlsx', 'Range', 'B2:B523');
metricsPrecision = table2array(metricsPrecision);

% Import Recal data to be plotted
metricsRecall = readtable('results.xlsx', 'Range', 'C2:C523');
metricsRecall = table2array(metricsRecall);

% Import mAP data to be plotted
metrics_m_AP = readtable('results.xlsx', 'Range', 'D2:D523');
metrics_m_AP = table2array(metrics_m_AP);

% Color to plot with
green = [0.4660 0.6740 0.1880]

% Plot precision;
figure(1)
plot(xaxis, metricsPrecision, 'Color', [0.4660 0.6740 0.1880], 'LineWidt', 1);
xlim([-10 350]);
ylim([0 1.1]);
title('Precision');
xlabel('Epoch');
ylabel('Normalised');
grid on;

% Plot recall;
figure(2)
```

```matlab
plot(xaxis, metricsRecall, 'Color', [0.4660 0.6740 0.1880], 'LineWidt', 1);
xlim([-10 350]);
ylim([0 1.1]);
title('Recall');
xlabel('Epoch');
ylabel('Normalised');
grid on;

% Plot m_AP;
figure(3)
plot(xaxis, metrics_m_AP, 'Color', [0.4660 0.6740 0.1880], 'LineWidt', 1);
xlim([-10 350]);
ylim([0 1.1]);
title('mean Average Precision (mAP)');
xlabel('Epoch');
ylabel('Normalised');
grid on;

% Plot recall;
figure(4)
plot([0; metricsRecall],[1; metricsPrecision], 'Color', [0.4660 0.6740 0.1880]);
title('Recall');
xlabel('Epoch');
ylabel('Normalised');
grid on;
```

C Ethernet and stepper motor values

This diagram illustrates the values utilized for both the stepper motor and Ethernet communication. It provides clarity on the values that need to be sent and received and their respective meanings. Although the values can be modified, it's important to ensure that an adequate number of values are maintained for each element.

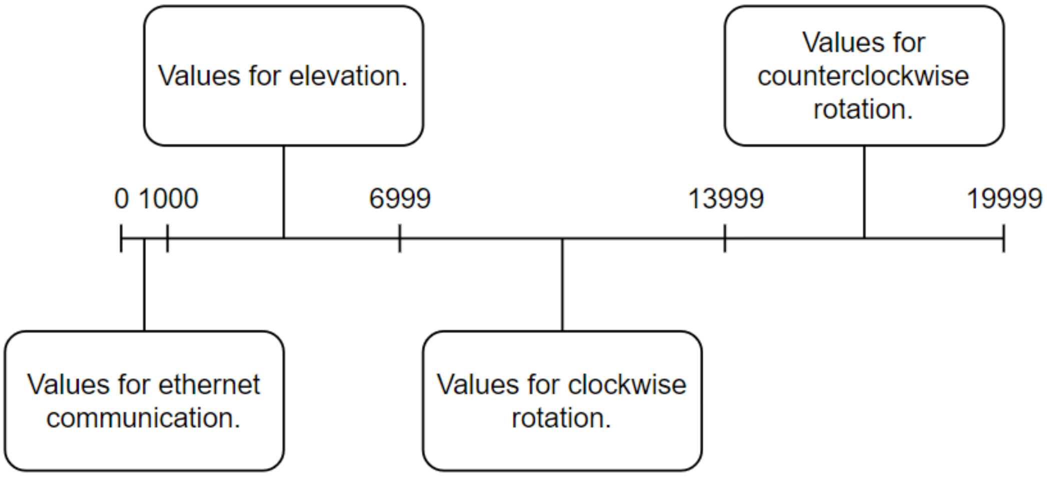

Values for elevation.
Values for counterclockwise rotation.
0 1000
6999
13999
19999
Values for ethernet communication.
Values for clockwise rotation.

Figure 1: Representation of the number values used for Ethernet and stepper motor communication.

D VeloView – real-time visualization

Requirement FR-01 states that the sensor for the system shall detect objects. The detection of objects requires a sensor, and that the sensor has the ability to recognize newly appeared objects in an area. The data delivered by the LiDAR sensor requires filtering to retrieve the desired data required for object detection. This is one of the options as to how data could be retrieved from the LiDAR sensor.

VeloView is a VeloDyne LiDAR viewer which is based of ParaView LiDAR, an open source analysis and visualization application. [120] VeloView has real-time visualization and processing abilities and provides a great view of the data retrieved from the sensor. [30] Points are plotted where objects are detected, making a picture, or a frame of any room and area within the chosen sensors limitations. The plotted points are continuously saved in a spreadsheet covering all information on each point such as azimuth, vertical angle, distance in meter and the points corresponding laser ID. The LiDAR sensor has a specific number of lasers depending on which sensor is chosen.

VeloView has tools and modules making the visualization of the retrieved data simple and user friendly. An example of the modules in VeloView is the Python Shell Console. The console enables users to execute commands or run pre-made scripts, allowing a modification and manipulation of the displayed retrieved data by selectively filtering out the desired information. This also gives the user the option to execute the application from the command line, following a pre-made script which runs the application with the functionality written in the script. By using the Python console, the desired information required for detecting an object could be filtered out. This is optimal as it is specific information on specific points and objects in the frame that is interesting for the project.

All data for each plotted point is viewed in a spreadsheet in VeloView that could be exported. The spreadsheet is updated continuously as the points are visualized. When the spreadsheet is exported, the plotted points for this moment is saved, making the spreadsheet past sense. For the object detection to be real-time, the saved data should be accessed and updated continuously.

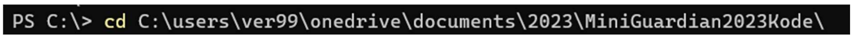
E GitHub tutorial for group members

Hvordan laste ned koden fra GitHub:

Først lager du en mappe på et passende sted lokalt på pcen din der du vil ha koden til prosjektet.

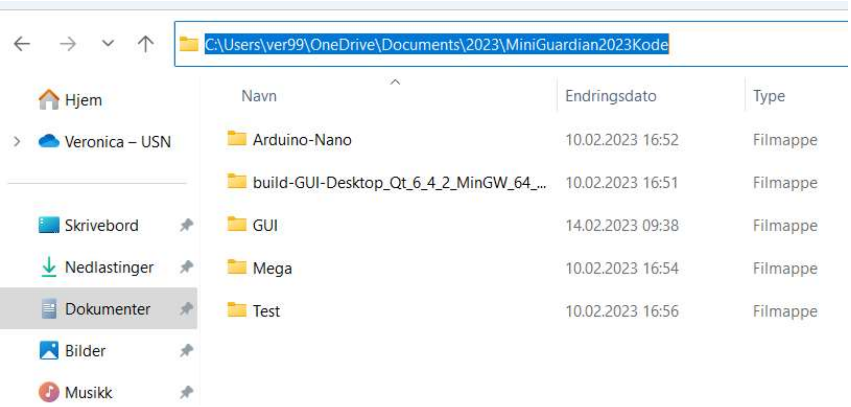
Så åpner du kommandolinjen (søk «cmd» i søkefeltet, eller åpne powershell eller bash).

Gå til filstien til mappen du lagde for koden. Dette gjør du ved å skrive «cd» etterfulgt av filstien:



```
PS C:\> cd C:\users\ver99\onedrive\documents\2023\MiniGuardian2023Kode\
```

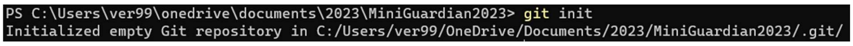
Er du usikker på hvordan du finner filstien kan du gå til mappen lokalt på pcen, høyreklikke på den og trykke Åpne i terminal. Du kan også gå til mappen i filutforskeren og markere og kopiere stien, som dette:



C:\Users\ver99\OneDrive\Documents\2023\MiniGuardian2023Kode

Navn	Endringsdato	Type
Arduino-Nano	10.02.2023 16:52	Filmappe
build-GUI-Desktop_Qt_6_4_2_MinGW_64_...	10.02.2023 16:51	Filmappe
GUI	14.02.2023 09:38	Filmappe
Mega	10.02.2023 16:54	Filmappe
Test	10.02.2023 16:56	Filmappe

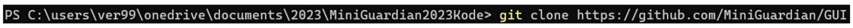
Du må nå gjøre mappen til et git repo. Skriv «git init»:



```
PS C:\Users\ver99\onedrive\documents\2023\MiniGuardian2023> git init
Initialized empty Git repository in C:/Users/ver99/OneDrive/Documents/2023/MiniGuardian2023/.git/
```

Nå skal du laste ned koden inn i mappen din. Gå til repositoriet inne i GitHub, f.eks GUI, og kopier URLen (https://github.com/MiniGuardian/GUI)

Gå til kommandolinjen, skriv «git clone» etterfulgt av URLen:



```
PS C:\users\ver99\onedrive\documents\2023\MiniGuardian2023Kode> git clone https://github.com/MiniGuardian/GUI
```

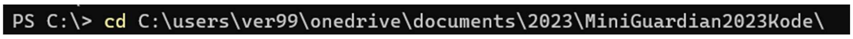
Dersom du nå skriver «ls» (eller «dir» om ls ikke fungerer), vil du se at GUI har lagt seg i mappen din. Det samme vil også skje lokalt på pcen din.

Gjør nå det samme for alle repositoriene du trenger. (Test og Hello-world er repoer for testing).

Hvordan redigere kode fra GitHub:

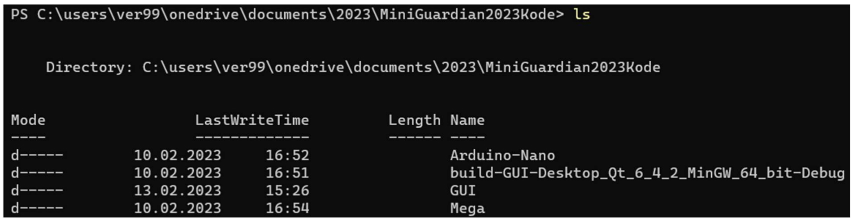
Åpne kommandolinjen (søk etter «cmd» i søkefeltet, eller åpne powershell eller bash)

Gå til filstien til mappene som inneholder koden til prosjektet. Skriv «cd» før filstien for å komme dit, som på eksempelet under:



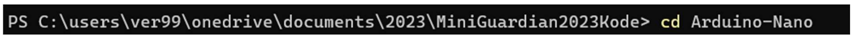
```
PS C:\> cd C:\users\ver99\onedrive\documents\2023\MiniGuardian2023Kode\
```

Når du har kommet hit, skriv «ls» (eller «dir» om ls ikke fungerer), for å få opp mappene:



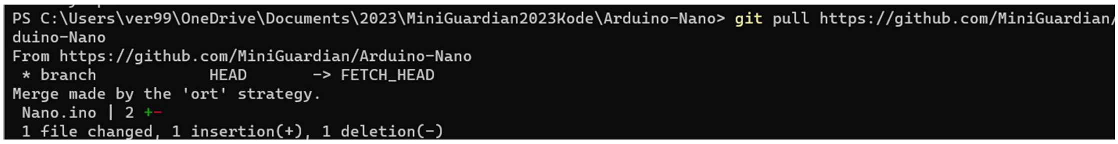
```
PS C:\users\ver99\onedrive\documents\2023\MiniGuardian2023Kode> ls


    Directory: C:\users\ver99\onedrive\documents\2023\MiniGuardian2023Kode


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        10.02.2023     16:52                Arduino-Nano
d-----        10.02.2023     16:51                build-GUI-Desktop_Qt_6_4_2_MinGW_64_bit-Debug
d-----        13.02.2023     15:26                GUI
d-----        10.02.2023     16:54                Mega
```

Skriv så «cd» etterfulgt av koden du skal endre, f.eks Arduino-Nano:
(Når du skal ut av mappen senere skriver du «cd ..»)



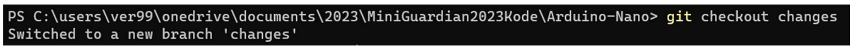
```
PS C:\users\ver99\onedrive\documents\2023\MiniGuardian2023Kode> cd Arduino-Nano
```

NB!!! For å forsikre deg om at du har riktig og oppdatert kode lastet ned, skriver du «git pull» etterfulgt av URLen til GitHub repositoriet for å oppdatere koden. Dersom du får beskjeden «Already up to date» betyr det at det ikke var noen endringer å oppdatere. Får du opp noe som dette, betyr det at noe har blitt oppdatert:



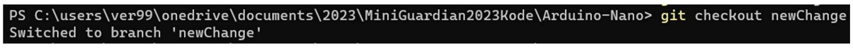
```
PS C:\Users\ver99\OneDrive\Documents\2023\MiniGuardian2023Kode\Arduino-Nano> git pull https://github.com/MiniGuardian/
duino-Nano
From https://github.com/MiniGuardian/Arduino-Nano
 * branch            HEAD       -> FETCH_HEAD
Merge made by the 'ort' strategy.
 Nano.ino | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)
```

Nå må du inn på noe kalt en branch for at koden som endres, ikke skal redigere hovedkoden liggende på GitHub. En branch er en gren ut av hovedkoden, dette forsikrer at hovedkoden forblir urørt. Skriv først «git checkout» etterfulgt av et passende navn for dine endringer.



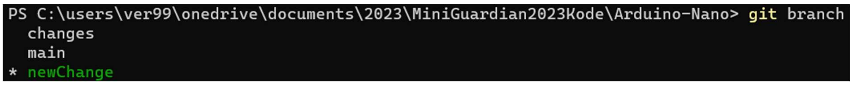
```
PS C:\users\ver99\onedrive\documents\2023\MiniGuardian2023Kode\Arduino-Nano> git checkout changes
Switched to a new branch 'changes'
```

Nå går du inn på den nye branchen ved å skrive «git checkout» etterfulgt av navnet på branchen:



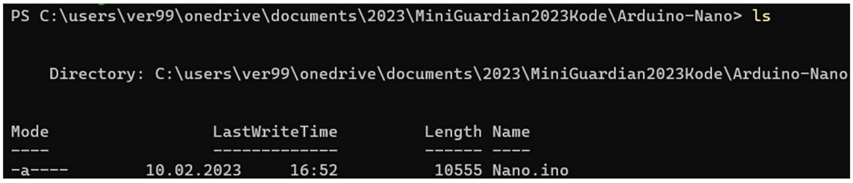
```
PS C:\users\ver99\onedrive\documents\2023\MiniGuardian2023Kode\Arduino-Nano> git checkout newChange
Switched to branch 'newChange'
```

For å forsikre deg om at du er inne i riktig branch, skriv «git branch». Branchen som er grønn med tilhørende stjerne er branchen du er inne i: (Du skal ikke være i main/master eller changes)



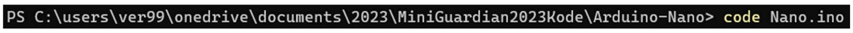
```
PS C:\users\ver99\onedrive\documents\2023\MiniGuardian2023Kode\Arduino-Nano> git branch
  changes
  main
* newChange
```

Herfra kan du skrive «ls» igjen for å se filene inne i mappen.



```
PS C:\users\ver99\onedrive\documents\2023\MiniGuardian2023Kode\Arduino-Nano> ls


    Directory: C:\users\ver99\onedrive\documents\2023\MiniGuardian2023Kode\Arduino-Nano


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        10.02.2023     16:52          10555 Nano.ino
```

For å kunne endre denne koden, må du åpne et program. Skriv «code» etterfulgt av filnavnet. Dette vil åpne Visual Studio Code med koden. Gjerne ha kommandolinjen åpen i bakgrunnen så slipper du gå gjennom alle stegene igjen når du er ferdig med endringene.

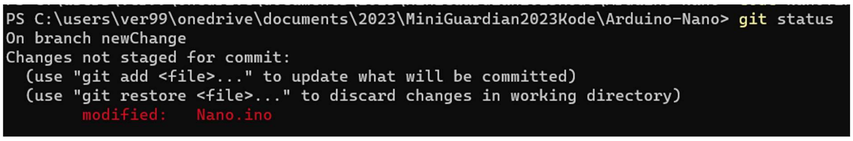


```
PS C:\users\ver99\onedrive\documents\2023\MiniGuardian2023Kode\Arduino-Nano> code Nano.ino
```

NB!!! Husk å lagre i programmet når du endrer koden.

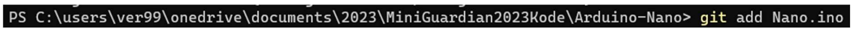
Når du har lagret dine endringer, går du inn i kommandolinjen igjen, til samme filsti og samme branch. Husker du ikke hvordan du kom dit, gå til første punkt.

Skriv «git status» inne i mappen for å se det du har endret. Under står filnavnet i rød skrift under 'modified'. Dette betyr at filen er endret.



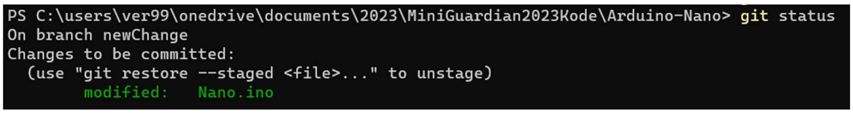
```
PS C:\users\ver99\onedrive\documents\2023\MiniGuardian2023Kode\Arduino-Nano> git status
On branch newChange
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   Nano.ino
```

Skriv deretter «git add» etterfulgt av filnavnet, eller punktum for å velge alle filer.



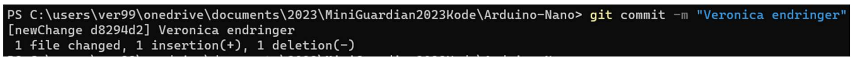
```
PS C:\users\ver99\onedrive\documents\2023\MiniGuardian2023Kode\Arduino-Nano> git add Nano.ino
```

Skriv «git status» igjen, nå skal filen ha grønn farge.



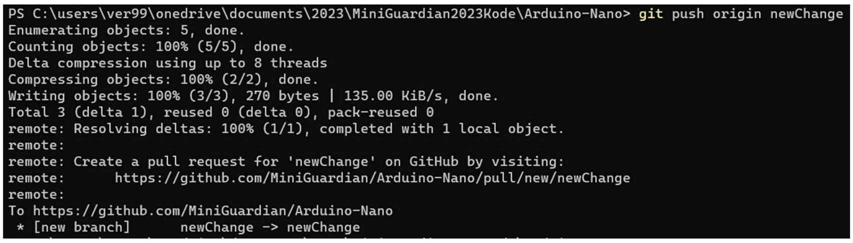
```
PS C:\users\ver99\onedrive\documents\2023\MiniGuardian2023Kode\Arduino-Nano> git status
On branch newChange
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   Nano.ino
```

Nå skriver du «git commit -m ''Navn endringer''»: (Kommentaren er ikke så viktig)



```
PS C:\users\ver99\onedrive\documents\2023\MiniGuardian2023Kode\Arduino-Nano> git commit -m "Veronica endringer"
[newChange d8294d2] Veronica endringer
 1 file changed, 1 insertion(+), 1 deletion(-)
```

For å sende endringene til GitHub til «godkjenning», må filene pushes. Skriv «git push origin» etterfulgt av navnet på branchen du er på, i dette tilfellet er det newChange. Du vil få opp en del meldinger.



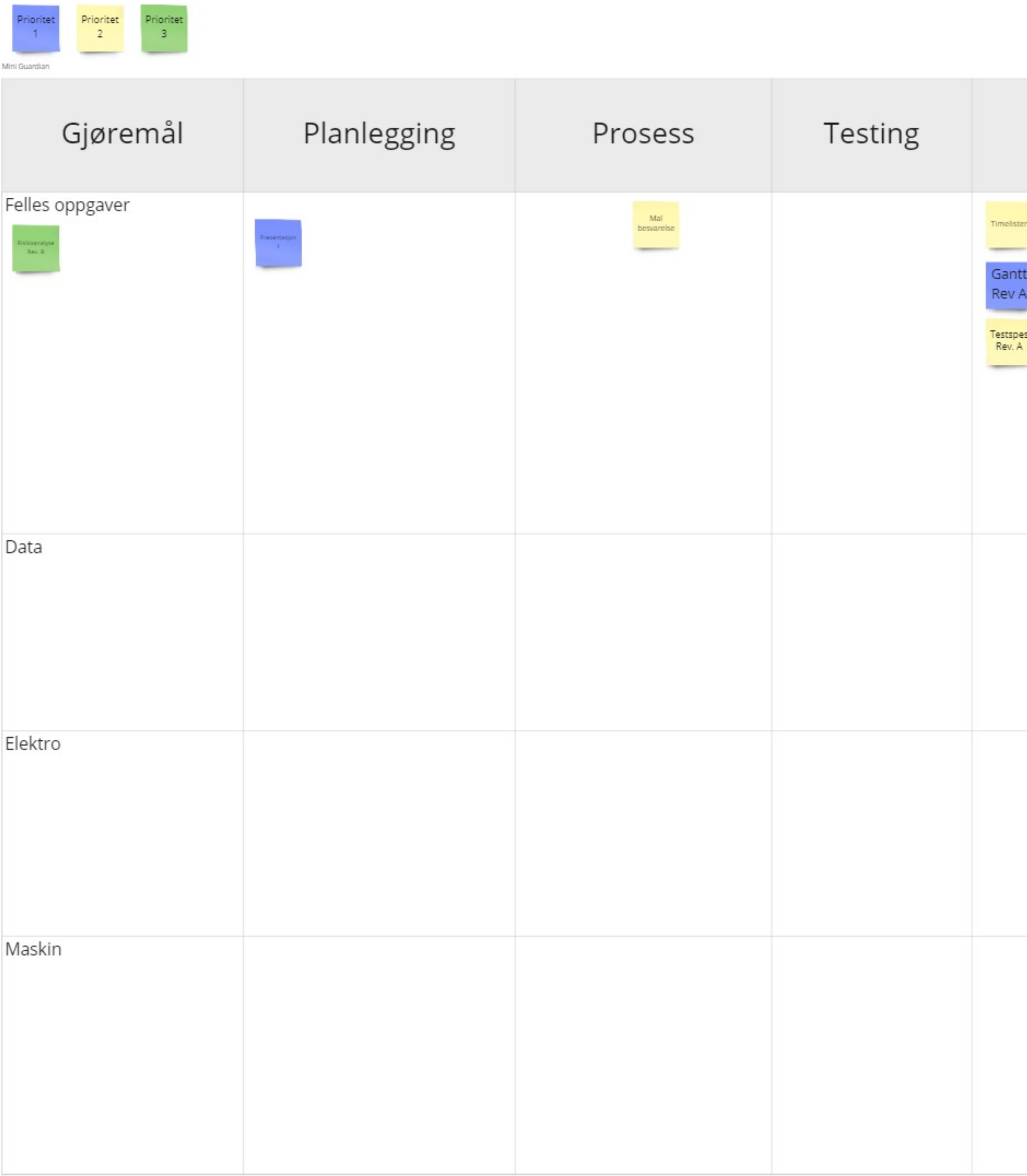
```
PS C:\users\ver99\onedrive\documents\2023\MiniGuardian2023Kode\Arduino-Nano> git push origin newChange
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 270 bytes | 135.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
remote:
remote: Create a pull request for 'newChange' on GitHub by visiting:
remote:      https://github.com/MiniGuardian/Arduino-Nano/pull/new/newChange
remote:
To https://github.com/MiniGuardian/Arduino-Nano
 * [new branch]      newChange -> newChange
```

Hvis du nå skriver «git status» vil alt være oppdatert.

FERDIG 😊

F Digital Scrumban board week 3

This Appendix shows the first digital version of the Scrumban board.



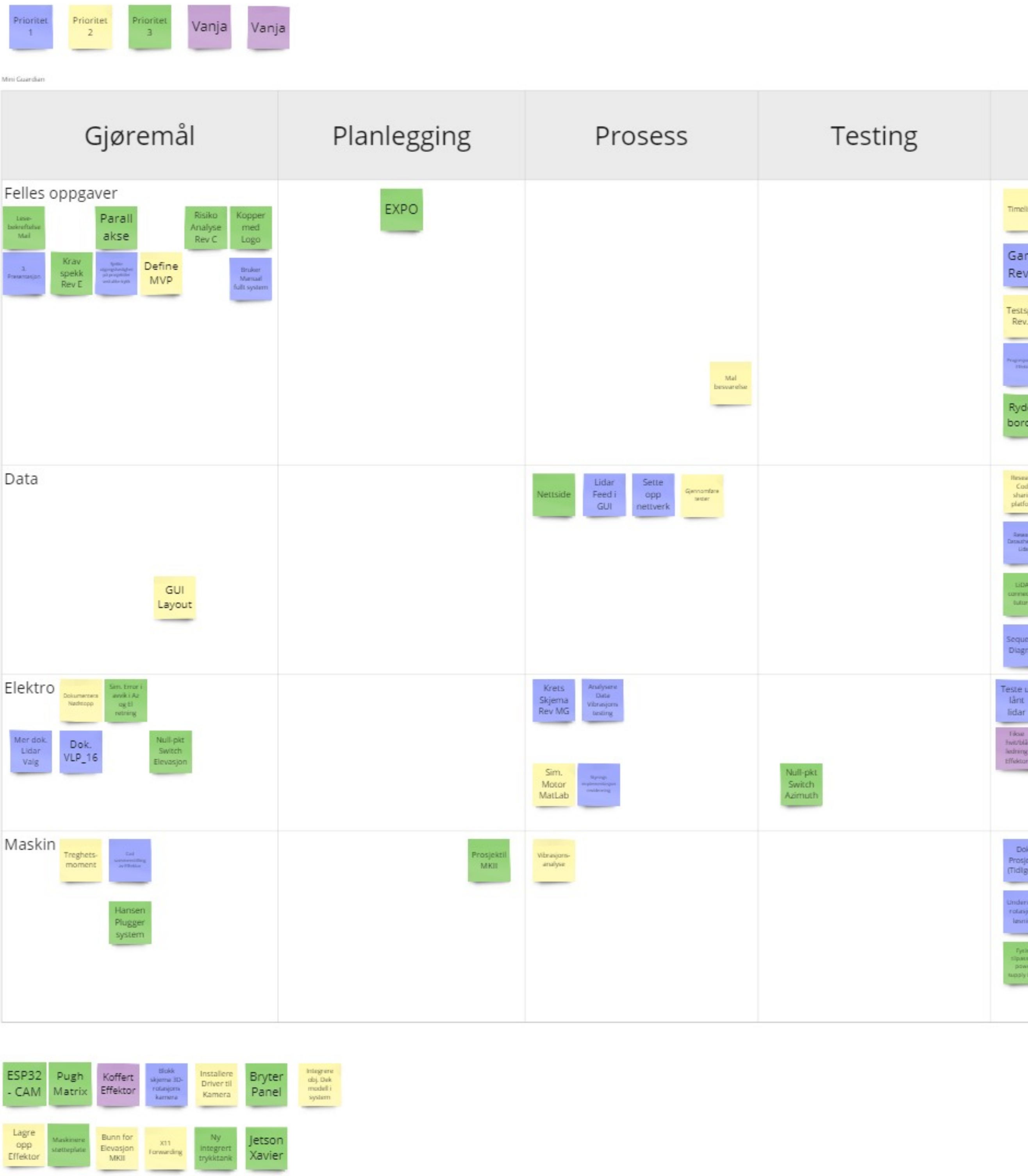
Prioritet 1 | Prioritet 2 | Prioritet 3

Mini Guardian

Gjøremål	Planlegging	Prosess	Testing	
Felles oppgaver Risikoanalyse Rev. B	Presentasjon 1	Mal besvarelse		Timelister Gantt Rev A Testspes Rev. A
Data				
Elektro				
Maskin				

G Digital Scrumban board week 19

This Appendix shows the last digital version of the Scrumban board.



Prioritet 1 | Prioritet 2 | Prioritet 3 | Vanja | Vanja

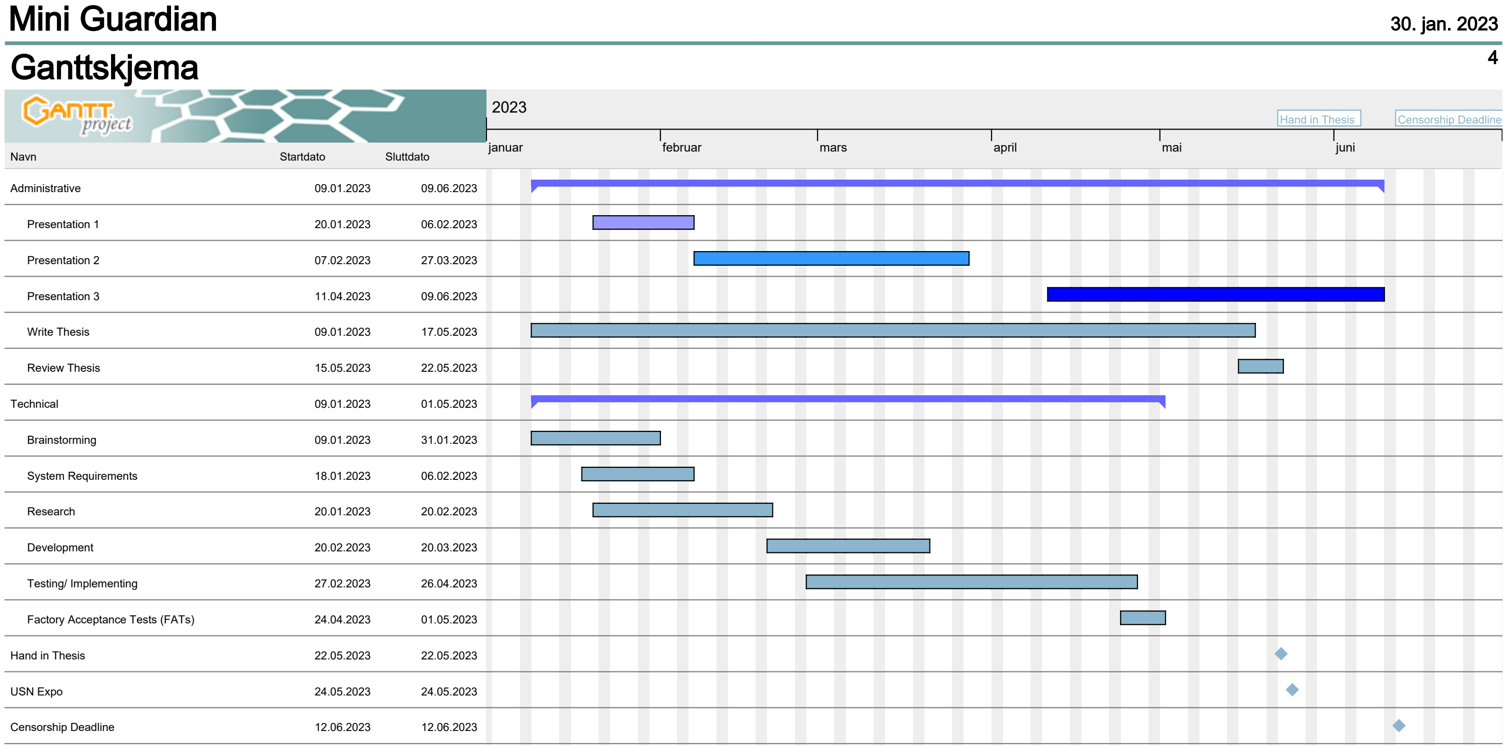
Mini Guardian

	Gjøremål	Planlegging	Prosess	Testing	
Felles oppgaver	Lese-bekreftelse Mail Parall akse Risiko Analyse Rev C Kopper med Logo 3. Presentasjon Krav spekk Rev E [illegible] Define MVP Bruker Manual fullt system	EXPO	Mal besvarelse		Timeli Gar Rev Tests Rev. [illegible] Ryd bord
Data	GUI Layout		Nettside Lidar Feed i GUI Sette opp nettverk Gjennomføre tester		Resea Cod shari platfo Resea Datash Lid LIDA connec tutor Seque Diagr
Elektro	Dokumentere Nødstopp Sim. Error i avvik i Az og El retning Mer dok. Lidar Valg Dok. VLP_16 Null-pkt Switch Elevasjon		Krets Skjema Rev MG Analysere Data Vibrasjons testing Sim. Motor MatLab [illegible]	Null-pkt Switch Azimuth	Teste u lånt lidar Fikse hvit/blå ledning Effektor
Maskin	Treghets-moment [illegible] Hansen Plugger system	Prosjektil MKII	Vibrasjons-analyse		Dok Prosj (Tidlig Under rotasj lasni Fysi tilpass pow supply

ESP32 - CAM | Pugh Matrix | Koffert Effektor | Blokk skjema 3D-rotasjons kamera | Installere Driver til Kamera | Bryter Panel | Integrere obj. Dek modell i system

Lagre opp Effektor | Maskinere støtteplate | Bunn for Elevasjon MKII | X11 Forwarding | Ny integrert trykktank | Jetson Xavier

H Gantt schematic Revision A.



Mini Guardian

30. jan. 2023

Ganttskjema

4

Navn	Startdato	Sluttdato
Administrative	09.01.2023	09.06.2023
Presentation 1	20.01.2023	06.02.2023
Presentation 2	07.02.2023	27.03.2023
Presentation 3	11.04.2023	09.06.2023
Write Thesis	09.01.2023	17.05.2023
Review Thesis	15.05.2023	22.05.2023
Technical	09.01.2023	01.05.2023
Brainstorming	09.01.2023	31.01.2023
System Requirements	18.01.2023	06.02.2023
Research	20.01.2023	20.02.2023
Development	20.02.2023	20.03.2023
Testing/ Implementing	27.02.2023	26.04.2023
Factory Acceptance Tests (FATs)	24.04.2023	01.05.2023
Hand in Thesis	22.05.2023	22.05.2023
USN Expo	24.05.2023	24.05.2023
Censorship Deadline	12.06.2023	12.06.2023