Kristoffer Halvorsen

Developing of a Shuttlecock Launcher using 3D Printing Technology and Readily Available Components

Master's thesis in Mechanical Engineering
Supervisor: Amund Skavhaug
June 2023

Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Engineering
Department of Mechanical and Industrial Engineering

NTNU
Norwegian University of
Science and Technology

Kristoffer Halvorsen

Developing of a Shuttlecock Launcher using 3D Printing Technology and Readily Available Components

Master’s thesis in Mechanical Engineering
Supervisor: Amund Skavhaug
June 2023

Norwegian University of Science and Technology
Faculty of Engineering
Department of Mechanical and Industrial Engineering

NTNU
Norwegian University of
Science and Technology

NTNU
Kunnskap for en bedre verden

Department of Mechanical and Industrial Engineering

TPK4960 - Robotics and Automation

Developing of a Shuttlecock Launcher using 3D Printing Technology and Readily Available Components

Author:

Kristoffer Halvorsen

Supervisor:

Amund Skavhaug

11.06.2023

Preface

This Master's thesis, authored by Kristoffer Halvorsen, represents the completion of my Master's degree at the Norwegian University of Science and Technology (NTNU).

I would like to express my deep appreciation for the unique opportunity that this thesis has provided. My sincere gratitude goes out to my supervisor, Amund Skavhaug. His guidance, invaluable assistance, and the provision of necessary equipment have been pivotal throughout the process. His depth of expertise and enlightening insights have been crucial in developing and writing this thesis.

My motivation was primarily driven by a desire to broaden my knowledge in the field of mechatronics, with a particular focus on electronics and programming. Aware of my background as a bachelor of mechanical engineering, Amund proposed a task that catered perfectly to my aspirations. The task involved developing a badminton launcher from scratch, allowing me to leverage the foundational knowledge gained during my bachelor's studies while deepening my understanding of electronics and programming.

I hope that this thesis not only contributes to the advancement of engineering knowledge but also serves as a motivation for future research and innovation.

K. Halvorsen

Kristoffer Halvorsen

Summary

In this Master's thesis, the primary aim was to engineer a cost-effective and user-friendly Shuttlecock Launcher that could replace the hand-feeding operation in badminton training. 3D printing technology and readily available components were employed to achieve this objective. The goal was to develop an automatic feeding mechanism for the shuttlecock, which could be conveniently assembled, and shuttlecocks should preferably be launched half a distance of a standardised court and directed in multiple directions.

The methodologies used in this study included rapid prototyping, static simulation and topology study, 3D printing, and slow-motion video recording. Rapid prototyping was instrumental in creating and analysing physical models quickly. This process involved 3D modelling and 3D printing. The components of the shuttlecock launcher were evaluated using static simulation and topology optimisation techniques to ensure an optimal design.

3D printing, combined with computer-aided design modelling, enabled the efficient production of the shuttlecock launcher prototype components. This process allowed for rapid testing and iterative design adjustments. The device's functionality was observed in detail through slow-motion video recording using an iPhone 14 Pro.

The final prototype successfully demonstrated its ability to replace manual hand-feeding in badminton training. However, for flawless usability, certain improvements are required. The achieved launch distances did not fully meet the target range of 6-8 meters, but the averages were sufficient for basic to intermediate-level training sessions.

All the design files for the shuttlecock launcher are made open-source, promoting future improvements and adaptations. This aspect and the use of cost-effective components make the shuttlecock launcher a viable and compelling alternative to existing products on the market. Despite not fully achieving the projected objectives and performance goals, the shuttlecock launcher has consistently performed satisfactorily, promising future development and improvement.

In conclusion, this study has successfully achieved its objective of creating an automated, cost-effective, and versatile shuttlecock launcher, promoting the use of readily available components and 3D printing technology. It also highlights the potential of using widely accessible tools like smartphones for comprehensive real-time study.

Sammendrag

I denne masteroppgaven var det primære målet å konstruere en kostnadseffektiv og brukervennlig badmintonkanon som kunne erstatte manuell serving i badminton trening. 3D-printteknologi og lett tilgjengelige komponenter ble benyttet for å realisere dette målet. Hensikten var å utvikle en automatisk matemekanisme for fjærballer, som lett kunne monteres. Videre skulle det være mulig å fyre av en fjærballene slik at den nådde halve avstanden av en standardisert bane, og kunne rettes i flere retninger.

Metodene som ble tatt i bruk i studien omfattet hurtig prototyping, statisk simulering og topologistudier, 3D-printing, samt slow-motion videoopptak. Hurtig prototyping viste seg å være nyttig for å raskt lage og analysere fysiske modeller. Denne prosessen involverte 3D-modellering og 3D-printing. Badmintonkanonens komponenter ble også evaluert ved hjelp av statisk simulering og topologioptimaliseringsteknikker for å sikre et best mulig design.

Kombinasjonen av 3D-printing og Dataassistert konstruksjon (DAK)-modellering muliggjorde effektiv fremstilling av badmintonkanonens komponenter. Denne prosessen tillot rask testing og iterativ justering av design. Enhetsfunksjonalitetene ble observert i detalj gjennom slow-motion videoopptak ved hjelp av en iPhone 14 Pro.

Den endelige prototypen demonstrerte vellykket evnen til å erstatte manuell serving. Imidlertid er det behov for visse forbedringer for å sikre feilfri brukervennlighet. De oppnådde skuddavstandene møtte ikke fullt ut målområdet på 6-8 meter, men gjennomsnittene viste seg å være tilstrekkelige for grunnleggende til mellomliggende treningsøkter.

Samtlige designfiler for badmintonkanonen er gjort tilgjengelig for offentligheten for å oppmuntre til fremtidige forbedringer og tilpasninger. Denne tilnærmingen, sammen med bruken av kostnadseffektive komponenter, gjør badmintonkanonen til et overbevisende alternativ til eksisterende produkter på markedet. Til tross for at de ytelsesmålene ikke ble fullt ut oppnådd, har badmintonkanonen jevnlig levert tilfredsstillende resultater, noe som åpner for fremtidig utvikling og forbedring.

I konklusjon har denne studien lykkes i å realisere målet om å skape en automatisert, kostnadseffektiv, og allsidig badmintonkanon. Den fremmer bruk av lett tilgjengelige komponenter og 3D-printteknologi, og understreker potensialet ved å bruke allment tilgjengelige verktøy som smarttelefoner for detaljert sanntidsstudie.

Table of Contents

List of Figures

List of Tables

Acronyms

BLDC	Brushless DC electric motor
CAD	Computer-aided design
CLT	Classical laminate theory
ESC	Electronic speed controller
E3P	Creality ender 3 pro
FDM	Fused deposition modeling
FEA	Finite element analysis
GPIO	General-purpose input/output
HSE	Health, Safety and Environment
IDE	Integrated Development Environment
MSA	Motor System A
MSB	Motor System B
NTNU	Norges Teknisk-Naturvitenskapelige Universitet
PCB	Printed Circuit Board
PWM	Pulse-Width Modulation
SCL	Shuttlecock launcher
SC	Shuttlecock
SW	SolidWorks

1 Introduction

This thesis is dedicated to designing a brand new badminton Shuttlecock launcher (SCL) concept, and builds upon the work initiated during the pre-project phase 2.

This thesis aims to ease the build process for others, as well as further develop the design of the SCL. The focus is not to develop a machine used for professional sports clubs or high-budget training facilities; the target audience is enthusiastic hobbyists. This includes individuals interested in badminton who want to elevate their game as well as those fascinated by 3D printing, seeking to explore new domains within their hobby. This thesis provides in-depth documentation of the development process and shares the necessary insights building upon previous work.

The broader vision for this thesis is to foster a community of developers interested in the SCL. Doing so aims to stimulate continuous improvement and innovation of the SCL. The aspiration is to create an open-source, accessible machine, allowing anyone to build and use one, as well as contribute to its development, thereby advancing the badminton training experience.

1.1 Motivation

Badminton is a racket sport played by two or four players on a standard court measuring 6.1 meters in width and 13.4 meters in length [1]. The game involves hitting an Shuttlecock (SC), also known as a birdie or shuttle, back and forth over a net to land the SC in the opponent's court without letting it touch the ground on one's side of the court [2].

For beginners, mastering the correct technique is essential. A training session often requires a third player to manually serve SCs either by hand or with a racket while the coach observes and guides the technique performance. However, in certain scenarios where a third player is unavailable, the coach must serve the SC while simultaneously observing the player's technique. Even worse, most often beginners don't have a coach available most of the time. These players often find it challenging to serve SCs precisely to each other for training purposes. However, SCL machines are available to address this challenge. An SCL, also known as an automatic badminton SCL, or badminton feeder, can benefit players looking to improve their badminton skills.

Firstly, an SCL can help players practice their shots without needing a third party. This can be useful for players who may not have a partner to practice with or those who want to train alone. Another benefit of an SCL is that it can help players increase their stamina and endurance. The machine can deliver shots faster than a human opponent, providing a more intense workout and improving physical conditioning. But most importantly, it can throw an SCs accurately in the same direction several times.

An SCL can also help players improve their consistency and accuracy by repeatedly delivering shots with the same speed, trajectory, and spin. Thus, players can focus on technique and footwork without worrying about the SCs flight path, resulting in more efficient practice sessions and quicker improvement.

1.2 Problem Description

The SCL available today can be broadly classified into three categories: simple and inexpensive, medium advanced and expensive, and very expensive.

An inexpensive SCL, illustrated in Figure 1, has an approximate cost of NOK 710, excluding tax or shipping [3]. This machine performs the basic function of launching SCs in one direction.

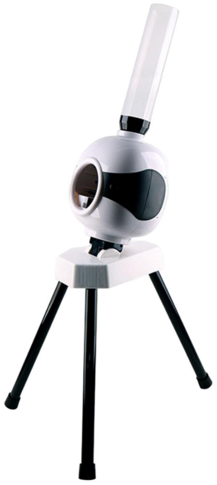


Figure 1: Illustration of an inexpensive SCL [3].

A medium advanced SCL, depicted in Figure 2, has a wider range of features. These machines, priced around NOK 21,000 without tax and shipping [4], offer adjustable speed, control over launching axes, a larger SC capacity, and remote-control functionality.

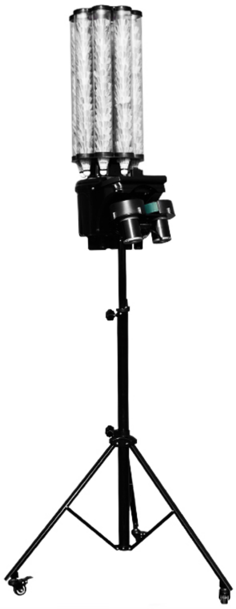


Figure 2: Illustration of an advanced SCL [4].

Lastly, the high-end, very expensive SCL, shown in Figure 3, is priced at NOK 140,000, exclusive of tax and shipping [5]. Besides being able to launch SCs autonomously, these machines have programmable routines for simulating real game scenarios. The speed and direction of launched SCs are adjustable, making these machines versatile and ideal for professional training.

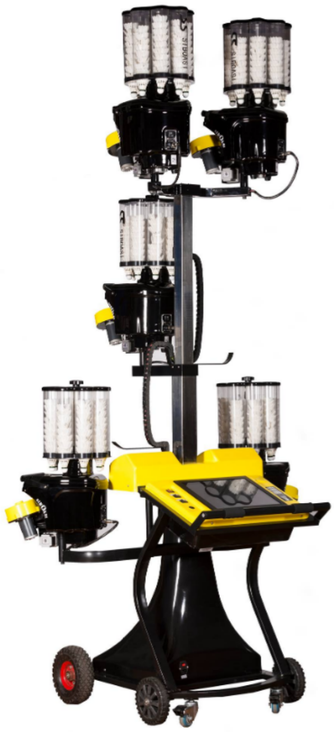


Figure 3: Example of a high-end SCL [5].

These SCL, particularly the advanced and high-end models, poses various challenges. One of the primary issues is their substantial size and weight, which complicates installation, transportation, and storage. This characteristic also restricts their use, especially among younger players or trainers who might find them difficult to operate. Furthermore, brand-specific machines often need exclusive spare parts when they malfunction. As a result, users are inconvenienced by manually serving SCs until the machine can be repaired or replaced.

Additionally, the high-end and advanced SCLs often demand specialised metalworking tools for any physical repairs or modifications. On the software side, they frequently rely on exclusive systems, introducing problems if the software malfunctions or becomes outdated. This protected code and lack of accessible debugging tools can render users helpless in fixing these issues.

However, increased accessibility of 3D printing technology and the affordability of open-source microcontrollers constitute a potential solution to these challenges.

The BADDY project [6], now discontinued, once embraced similar purposes. This initiative was the subject of various master's projects, yet it was also discontinued. Its semi-open nature, limited user accessibility, and unstable foundation were contributing factors to its discontinuation [7]. Furthermore, BADDY did not utilise readily accessible parts and equipment, which presented an additional hurdle for widespread adoption and maintenance.

Consider a concept for an open-source, brand-independent SCL, designed to be assembled, maintained, and repaired using basic hobby tools, 3D printing technology, open-source software, and programmable microcontrollers. This approach could address the problems currently experienced. The model, conceived using universally accessible hardware and software, aims to democratise access to this technology. This could pave the way for a safer, more cost-effective, and more manageable environment for users across a wide spectrum.

By implementing this solution, it could substantially reduce downtime and costs. It also promotes a more inclusive and innovative sporting community. Integrating open-source software and microcontrollers offers a further advantage as it allows the community to resolve software and control issues, fostering an environment conducive to collaborative and independent problem-solving. This approach could prompt further innovation by utilising the capabilities and passion within the 3D printing, open-source software, and hobbyist electronics communities.

1.3 Objectives

Building upon initial objectives and the lessons learnt from the pre-project, this thesis aims to extend the development of the SCL to replace hand feeding as the primary objective. The SCL designed in this thesis aims to be constructed lightweight for simplified transportation and storage. Moreover, it aims to be an open-source, affordable solution assembled from easily available parts, encouraging community-driven development and innovation.

The specific objectives of this thesis are:

- **Design & Accessibility:**
  - Develop an SCL using easily available components to the general public, fostering a community-driven development process.
  - Ensure the design of the SCL is compatible with a standard 3D printer equipped with a heated bed.
- **Cost & Construction:**
  - Build the SCL with cost-effective components, including parts that can be fabricated using 3D printing technology.
  - Design a lightweight SCL, promoting ease of transport and storage.
- **Performance & Versatility:**
  - Configure the SCL to be versatile, capable of launching SC at adjustable speeds and varying directions.
- **Future Development:**
  - Design the SCL to enable future enhancements and development of both hardware and software.

By achieving these objectives, this thesis aims to significantly contribute to sports training equipment and establish a foundation for open-source, accessible, and continuously improvable SCL design.

In this process, it is noteworthy that while the objectives from the pre-project phase might seem less relevant now, they have been integrated into the current objectives. The key objectives of the pre-project phase were:

- *Design a cost-effective SCL that can be recreated by anyone with a 3D printer.*
- *Ensure user-friendliness with standardised size and market accessibility.*
- *Create an automatic feeding mechanism for the SCs.*
- *Provide an open design for future development, such as the integration of sensors or other mechanisms.*
- *Develop a compact design.*
- *Engineer the machine to launch SC across a standardised court.*
- *Design the machine to shoot in different directions.*

1.4 Performance Goals

Performance goals have also been set. The SCL developed in this thesis is designed to meet specific performance goals, with a focus on the practical operation of the machine and its applicability for badminton training:

- *Shoot SCs distance up to a maximum of 6-8 meters.*
- *Achieve reliable, jam-free operation to prevent training disruptions and potential machine damage.*
- *Ensure straightforward assembly, thereby increasing the accessibility and practicality of the device.*

1.5 Limitations

As approved by the thesis supervisor, Amund Skavhaug, the primary focus of this thesis is the development and testing of mechanisms for a functional SCL. The key areas of investigation include the feeder mechanism, the selection of suitable wheels to launch the SCs, and the basic mechatronics component, which contains hardware selection, elementary programming, and soldering.

1.6 Report Structure

This thesis is organised in a sequence of 13 chapters, designed to guide the reader through the intricacies of the thesis with ease. The introductory section is followed by chapters 2-4, clarifying the background from the pre-project, then theoretical underpinnings, and the methodological approach used. These chapters form the cornerstone of the research, underpinned by preceding work that directly guides the study in this thesis.

The following sections, from chapter 5 through to chapter 9, maintain a consistent structure to boost readability and comprehension. Each chapter spotlights a particular stage or component of the SCL development, separating the process systematically and comprehensively.

Each chapter begins with a general problem statement and a series of potential solutions tested using methods. For the purpose of clarity and improved understanding, relevant discussions and conclusions are integrated throughout the individual chapters. This structure is designed to aid the reader in understanding the sequential progression of the thesis.

A comprehensive construction review succeeds this in chapter 10, where all developed parts are presented, each with an exhaustive explanation. The chapter concludes with a choice of material and an estimated production cost for the SCL.

Following the construction review, chapter 11 is devoted to the experimental testing, focusing on the SCL performance. Subsequently, a discussion chapter 12 delves into both the outcomes of the experimental testing and the overall development process. The thesis concludes with a final chapter, summarising the results and drawing the a conclusion.

Please note that for PDF readers, all references in this thesis are clickable, directing the reader to the cited source when required. This feature enhances the thesis's user-friendly nature, making it easier for readers to trace the sources of information used throughout the work.

An overview of the chapters:

- **Chapter 1 - Introduction:** Sets the stage with the study's motivation, problem statement, objectives, and limitations.
- **Chapter 2 - Background Material:** Presents results from the pre-project, providing context for the thesis.
- **Chapter 3 - Theory:** Introduces essential theoretical concepts and background knowledge.
- **Chapter 4 - Method:** Describes methodologies employed during the SCL

development.

- **Chapter 5 - Wheel Selection:** Explains the rationale and decision-making process for wheel selection.
- **Chapter 6 - Pan Tilt Mechanism Development:** Discusses the development and implementation of the Pan Tilt mechanism.
- **Chapter 7 - Feeding Mechanism Evolution:** Reviews the initial feeding mechanism, identifying the need for a new version and detailing its development.
- **Chapter 8 - Transition from Raspberry Pi to Arduino:** Outlines transitioning from Raspberry Pi to Arduino.
- **Chapter 9 - Adjustments and Testing:** Documents the iterative refinement and testing of the SCL.
- **Chapter 10 - Construction Review:** Offers an overview of the final configuration of the SCL, detailing each component's function and estimating production cost.
- **Chapter 11 - Experimental Testing of the SCL:** Reports on experimental testing and performance results of the SCL.
- **Chapter 12 - Discussion:** Reflects on the results and development process, dissecting outcomes and implications.
- **Chapter 13 - Conclusion:** Concludes the thesis findings, achievements, and contributions to the field of badminton training equipment.

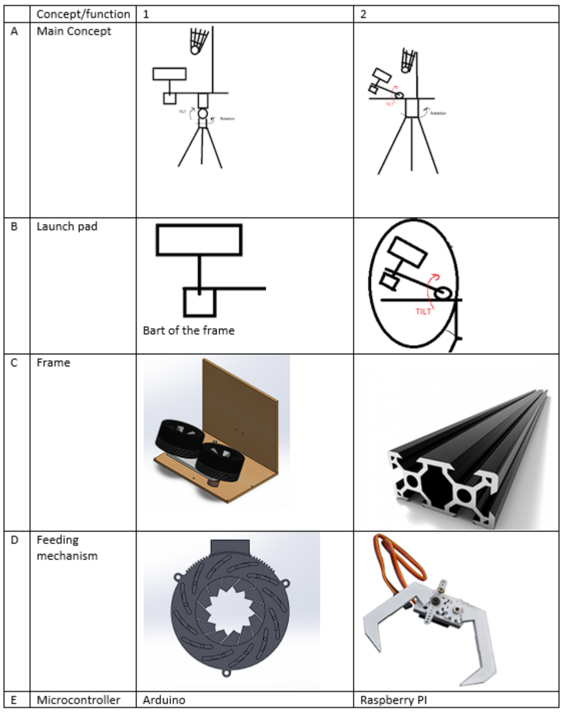
2 Background Material

This chapter provides an overview of the pre-project and serves as this thesis's foundation. It covers the crucial decision-making process of the concept development, the eventual outcome, and the key learnings derived from the project.

2.1 Pre-Project

The theoretical foundation established during the pre-project is relevant to the current thesis, specifically emphasising product development, fundamental electronics, and basic mechanics [8]. This includes an understanding of programming concepts like syntax, control structures, and data types, as well as the operation of a closed-loop servo motor. It is assumed that readers have a foundational understanding of these concepts.

The project began with identifying issues and motivations and determining the need for a brand-new development. The product development was then based on the ULRIC and Eppinger methods utilised for concept evaluation. Five main concepts were selected for evaluation, i.e. the main concept, launch pad, frame, feeding mechanism, and microcontroller. The evaluated concepts are shown in Figure 4, where both (A1) and (A2) are based on two spinning wheels, which shoot the SC when it gets between.



	Concept/function	1	2
A	Main Concept		
B	Launch pad	Bart of the frame	
C	Frame		
D	Feeding mechanism		
E	Microcontroller	Arduino	Raspberry PI

TILT
Rotation
TILT
Rotation
TILT

Figure 4: Pre-project concept matrix [8].

After a discussion, concept (A2), in association with (B2), was chosen as it was considered advantageous in avoiding the tilting movement of the entire structure.

Furthermore was (C2) chosen despite being an expensive solution. It was selected for its flexibility in relocating components on aluminium profiles via T-nuts attachments. This selection provided a more practical alternative in contrast to the laborious process of drilling or 3D printing new plates for each respective configuration or alteration. Also, aluminium is stronger and was assumed to be used as a strong foundation.

The iris shutter (D1) was chosen as a concept for the feeding mechanism as it can be 3D printed. It was assumed to be relatively easy to adjust and program compared to a claw mechanism (D2).

After selecting the mechanisms and construction methods, the discussion centred on what should control the system. The choices were between Raspberry Pi (E1) and Arduino (E2). Both are available in nano size; the only difference is based on preference. Given that there was no experience with either and that the Raspberry Pi 4 model B was available, it was chosen for the prototyping. This model is a single-board computer and is relatively expensive. A single-board computer can have an operating system like a normal computer, costing roughly 400-1500 NOK depending on integrated memory and computing power [9].

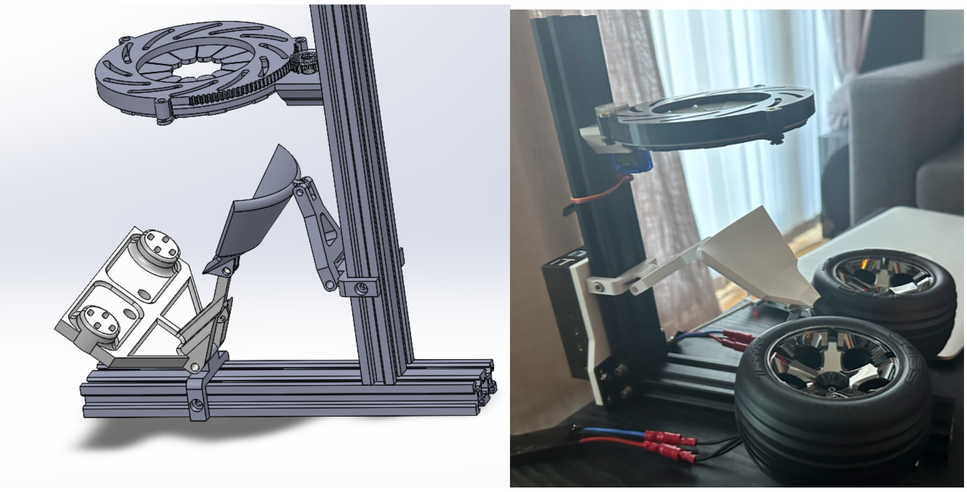
Regarding the wheels, the Traxx Alias wheels were chosen based on the assumption that the thick rubber layer will not damage the SC since it deforms when the SC comes under pressure between the wheels. Furthermore, two different Brushless DC electric motor (BLDC), an in-runner and an out-runner, were chosen to drive the Traxx wheels instead of a traditional DC motor.

Once the concepts were chosen, the actual development of the prototype began with preparations such as calculating the Electronic speed controller (ESC) size and writing the calibrating and speed-controlling code from a microcontroller. Thereafter, the SG90 servo motor was modified from 180° to a continuous servo motor due to a shortage of continuous servo motors. SolidWorks (SW) was used for Computer-aided design (CAD) all parts of the SCL.

Each sub-chapter describes the development process comprehensively, including various pitfalls and challenges encountered, how they were solved, and why the chosen solutions were selected with alternative solutions.

2.1.1 Result

Figure 5 shows the result of the development process. The report presents all the constructed parts and comprehensively explains their design, including the reasoning behind their specific configurations. Additionally, the results offer a financial breakdown of the projected cost of the final product.



(a) CAD assembly. (b) Assembly of the prototype.

Figure 5: Prototype assembled.

The study's conclusions revealed that the wheels did not operate at the desired speed due to heavy vibrations, and the feeding mechanism exhibited considerable slowness with excessive friction, leading to the overheating of the servo motors. Despite being a more costly alternative, the chosen solution offered the possibility of further development to include additional functionalities, hence justifying the higher investment.

2.1.2 Lessons Learned

This project provided valuable opportunities for learning new skills, such as using a microcontroller to program and control BLDC and digital servos. In addition, the theory of Pulse-Width Modulation (PWM) was a central theme in controlling these components. It was therefore taught what a microcontroller is and how it works, and how to set up a raspberry pi 4 model b, connect it to the network and control it via SSH[1] from different computers.

Furthermore, the control of servo motors was also new, where the theory of PWM became a central theme as both servos motors and ESC use PWM to control either the position for a servo motor or the speed for a BLDC via the ESC.

Considerable time was spent on planning and designing. This approach delayed the testing of the wheels, which were not examined until later in the project. Consequently, there was insufficient time to determine and procure alternative wheels that would have been functional.

Additionally, considerable effort was invested in the feeding mechanism and the 3D printing of this system. It was discovered that the resolution of the 3D printers available to students at Norges Teknisk-Naturvitenskapelige Universitet (NTNU) was inexact due to frequent usage and suboptimal maintenance. Consequently, the project encountered numerous failed test prints, and the only feasible solution was to limit the use of a 0.4 nozzle as standard.

2.2 Connection

The insights acquired from the pre-project will significantly influence the thesis, particularly concerning the choice of wheels and feeding mechanisms. The foundational knowledge established in the pre-project and introduced concepts remain relevant. As previously demonstrated, 3D printing represents a distinctive field of study and is given appropriate consideration in this thesis.

[1] SSH (Secure Shell) is a network protocol that provides a secure way to access and manage remote devices over an unsecured network. It allows users to log in to a remote system and execute commands securely, as all data transmitted between the two devices is encrypted [10, 11].

3 Theory

This chapter presents the theory relevant to further developing the SCL. It focuses on mechatronics, a combination of mechanical engineering, electronics, and systems design. Key components such as breadboards, Printed Circuit Board (PCB), automation, and microcontrollers are presented, highlighting hardware platforms like Raspberry Pi and Arduino, and servo motors.

This chapter also explains fundamental production methods such as soldering and 3D printing, examining CAD and Finite element analysis (FEA). Lastly, it introduces the Classical laminate theory (CLT) to facilitate an understanding of the behaviour of laminated composite materials. This overview provides the essential theoretical foundation for understanding the development of the SCL.

3.1 Mechatronics

Mechatronics is the multidisciplinary field combining mechanical- and electrical engineering and computer science. Mechatronic systems use mechanical components, sensors, and actuators, to control a machine, typically managed by computer software [12]. Microcontrollers serve as the central processing unit that manages and controls various components in the system. These compact integrated circuits contain processor, memory, and programmable input/output connectors, making them suitable for specific tasks in embedded systems. One of the essential features of microcontrollers is their ability to generate and control signals, such as PWM, which is crucial for controlling devices like LED, servo motors and electronic speed controllers in mechatronic systems.

3.1.1 Breadboards

Breadboards allow for the temporary connection of components without soldering, facilitating the testing and modification of circuits during the design phase. There are two main types of breadboards: solderless or plastic breadboard and solderable breadboard, perfboard or stripboard [13]. Figure 6 shows a breadboard for temporary connection on the right, and on the left is a solderable perfboard.

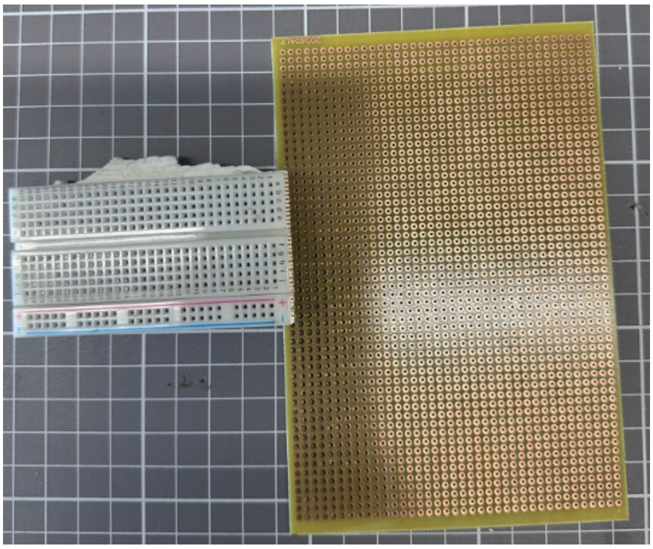


Figure 6: Example of breadboards.

The solderless breadboard, made of plastic, is the most common type. It consists of a grid of interconnected holes, where electronic components and jumper wires can be inserted to create circuits. These breadboards enable rapid prototyping and easy modifications, as components can be inserted and removed without damage [13].

Solderable breadboards are used when a permanent connection is desired. Components are soldered directly to the board, creating a robust and reliable connection. Solderable breadboards are typically used in the later stages of development when a circuit design has been finalised and needs to be implemented permanently [13].

In the context of mechatronics, breadboards are often used as a PCB during the prototyping and development phase. While breadboards facilitate the testing and debugging of electronic circuits, PCBs provide a permanent and reliable foundation for the final implementation of the electronic components in the mechatronic system.

3.1.2 Printed Circuit Boards

PCB are essential components in mechatronic systems, providing the physical foundation and electrical connections for various electronic components. A PCB connects components through conductive pathways, which are etched or printed onto non-conductive substrates [14]. The use of PCBs enables the efficient organisation and integration of electronic components, significantly reducing wiring complexity and improving overall system reliability.

In mechatronic systems, a PCB can integrate microcontrollers, sensors, actuators, and other components necessary to operate. The use of PCBs allows for the miniaturisation of the system and enables easy assembly, repair, and replacement of individual components. Moreover, a PCB can be designed to include specific features, such as mounting holes and connection pads, that accommodate the unique requirements of the mechatronic system. In developing prototypes, PCBs play a vital role in providing a platform for testing, debugging, and refining the electronic circuits and components of the mechatronic system [14].

3.1.3 Pulse Width Modulation

PWM is a technique for controlling the power delivered to electrical devices, such as motors and LEDs, by modulating the duty cycle of the electrical signal. The duty cycle refers to the percentage of time the signal is high during each PWM period, as shown in Figure 7. The duty cycle can be calculated to be $\frac{A}{T} = Dutycycle$. The pulse width is the duration of the high signal in each period (A) in Figure 7 often called the "on time". By varying the pulse width or duty cycle, the average power supplied to the device can be controlled, affecting the device's speed, position, or brightness [15].

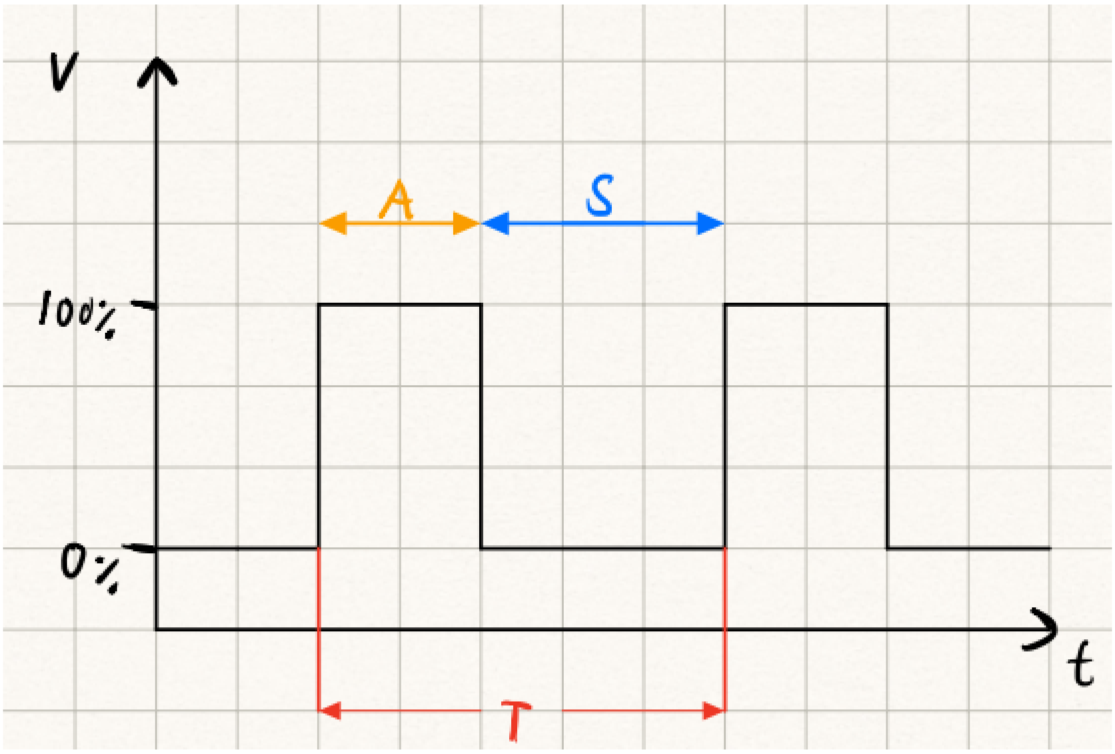


Figure 7: Example of a PWM signal.

PWM signals can be used to control the speed of a BLDC regulated by an ESC by manipulating the average voltage the motor receives. A PWM signal is a series of on and off pulses. When the signal is on, the full supply voltage is applied to the motor; when the signal is off, no voltage is applied. When the PWM signal is ’on’ for a longer duration (high-duty cycle), the motor receives more power, increasing its speed. Conversely, when the signal is ’on’ for a shorter duration (low duty cycle), the motor receives less power, reducing speed [15].

3.1.4 Automation

Automation refers to utilising technology and computer software to perform tasks that humans previously did. The main goals of automation are to simplify processes, reduce errors, increase efficiency, and save time and money [16]. Automation can free up human resources to focus on more complex tasks, allowing businesses to respond to marketplace changes. It can perform tasks faster and often more accurately than humans, increasing productivity and reducing costs. Automated systems are less likely to make any errors, reducing the likelihood of mistakes and improving quality control [16].

3.1.5 Microcontrollers

A microcontroller is a compact, integrated circuit that contains a processor, memory, and programmable input/output ports. Microcontrollers are typically used for specific tasks in embedded systems and can execute a single program repeatedly. They provide a simple and cost-effective solution for controlling and managing various components in a mechatronic system, such as sensors, actuators, and communication modules [17].

In the Pre-project, a Raspberry Pi 4 Model B was used as a microcontroller and will be used in this thesis for testing. This chapter will cover essential knowledge about the microcontrollers by Raspberry Pi and Arduino.

Raspberry Pi

Raspberry Pi 4 Model B is a single-board computer designed for various applications, ranging from basic computing to complex embedded systems. It was chosen primarily because it was desirable to try one, and the opportunity was there.

One of the most significant features of the Raspberry Pi 4 Model B is its General-purpose input/output (GPIO) interface, which allows for interaction with various electronic components and devices [18].

The GPIO library provides an interface for interacting with Raspberry Pi's GPIO pins, which can be configured as inputs or outputs for controlling various devices. The library lets users access and manipulate the GPIO pins using functions by enabling setting the mode of each GPIO pin as input, output, or alternate functions like I2C[2] or SPI[3], and reading or writing values to the pins [19]. The GPIO library also provides functions for generating PWM signals with a specified frequency, pulse width, and resolution, making it suitable for controlling servo motors and ESC.

Arduino

The core of each Arduino board is a microcontroller, which provides the processing power and programmable required for various tasks. Arduino boards come in different sizes and configurations, with varying memory, I/O pins, and processing capabilities, allowing users to select the most suitable board for any project.

Arduino IDE

Arduino Integrated Development Environment (IDE) is an open-source platform with various ”easy-to-use” hardware and software solutions. The IDE platform includes a variety of microcontroller boards designed for different needs and requirements. Arduino boards have gained widespread recognition for their simplicity, flexibility, and user-friendly development environment, making them an ideal choice for rapid prototyping and various applications, from simple DIY projects to complex embedded systems [20]. The IDE supports the C and C++ programming languages and includes built-in libraries and functions for easy access to the microcontroller's hardware features. Servo. h is, among other one of the libraries the IDE has built-in [20].

[2] I2C, or Inter-Integrated Circuit, is a serial communication protocol used for communication between multiple devices, typically a microcontroller and peripheral devices like sensors, memory chips, or displays [11].

[3] SPI, or Serial Peripheral Interface, is a synchronous serial communication protocol used for communication between microcontrollers and peripheral devices, such as sensors, memory chips, and display controllers. It is a full-duplex protocol, meaning data can be transmitted and received simultaneously [11].

Servo.h

The Servo.h library is an integral tool of the Arduino IDE platform. It is specifically designed to facilitate the control of servo motors and ESC with Arduino boards. Servo motors, a class of actuators renowned for their precise position control capabilities, are frequently employed in robotics, automation, and various mechatronic systems. ESCs, on the other hand, are used to control the speed of BLDC, which are commonly found in applications such as drones and remote-controlled vehicles that need accurate motor control [21].

By abstracting the complexities of generating the necessary PWM signals for controlling servo motors and ESC, the Servo.h library allows users to integrate these components into their Arduino projects efficiently. This library offers a set of high-level functions that enable control over the position or speed of servo motors and ESC without requiring users to generate PWM signals manually.

To use the Servo.h library, users must first include the library in their code and create an instance of the Servo class for each servo motor or ESC that requires control. Each instance represents an individual servo motor or ESC, and multiple devices can be controlled by creating multiple instances.

Thereafter, users must connect the appropriate pin on the Arduino board to the servo motor or ESC using the attach() function, which accepts the pin number as a parameter. Once connected, the servo motor's position or the ESCs speed can be controlled using the write() or writeMicroseconds() functions. The write() function accepts an angle (in degrees) as input, while the writeMicroseconds() function takes a pulse width (in microseconds) as input. Furthermore, the current position or pulse width of the servo motor or ESC can be determined using the read() and readMicroseconds() functions, respectively [21].

3.2 Production Technique

Production techniques is the process from raw materials such as metal, wood, or plastic into a final, ready-to-use product. This field involves various methods, including shaping, chip separation processes, and additive manufacturing. At NTNU, students are fortunate to have access to various advanced production tools, including milling machines, lathes, steel 3D printers, and laser cutters. Few people have access to these machines, which may cause this thesis irrelevant for some individuals and will not be used to develop the SCL. However, at NTNU, students are provided with labs equipped with mechatronics tools like soldering equipment, drills, pliers, and Prusa MK3 and MK3s 3D printers with either 0.6 or 0.4 mm nozzles.

3.2.1 Soldering

Soldering is a process used to join two or more metal components by melting a filler, known as solder, with a lower melting point than the joined components. Soldering is commonly used in electronics to make secure electrical connections between components and PCB [22].

The soldering process typically involves heating the components and solder using a soldering iron with a heated metal tip [23]. When the solder is melted, it flows between the components and forms a metallurgical bond upon cooling, creating a strong and electrically conductive joint. There are various types of solder, each with different compositions and melting points, suitable for different applications. The most common types of solder used in electronics are lead-based and lead-free solders, with lead-free solder becoming increasingly popular due to health and environmental concerns related to lead exposure [22].

The proper soldering technique is crucial to ensure the quality of the electrical connection and prevent damage to components or PCB. This includes selecting the appropriate solder and soldering iron tip, heating the components and solder to the correct temperature, and applying the correct amount of solder to create a smooth joint. Cleanliness of the soldering iron tip and surfaces to be soldered is also essential for optimal soldering results. In addition, using flux, a chemical cleaning agent, can improve the wetting properties of the solder and facilitate the formation of a reliable joint [22].

3.2.2 3D Printing

3D printing is the creation of three-dimensional products through additive manufacturing techniques, where the products are built layer by layer based on a digital 3D model. Various methods are available, each suited to specific material and strength requirements. The Fused deposition modeling (FDM) method, which uses plastic as the primary material, has been selected for this thesis due to its commercial availability and cost-effectiveness [24].

FDM works by extruding molten filament from a nozzle that follows a predetermined path corresponding to each structure layer. The material quickly solidifies after leaving the nozzle and bonding to the layer beneath. This process is repeated until the entire object has been printed, and the user then removes optional support structures. In the FDM process, filaments commonly contain thermoplastics or a blend of thermoplastics and other materials. These filaments come in different types, each with specific material properties suitable for various applications. When choosing the appropriate plastic type for printing, factors such as yield stress, elasticity, and elongation need to be considered [25]. FDM is a fast and inexpensive technique that is relatively easy to use, with readily available and affordable equipment. As such, it is particularly suited to rapid prototyping and proof of concept prototyping [26].

However, FDM has limitations, including lower resolution, accuracy and strength than alternative methods such as Selective Laser Sintering (SLS). As a result, parts produced through FDM are typically unsuitable for heavy load-bearing applications and may not meet strict tolerance requirements [25].

3.3 3D Printing Guidance

As an experienced 3D printer user owning a Creality ender 3 pro (E3P), it is evident that the Prusa printer NTNU supplies for their students are generally better than several other popular 3D printers. However, comparing the Prusa and E3P is particularly relevant as both printers share many geometric similarities. The Prusa MK3 costs roughly $1000 (NOK 14 000). It features linear rails, automatic bed levelling direct drive material feeding as shown in Figure 8 and a hot end that can print up to 300 degrees Celsius [27].

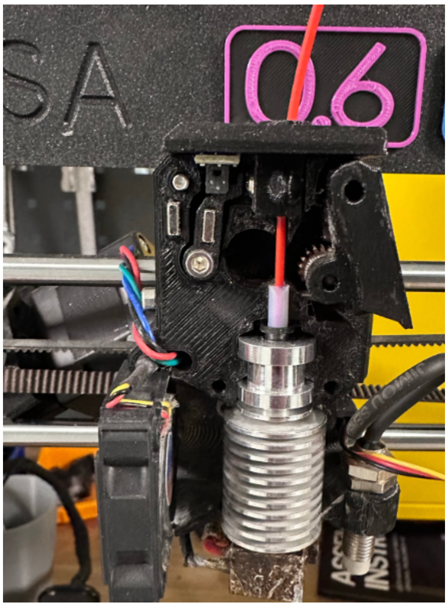
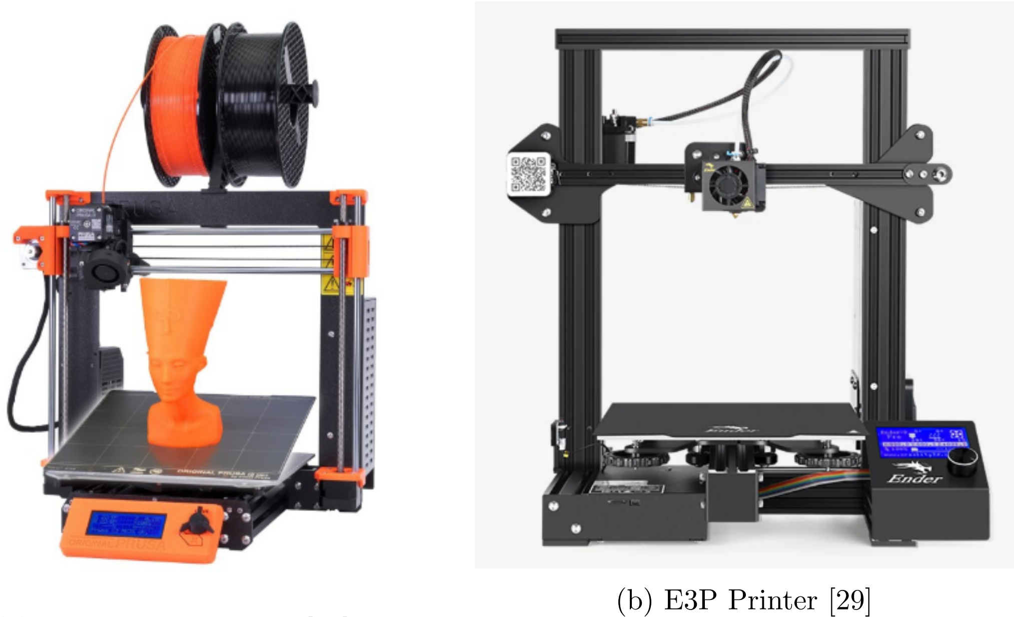


Figure 8: Example of the hot end and the direct extruder during an overhaul.

In contrast, the E3P is priced at $200 (NOK 2 500), uses bowden material feeding, does not possess automatic bed levelling nor linear rails, and instead, is equipped with adjustable wheels for all axis [28]. Technical comparisons are shown in Table 1 and visual Figure 9.



(a) Prusa MK3s printer [27]

(b) E3P Printer [29]

Figure 9: Visual comparison of the 3D printers.

Table 1: Comparison of Prusa and E3P.

Feature	Prusa	E3P
Price [$]	958	236
Print size (W*L*H) [mm]	210*210*250	220*220*250
Max hotbed temperature [°]	120	100
Max nozzle temperature [°]	300	260
Filament Compatibility	PLA, PETG, ASA, ABS, PC, CPE, PVA, PVB, HIPS, PP, Flex, Nylon, Carbon filled and Wood filled	PLA, ABS, TPU
Firmware	Marlin	Marlin

The print resolution owing to its auto bed levelling and linear rails, makes the Prusa superior compared to a standard E3P. On the other hand, the E3P has some advantages. For example, it is more affordable than the Prusa MK3, making it a better option for those on a budget. It is also a popular choice among many 3D printing enthusiasts due to its ease of use, build quality and lots of possibilities to upgrade. While it lacks automatic bed levelling and linear rails, it uses adjustable wheels for all axes, which can achieve similar results as a Prusa MK3 with proper calibration.

However, based on a E3P owner with extensive experience, the Prusa is faster, more reliable over numerous prints, and requires less maintenance than average 3D printers, making it a suitable choice for NTNU students to use as a prototype 3D printer.

It has, therefore, been acknowledged that not all individuals have access to Prusa printers. Consequently, all prototype components will be produced using PLA initially as it is both cheap and quick to print, and possible for anyone with a 3D printer without any upgrades. At last, none of the components will be exclusively sized to fit a Prusa MK3 print surface.

Slicer

Prusa slicer is used as the slicer software because it is already adapted to the printers and makes it easier through the thesis since there is no need to fine-tune settings. Prusa's software is adapted to Marlin firmware and offers pre-made profiles for several printers. Since the Marlin firmware is open source, there are also several profiles for 3rd party printers, e.g. Github [30]. Prusa slicer can thus be used for several printers such as E3P since it also uses Marlin firmware.

Pitfalls

There are several typical pitfalls when it comes to 3D printing for beginners. The most common is getting a good first layer, e-step calibration and z-wobbling. Table 2 contains links for help with the typical issues. All help is by all3dp.com.

Table 2: Common 3D printer troubleshooting

Problem	Link
First layer [31]	3D Printing First Layer Problems: How to Make It Perfect
E-step [32]	Extruder Calibration: How to Calibrate E-Steps
Z-wobbling [33]	Z Banding / Z Wobble: How to Prevent It

All3DP.com is an editorially independent publication that provides articles related to 3D printing and additive manufacturing. The site also offers news updates and in-depth reviews about 3D printing technologies, materials, software, and more [34].

In addition, All3DP.com offers practical advice, buyer's guides, and how-to information for people interested in 3D printing, from beginners to professionals. They also provide free 3D printing models and host a marketplace for 3D printing services [34].

3.4 CAD and FEA

CAD is a digital solution that simplifies the creation of technical drawings. Many CAD programs offer a visualisation feature that enables the creation of digital three-dimensional models. These programs typically allow for easy modification of details in the model by adjusting parameters within the software. While CAD is used in various fields, it is most prevalent in architecture, construction, and engineering [35].

FEA is a numerical analysis method used to simulate various scenarios to analyse constructions digitally. It requires a CAD model and a mathematical model based on selected loads, restraints, material properties, and analysis types [36, 37]. An element mesh must be created over the CAD model before the analysis can be carried out, consisting of many finite elements that can be of first or second order. Simplifying the CAD model can reduce compilation time by removing irrelevant design details that do not affect the strength. This numerical method is commonly used in engineering and design to optimise and validate product performance and in various fields of physics, such as solid mechanics, fluid dynamics, and electromagnetism. The analysis is solved numerically, and the results are interpreted manually by the engineer, depending on the construction requirements and the types of analysis chosen [37]. SW was used in this thesis for the analyses, and the theoretical concepts apply specifically to this program.

3.4.1 Static Analysis

In static analyses, it is assumed that the material is linear and the loads are static, i.e. stresses and strains are proportional, which is based on basic mechanics. However, this does not correspond to the reality where plastic deformation will occur, and maximum stresses are limited by strain or fracture. The static load is constant throughout the analysis in contrast to dynamic loads that vary over time. To perform a static analysis, restraints, material properties, loads, and the element mesh must be added to the CAD model.

From static analysis, four types of stresses can arise. The normal stress 1, bending stress 2, shear stress 3 or torsional stress 4

$$\sigma_n = \frac{F}{A}[\frac{N}{mm^2}] \tag{1}$$

$$\sigma_b = \frac{M}{W_x}[\frac{N}{mm^2}] \tag{2}$$

$$\tau_s = \frac{F}{A}[\frac{N}{mm^2}] \tag{3}$$

$$\tau_t = \frac{T}{W_p}[\frac{N}{mm^2}] \tag{4}$$

Von Mises criterion is further used to calculate the combined stress as shown in simplified form in the equation 5 [38].

$$\sigma_{vm} = \sqrt{(\sigma_s + \sigma_b)^2 + 3(\tau_s + \tau_t)^2} \tag{5}$$

Von Mises with three stress states: [37, s. 24]:

$$\sigma_{vm} = \sqrt{\frac{1}{2} \times [(\sigma_x - \sigma_y)^2 + (\sigma_y - \sigma_z)^2 + (\sigma_z - \sigma_x)^2]} \tag{6}$$

where:

σ_x - stretch or compression in the x-direction

τ_{xy} - shear stress in xy-plane

Von Mises with six stress states can be used:

$$\sigma_{vm} = \sqrt{\frac{1}{2} \times [(\sigma_x - \sigma_y)^2 + (\sigma_y - \sigma_z)^2 + (\sigma_z - \sigma_x)^2] + 3 \times (\tau_{xy^2} + \tau_{yz^2} + \tau_{zx^2})} \quad (7)$$

3.4.2 Topology Optimisation

Topology is a mathematics specialisation that focuses on the study of shapes. Unlike in geometry, shapes in topology are described by properties other than length and angle measurements. Optimisation and algebraic topology are the two primary fields where algebraic is particularly relevant for this thesis [39, 40].

Topological optimisation is a design study that helps the engineer choose the most optimally designed with regard to weight and strength. It can be used in the product development of new products and for the optimisation of an existing product. The mathematical model that will be used in this thesis and what SW uses is Solid Isotropic Material with Penalisation method (SIMP) [41].

SIMP-based topological optimisation determines the optimal material distribution to achieve maximum strength. Traditionally, element density in topological optimisation is defined as binary values, where 0 represents an empty element, and 1 represents a fully material-filled element. In SIMP, density is relative and can vary between a minimum selected density and 1, affecting the E-modulus of the element as density changes. This relationship is expressed by the formula 8. To minimise the contribution of elements with intermediate densities (between 0 and 1), a power law is used with a p-factor, typically set to 3 for optimal results [41].

$$E(\rho_e) = \rho_e^p E_0 \quad (8)$$

When the E-modulus is reduced, it also reduces the element's strength. This strength is calculated as shown in the following equation [41].

$$K_{SIMP(\rho)} = \sum_{e=1}^{N} [\rho_{min} + (1 - \rho_{min})\rho_e^p] K_e \quad (9)$$

Restraint, forces, material type, element mesh and the optimisation goal must be selected. Relevant to the task, the lowest weight and highest strength is desirable. The optimisation of the design is run in a loop until the result is most optimal with regard to the selected requirements. Then it is up to the engineer to evaluate and post-process the result.

The whole component structure is in solid materials when simulating in the context of SW simulations. In contrast, 3D printing entails a layer-by-layer fabrication process with structured infill. The successful construction of printed components depends upon considering two primary factors: the orientation of anticipated stress forces and the orientation of plastic layers deposited by the 3D printer. The designer must therefore determine the optimal orientation of the component within the printer, based on the principles of the CLT.

3.5 Classical Laminate Theory

A laminate is two or more lamina/layer/ply bonded together in structured directions. Figure 10 shows an example of a 3-layer lamina where layer 1 is 90 degrees to the x-axis, layers lie along the x-axis and become 0 degrees, and finally, the third layer lies at -45 degrees to the x-axis. The notation for the layer configuration would then be counted from the top layer to the bottom, such as [-45,0,90].

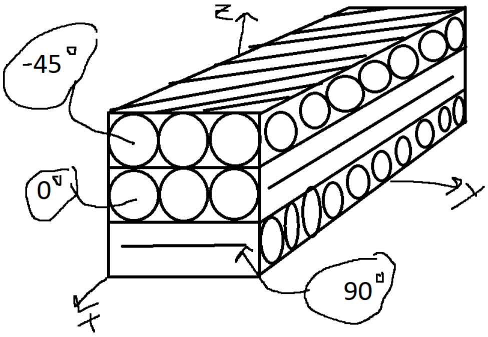


Figure 10: Laminate/ply direction example.

CLT provides a useful approach to analysing the coupling effect within composite laminates by predicting the stresses, displacements, and curvature that arise within them [42]. With this, is it possible to deduce the stiffness matrix based on the number of N layers, and calculate stresses and any displacements in the lamina. The calculation is not to be done for this thesis, but after several tests, it is proven that the perpendicular layer configuration :

[0,90]t = [-90,0]t = [-45,45]t = [45,-45]t

where t is $2 \times N$

The configurations are all the same, which means they go crisscross and are often the strongest configuration [25]. This can be changed in the 3D printer's slicer, but often a standard.

However, CLT are more relevant when parts are optimised and tested against failure. This will not be done in this thesis, but CLT will be considered when parts are constructed for 3D printing.

3.6 Prototyping

A prototype is a model of an individual component or product. Prototyping can be divided into two main groups: a physical product and a digital product. Prototypes are often used as physical evidence that the invention can be produced and/or function properly. Physical prototypes can also undergo various forms of real-world testing. On the other hand, there are digital prototypes [43].

A digital prototype in the form of a CAD drawing is often used in earlier stages. CAD, along with FEA, provides opportunities to simulate and evaluate prototypes on an ongoing basis to develop the product continuously [44], without having to construct a physical model that takes up a lot of time. Rapid prototyping is a development process, one of the methods discussed in the next chapter.

4 Method

This chapter presents the methods employed throughout this thesis, highlighting rapid prototyping. This process incorporates 3D modelling for design visualisation and strength assessment, facilitated through SW software. Complemented by 3D printing technology, these methodologies permit swift and efficient creation and analysis of prototypes, driving continuous advancements in the design process.

4.1 Rapid Prototyping

Rapid prototyping was used in product development, simplifying the creation of physical prototypes using CAD software and 3D printing technology. One of the goals of rapid prototyping is to set and validate product designs before large-scale manufacturing, reducing the risk of costly errors and expediting iterations. CAD software enabled easy adjustment and fine-tuning of digital models, allowing developers to examine prototypes, identify potential design issues, and implement necessary modifications [44]. Rapid prototyping in conjunction with CAD and 3D printing saved time and resources during product development and enhanced the final product's overall quality.

4.1.1 Static and Topology Simulation

Static simulation with SW was employed to study the components being made that could carry the weight or loads that occurred when the SCL was running. Some assumptions had to be made to do this, such as the material, load, and connection between reality and a virtual test.

Boundary conditions

The material used in the simulation was ABS, with the material properties for a given material shown in Table 3. The material was chosen as it most closely resembles the behaviour and strengths of other plastics such as PLA and PetG.

A weight test was performed on the pre-project parts, yielding approximately 1.6kg. Thus, all static weight in this thesis would take that as a starting point considering the CLT, parts in SW will be solid elements, while in reality, they will be laminate with patterns of cavities. Thus, the static total weight was set to 5kg, which became the main static load condition for the simulations to compensate for the fact that the parts were not solid and that PLA was used.

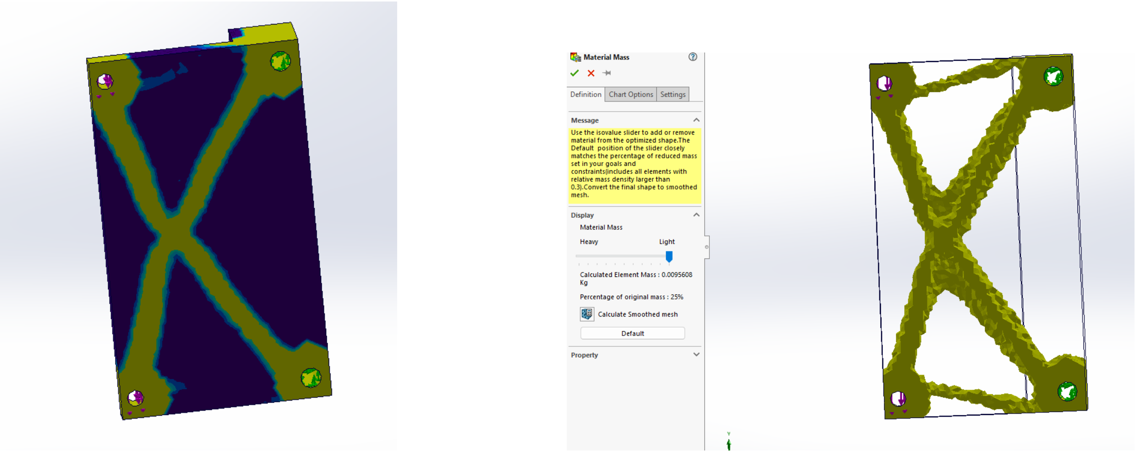
Table 3: SW ABS material properties.

What	value	Unit
Elastic Modulus	2000	MPa
Poisson's Ratio	0.394	
Mass Density	1020	Kg/m^3
Tensile strength	30	MPa

Furthermore, it has not been determined how the fixtures are placed and were placed under each component since all components have different fixtures.

The goal for the topology optimisation simulation was established as achieving the "best stiffness-to-weight ratio", which was set at 50% , and the subsequent results were evaluated based on this target.

Topology optimisation was extensively utilised when components lacked structure or designs were created. This procedure was reiterated with assumed load cases for topology optimisation, facilitating the determination of optimal design. Figure 11 provides an example of a part undergoing this process. Figure 11a show a component where the purple and yellow areas combined indicate the original design. Figure 11b presents the outcome when all theoretically extra material has been removed.



Use the isovalue slider to add or remove material from the optimized shape.The Default position of the slider closely matches the percentage of reduced mass set in your goals and constraints(includes all elements with relative mass density larger than 0.3).Convert the final shape to smoothed mesh.

Calculated Element Mass : 0.0095608 Kg

Percentage of original mass : 25%

(a) Initial design with marked material for retention.

(b) Outcome following removal of theoretically unneeded material.

Figure 11: Example of the topology optimisation.

4.1.2 3D printing

3D printing technology as a rapid prototyping method was utilised to test and evaluate the design of the individual parts of the SCL. The printers used were the Prusa MK3 and MK3s, equipped with a 0.4 or 0.6 nozzle. Table 4 shows the printer settings used as a standard setting from Prusa Slicer.

Table 4: Standard print settings

Setting	MK3S/MK3 0.4	MK3s/MK 0.6	unit
Nozzle diameter	0.4	0.6	mm
Nozzle temp	215	215	$C°$
Bed temp	60	60	$C°$
Layer height	0.2	0.2	mm
First layer height	0.2	0.2	mm
Fill density	15	15	%
Fill pattern	Gyroid	Gyroid	
Top fill pattern	Monotonic	Monotonic	
Bottom fill pattern	Monotonic	Monotonic	

3D printing technology facilitated the efficient production of physical prototypes in various designs, enabling evaluation and modification as needed. There were numerous advantages to employing 3D printing technology as a rapid prototyping method. First, it aided in identifying and addressing potential design flaws. Second, it permitted the examination of the product's functionality and aesthetics in a significant manner, which was challenging to accomplish through the digital version.

Additionally, using the Prusa MK3 and MK3s printers with a 0.4 or 0.6 nozzle, and standard settings helped to achieve high-quality prints—the 0.2 layer height balanced print speed and resolution.

4.2 Video Recording and Slow-Motion Filming

Considering the thesis's aim for individuals without access to specialised equipment, the iPhone 14 Pro, with its slow-motion capability, was selected as the primary tool for video recording. This choice was motivated by the widespread availability of smartphones with slow-motion features, making it a more accessible option for capturing detailed footage.

The iPhone 14 Pro, equipped with a high-quality camera, was utilised to observe and document the functioning of the feeding mechanism and the behaviour of the SCs as it interacts with the feeding mechanism and wheels.

Due to the absence of a tripod, all captures were filmed by hand. Although this approach may introduce slight variations in camera stability, efforts were made to maintain consistency during filming to minimise potential distortions or artefacts that could affect the analysis.

The captured footage, filmed at a frame rate of 240 frames per second (fps), was subsequently reviewed and analysed manually. Each recording was carefully observed, allowing for the identification of key aspects related to the feeding mechanism's operation and the SCs behaviour upon leaving the tube and making contact with the rotating wheels. Rapid prototyping methodologies were employed to make adjustments based on the observations made during the manual review process.

5 Choice of Wheels

The focus of this chapter lies in the identification of appropriate wheels for a SCL. The Traxx Alias wheels employed in the pre-project did not yield satisfactory results [8]. The study in this chapter centres on two types of wheels: 5-spoke and VOLANTE, evaluating their capacity to achieve the desired velocity without inciting excessive vibrations or causing damage to the SC.

This chapter initiates a calculation of the requisite RPM for the wheels. The presentation and discussion of the velocity test results for both wheels. Issues experienced with the 5-spoke wheels are further elaborated upon, leading to the proposal of VOLANTE wheels as a potential replacement.

5.1 Pre-Calculation

The initial velocity of a SC during a serve in badminton can vary depending on several factors, such as the serve technique used by the player, and the type of SC being used. However, on average, a SC can have an initial velocity of around 100 to 150km/h during a serve [45].

With a baseline velocity of 100 to 150km/h, the objective was to find appropriate wheels capable of achieving a minimum 100km/h velocity. Due to the rotational nature of wheels, the correlation between RPM and linear velocity must be established. The calculation of the linear velocity to angular velocity can be achieved by equations 10,11.

$$\omega = \frac{v}{r}[\frac{\frac{m}{s}}{m}] \tag{10}$$

r is the wheel's radius in meters, and v is the minimum initial velocity in meters per second. This gives the angular velocity ω [rad/s] and can be used to find the required Rounds Per Minute:

$$RPM = \frac{60 \times \omega}{2 \times \pi} = \frac{30\omega}{\pi} \tag{11}$$

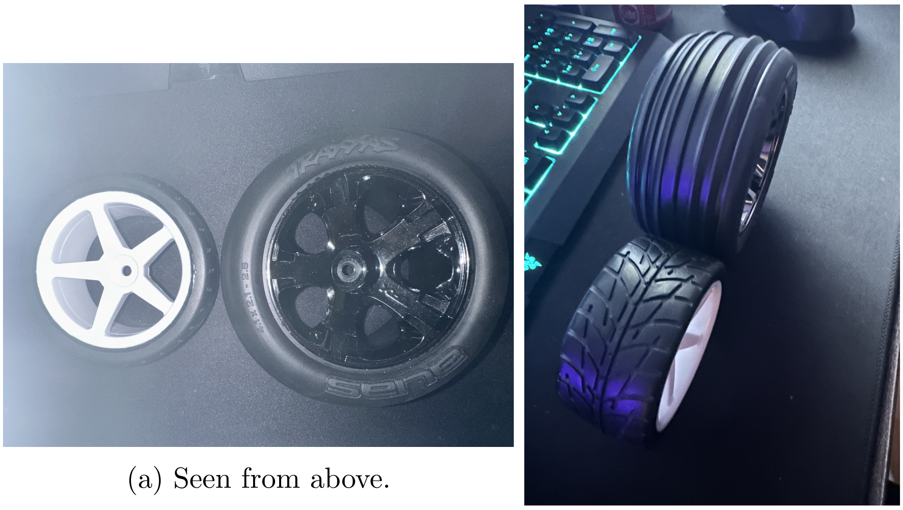
5.2 Velocity Test

A velocity test determines if the wheel can reach the target RPM without producing significant vibrations destabilising the entire SCL. Additionally, identifying the wheel's maximum RPM is crucial to confirm if the target RPM is possible and to determine the remaining capacity the wheels can achieve.

The velocity test process involves using a Raspberry Pi 4 model B and the code in appendix A from the pre-project, connected to the BLDC Xing 2604 1650 KV with the Dualsky 45Amp ESC. A tachometer is used to measure RPM, with the set pulse width and RPM documented in a list for comparative analysis.

5.3 5-Spoke Wheels

A new set of wheels was brought from Løten Rcshop. Figure 12 compares the newly acquired wheels to the left and the former by Traxx at the right. The new wheels are not associated with any specific brand and are named "5-spoke" wheels. The purchase of the 5-spoke wheels is based on the assumption that "street wheels" featuring thinner layers of rubber can endure higher velocity without substantial deformation of the rubber, thus preventing imbalances.



(a) Seen from above.

(b) Seen from the side.

Figure 12: Comparison of wheels.

The 5-spoke wheels have a diameter of 80mm and a width of 37mm. They feature a centre hole with a diameter of 5mm and a 12mm hexagonal hole, matching the dimensions of the Traxx wheels.

After adapting the motor mount to fit the 5-spoke wheel and calculating the theoretical RPM using equations 11 and 10, as shown in equation 12, the 5-spoke wheels were thereafter mounted and connected for velocity testing.

$$RPM = \frac{30 \times \frac{\frac{100}{3.6}}{0.040}}{\pi} \approx 6631[RPM] \tag{12}$$

During the initial velocity test with the 5-spoke wheels, the starting velocity was not set to 0. The consequence of this oversight is shown in Figure 13, which illustrates the shattering of the motor plate that tilts. Additionally, the tilt ear sustained damage and broke, as well as one 5-spoke wheel.

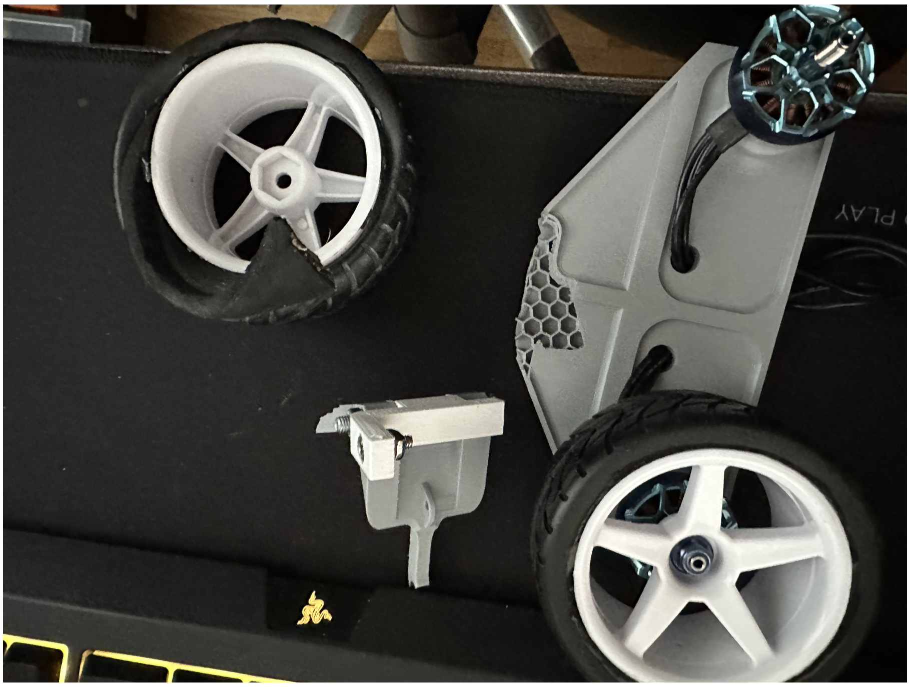
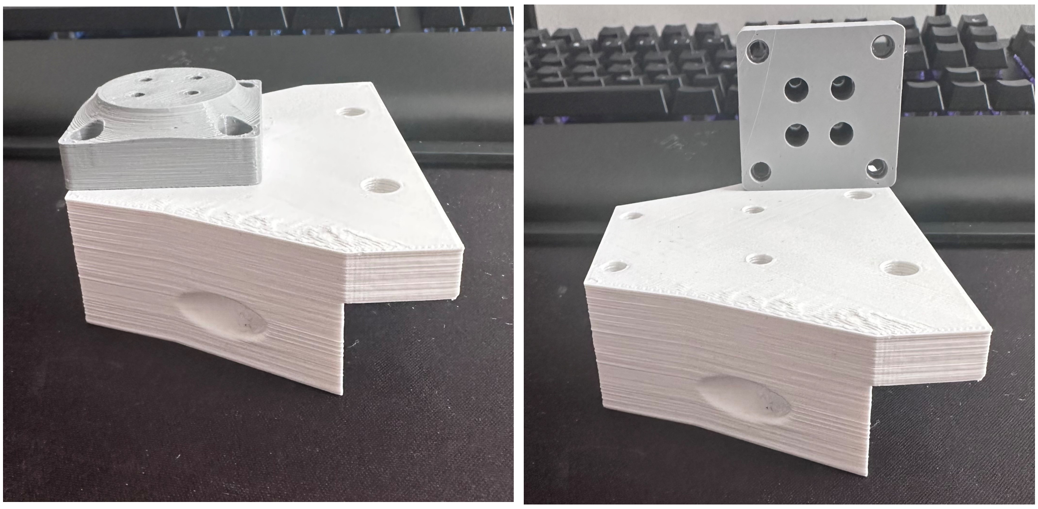


Figure 13: 5-spoke wheel torn.

In consideration of Health, Safety and Environment (HSE), the design involving a tilting plate was abandoned due to the potential for additional accidents. An internal review was performed, and it was decided that two robust motor blocks would be constructed instead for testing and further development. These motor blocks are to be attached to the aluminium profile, with a motor holder on top that is secured onto the motor block. Figure 14 displays the white motor block and the grey motor holder.

Moreover, 25 M5x8mm screws and 25 M5 T-nuts were procured from Rat Rig to standardise all screw components. In the interest of rapid prototyping and efficiency, the testing motor mount will be divided into two parts. This approach ensures that only the motor block needs to be reprinted in case of modifications, rather than the entire assembly, simplifying the development process.



(a) Assemble position.

(b) Parts overview.

Figure 14: New motor block with motor holder.

Although one wheel sustained damage, an effort was made to fix it using glue. Despite the repair, the wheel had a small notch at the initial tear location. Nevertheless, the undamaged wheels were examined before ordering new ones.

Before the test, both BLDC and their respective ESC were labelled as Motor System A (MSA) and Motor System B (MSB) to differentiate them in case of any disparities and were then operated without the wheels attached. This procedure was done to find the minimum and maximum pulse width, as presented in Table 5. The intact 5-spoke wheel was installed and tested on MSA. The outcome is shown in Table 6.

Table 5: min. and max. Pulse width.

	MSA	MSB
min	1110	1200
max	1650	1800

Table 6: Result of the velocity test with the 5-spoke wheel.

Pulse width	Velocity [RPM]
1100	0
1110	3271
1120	3924
1130	4479
1140	4559
1150	4804
1160	5214
1170	5860
1180	6027

Vibrations became noticeable from 4000 RPM onwards. As illustrated in Figure 15, an attempt was made to attenuate some of these vibrations by adapting the motor mounts to incorporate springs. The springs were mounted directly on the screws connecting the motor mount and the holder. An alternative solution for vibration suppression could involve using rubber dampers [46].

Furthermore, two aluminium 12mm hex nuts were procured. These are compatible with all wheels featuring 12mm hex holes, as shown in Figure 16. In RC cars, 12mm hex nuts serve as adaptors that connect the wheel to the axle or drive shaft. These hexagonal nuts fit into corresponding hexagonal-shaped recesses in the wheels, ensuring a secure and stable connection between the wheel and the axle. These hex nuts have a 5mm untreated hole, simplifying and improving the wheel centring process.

However, excessive vibrations continued despite the vibration suppression measures and improved centring with the hex nut. Consequently, the 5-spoke wheels were set aside, and a new variety of wheels from VOLANTE was ordered.

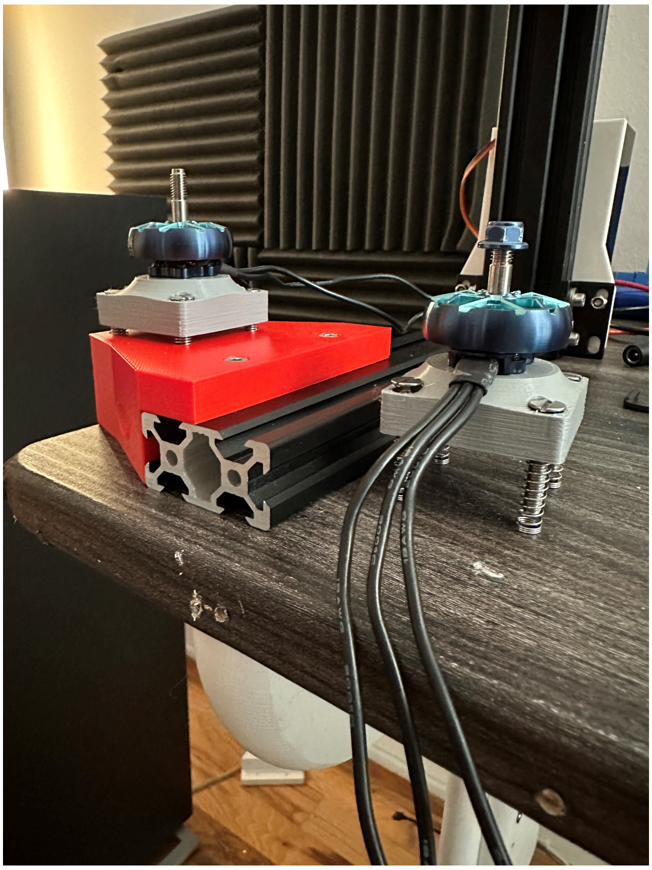
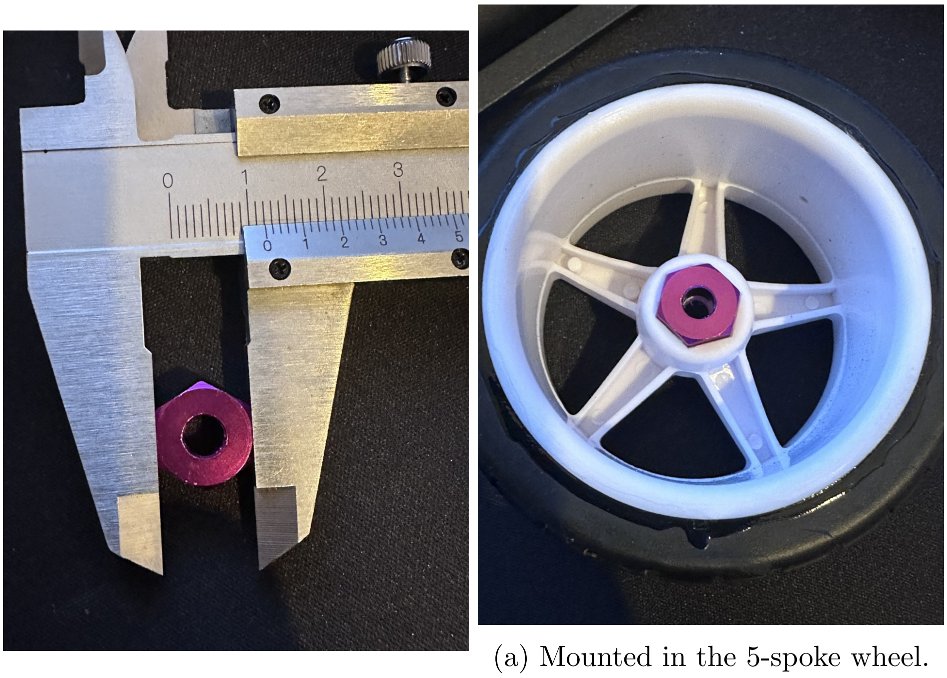


Figure 15: Engine Mount with springs.

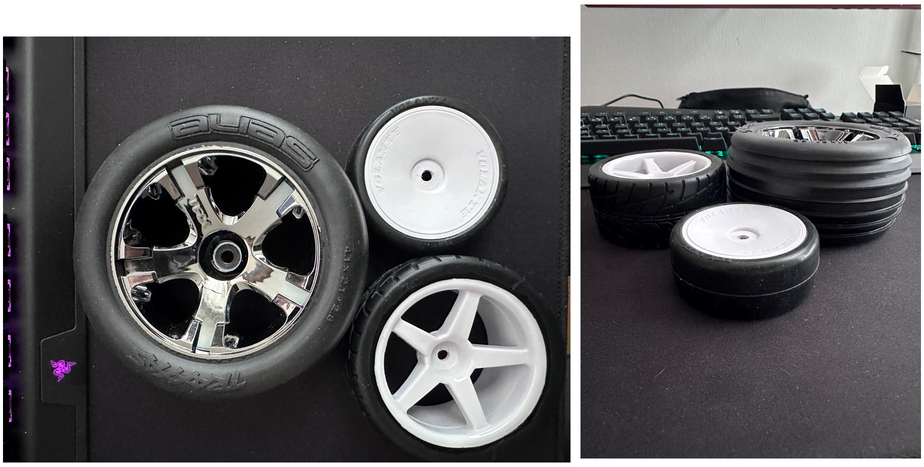


(a) Mounted in the 5-spoke wheel.

Figure 16: Hex nut for wheel centring.

5.4 VOLANTE wheels

Figure 17 compares the VOLANTE, 5-spoke, and Traxx, previously examined in the pre-project.



(a) Seen from above, With the Volante wheel at the top right.

(b) Seen from the side with the VOLANTE Wheel in the center front.

Figure 17: Comparison of all wheels.

The VOLANTE wheels have a diameter of 64mm and a width of 24mm. The centre hole measures 5mm in diameter and features a 12mm hex hole, consistent with the dimensions of both previously tested wheels. After calculating the theoretical desired velocity, as demonstrated in equation 13, and adapting the motor mount to accommodate the VOLANTE wheels, they were installed and connected in preparation for velocity testing. Figure 18 presents the VOLANTE wheels mounted.

$$RPM = \frac{30 \times \frac{\frac{100}{3.6}}{0.032}}{\pi} \approx 8289[RPM] \tag{13}$$

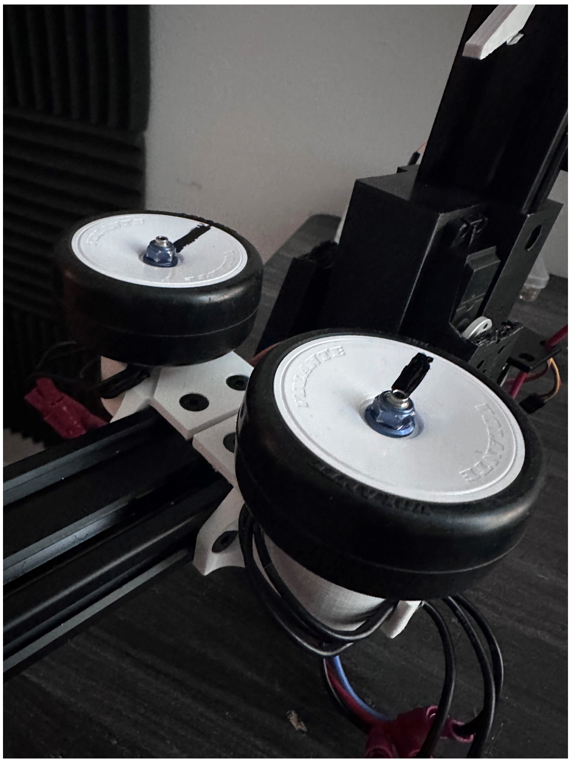


Figure 18: VOLANTE wheels mounted on the new motor mounts.

During the testing process, one motor system was operated at a time, with the velocity incremented by five pulse widths per measurement until the wheel and structure vibrated to such an extent that the SCL began to move. Table 7 shows the results for MSA and MSB. When the assembly started to move physically, the pulse width value at which it vibrated was deemed too high, and the previous value velocity and corresponding pulse width were considered the maximum.

Table 7: Result of the velocity test with the VOLANTE wheels.

pulse width	ESA [RPM]	pulse width	ESB [RPM]
1200	4182	1110	4035
1205	4800	1115	4731
1210	5430	1120	5608
1215	6189	1125	6196
1220	6708	1130	6708
1225	7193	1135	7201
1230	7862	1140	7849
1235	8355	1145	8301
1240	8759	1150	8706
1245	9296	1155	9183

As shown in Table 7, the VOLANTE wheels achieve a velocity of approximately 9200 RPM, corresponding to a tangential velocity of 111 km/h. This result is obtained by reversing equations 11 and 10, as demonstrated in equation 14. Since this velocity surpasses the minimum requirement of 100 km/h, the VOLANTE wheels are suitable for continued use in the thesis. A more comprehensive discussion of the motor mounts is provided in the Construction Review 10.

$$\frac{RPM \times r \times \pi}{30} = v = \frac{9200 \times 0.032 \times \pi}{30} \approx 30.8m/s \approx 111km/h \tag{14}$$

It is noteworthy that the velocities and pulse width values in Tables 7 and 6 do not correspond, even though both ESCs originate from the same supplier and share identical specifications, and both BLDCs are of the same type.

The RPM of a BLDC motor can be influenced by various factors, including the load on the motor. As a motor experiences increased load, its velocity may decline, since it must exert more effort to maintain the same velocity. In such circumstances, the ESC might need to raise the average voltage delivered to the motor to preserve the desired velocity. However, the PWM signal generated by the microcontroller is adjusted based on the target motor velocity, rather than the load. Consequently, the PWM signal produced does not directly alter the motor's output RPM due to the load. Instead, it indirectly impacts the RPM by controlling the average voltage applied to the motor, affecting the motor's velocity under different loads [47, 48, 49]. Some ESCs may feature dynamic timing adjustment or governor mode to modify the PWM signal based on motor load. However, these features are often not typically employed in hobby ESC.

6 Pan Tilt Mechanism

This chapter undertakes the development of the pan/tilt mechanism. Starting with why and inspiration, then an explanation about the design that has been done for each part of the mechanism with key functions.

6.1 Inspiration

From chapter 5, the BLDC foundations are changed from being able to tilt to standing firmly on the aluminium profile. This means that the concept (A2) and (B2) is no longer valid from the pre-project concept mentioned in chapter 2. Based on the changes, creating a new type and design of the pan/tilt mechanisms is necessary to achieve the objective (1.3) of Versatility; Automating the pan/tilt mechanism is desirable, requiring either a servo or a stepper motor.

To find a new method to create a pan/tilt mechanism, thingiverse[4] was used as inspiration.

On Thingiverse, ”pan tilt camera” was searched, and (hyperlink here) was found as an inspiration and 3d printed as shown in Figure 19. The 3D print was simple and provided the basis and inspiration needed to design a custom fit for the SCL.

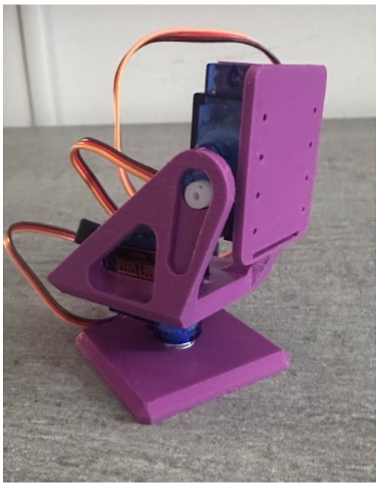


Figure 19: Pan/tilt mechanism from Thingiverse [50].

Since the pan/tilt the entire SCL, an SG90 motor will be relatively weak. Buying or designing gear transmissions may be an option if the SG90 motors are desired.

[4] Thingiverse is a website and online community for sharing user-generated digital designs, mainly for 3D printing. The platform allows users to upload, share and discover 3D printable models and collaborate with others on design projects.

This was not done as it seems to cost more than buying a larger servo motor. Thus, two 20kg digital servos were obtained for the thesis.

The 20kg servo motors are relatively expensive in Norway, but at a relatively lower price if waiting for shipping is an option, as shown in Table 8. With the servo motors chosen, CAD process could begin.

Table 8: Price of different servos.

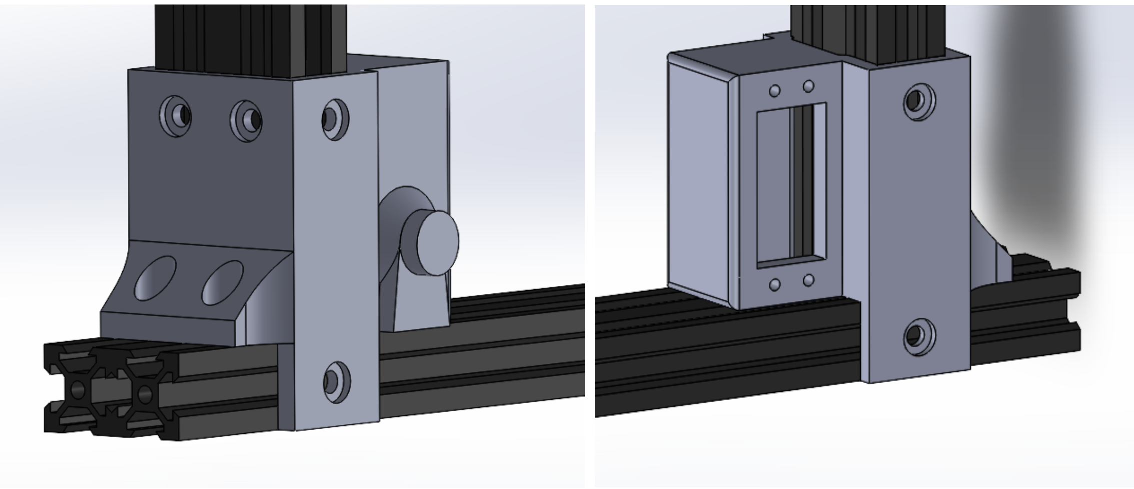
From	Price [NOK]	pcs
Elefun	895 [51]	1
Elefun	995 [52]	1
Amazon	191 [53]	1
Amazon	380 [54]	2

The concept was to take inspiration from the three components shown in Figure 19 and modify them to create three components fit for the SCL: a Fixed Attachment that securely mounts the tilting servo to the aluminium profiles, a Tilt Attachment that rotates and supports the Fixed Attachment and the rotating servo, and a Base Attachment that can be mounted on the ground or a tripod and serves as the foundation for the other components.

All three components have the common task of carrying the SCL; therefore, each component is tested virtually on SW to ensure that the components are not undermentioned. Each component will thus be presented in the following sub-chapter with approximately the same layout where they are presented, the task they have and simulation conditions with the results of the static simulation.

6.2 Fixed Attachment

The fixed attachment is designed to hold together both aluminium profiles while simultaneously holding the servo motor, as illustrated in Figure 20. This enables movement of the entire structure along the vertical aluminium profile, making it easier to adjust the position.



(a) Rear view.

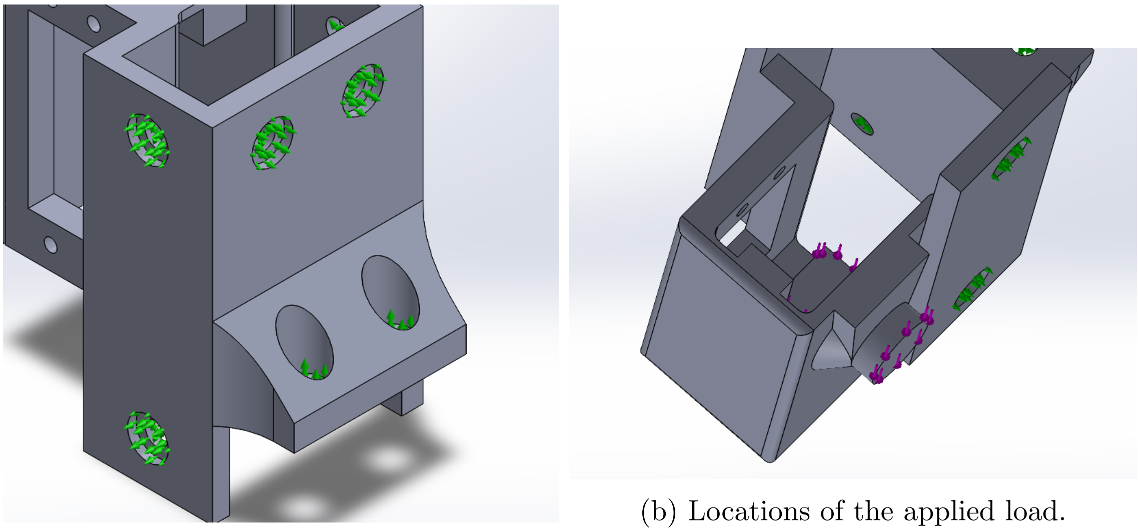
(b) Front view and servo mount.

Figure 20: CAD of the fixed attachment.

It is also intended that the servomotor can be pressed in and possibly screwed if preferred. At the same time, the cable can be pulled through the slot in one of the aluminium profiles. Furthermore, the shaft for the servo motor is lined up towards the centre of the shaft shown in Figure 20a (a) and carries the fixed attachment on the opposite end.

6.2.1 Simulation

A static simulation has been carried out where the choice of material and loads was set as mentioned in chapter 4. The fixtures are set where the screws are placed as shown in Figure 21a. Furthermore, the total load distributed on the surface where the servo motor is supposed to rest and at the tip of the axle is shown in Figure 20a as shown in Figure 21b.



(a) Fixtures.

(b) Locations of the applied load.

Figure 21: Simulation conditions for the fixed attachment.

Figure 22 displays the simulation results, revealing insignificant stresses. This suggests that the construction may be properly dimensioned.

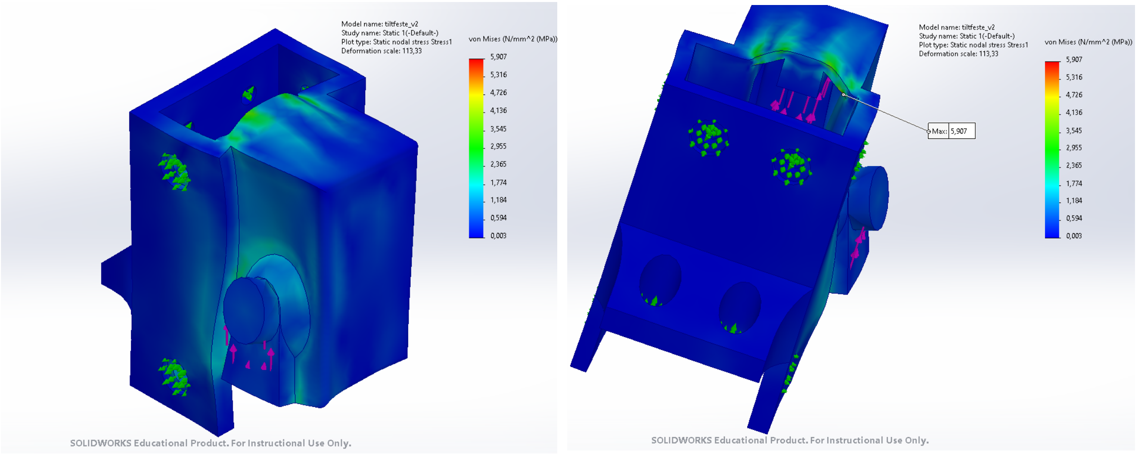
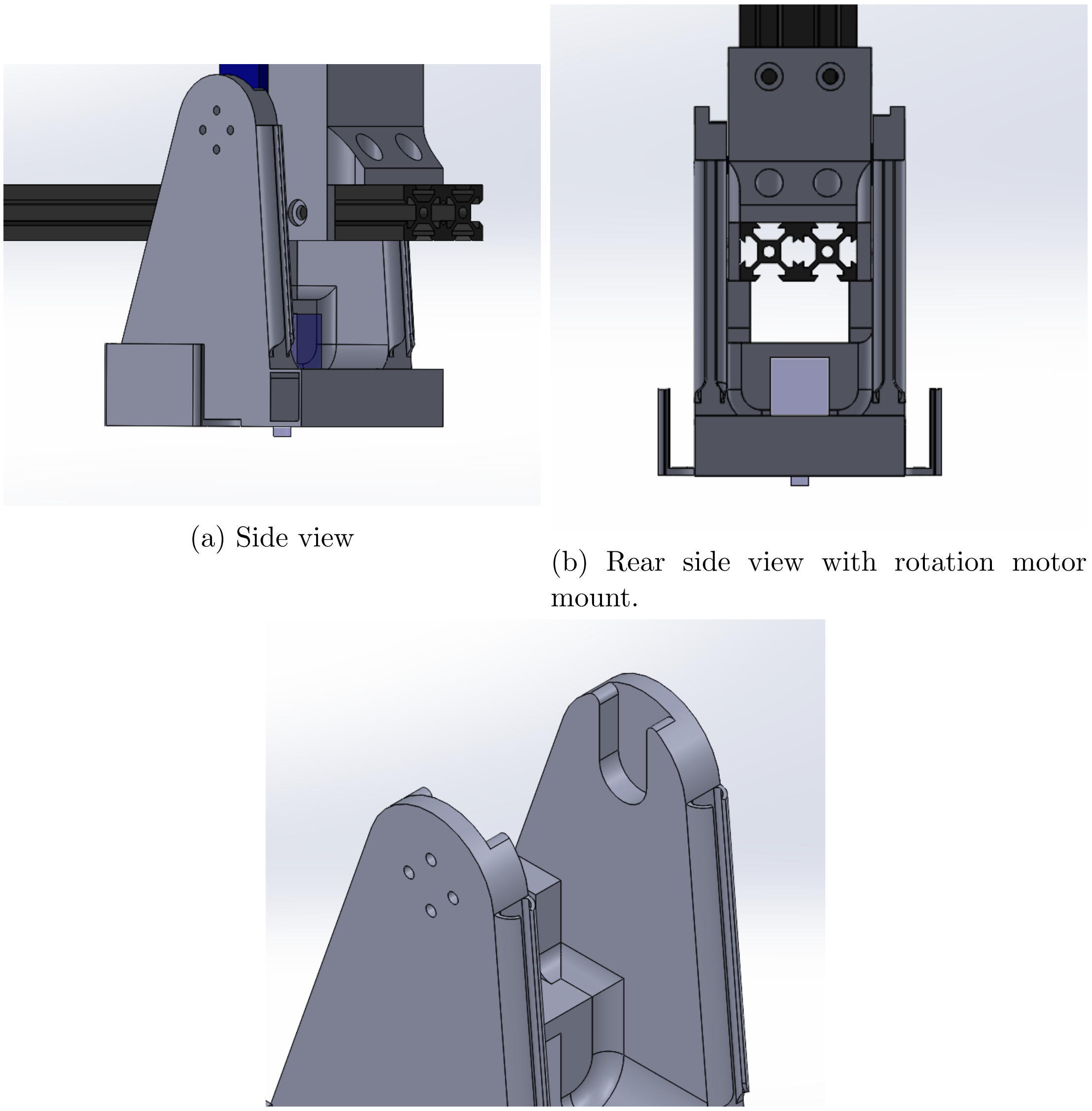

Model name: tiltfeste_v2
Study name: Static 1(-Default-)
Plot type: Static nodal stress Stress1
Deformation scale: 113,33
von Mises (N/mm^2 (MPa))
5,907
5,316
4,726
4,136
3,545
2,955
2,365
1,774
1,184
0,594
0,003
SOLIDWORKS Educational Product. For Instructional Use Only.
Model name: tiltfeste_v2
Study name: Static 1(-Default-)
Plot type: Static nodal stress Stress1
Deformation scale: 113,33
Max: 5,907
von Mises (N/mm^2 (MPa))
5,907
5,316
4,726
4,136
3,545
2,955
2,365
1,774
1,184
0,594
0,003
SOLIDWORKS Educational Product. For Instructional Use Only.

Figure 22: Simulation result for the fixed attachment.

6.3 Tilt Attachment

Figure 23 shows the tilt attachment at different angles. At the top are two holes where the fixed attachment is to be mounted, and at the bottom is a square hole with four screw holes for the rotation motor.



(a) Side view

(b) Rear side view with rotation motor mount.

(c) Rear side view with both servo mounts.

Figure 23: CAD of the tilt attachment.

The centre of the holes from the fixed attachment servo motor flang is lined up to the centre of the shaft of the rotary motor as illustrated in Figure 24.

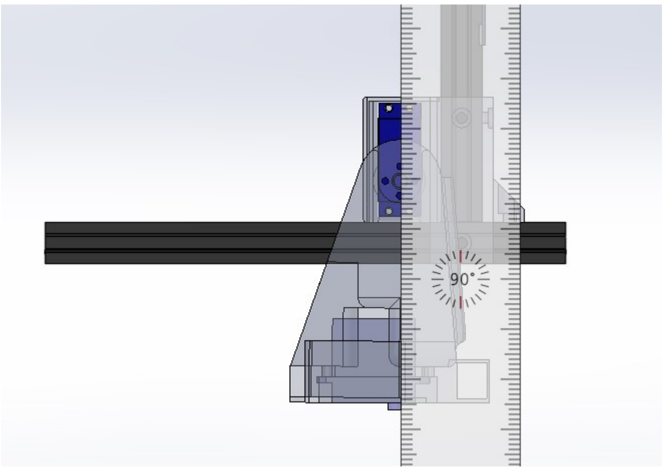


Figure 24: Shaft lineup.

This has been done with regard to the stresses. By eliminating displacements from the force directions shown in Figure 24, this plane will only have a pressure tension. Thus the combined stresses from Von mises criteria from equation 5 will become $\sigma_{vm} = \sqrt{\sigma_{pressure}^2}$

Furthermore, it is also designed so that the motor for rotation rests against a cross beam which can be shown in Figure 25. This has been done to distribute the carrying capacity for the original flange of the motor.

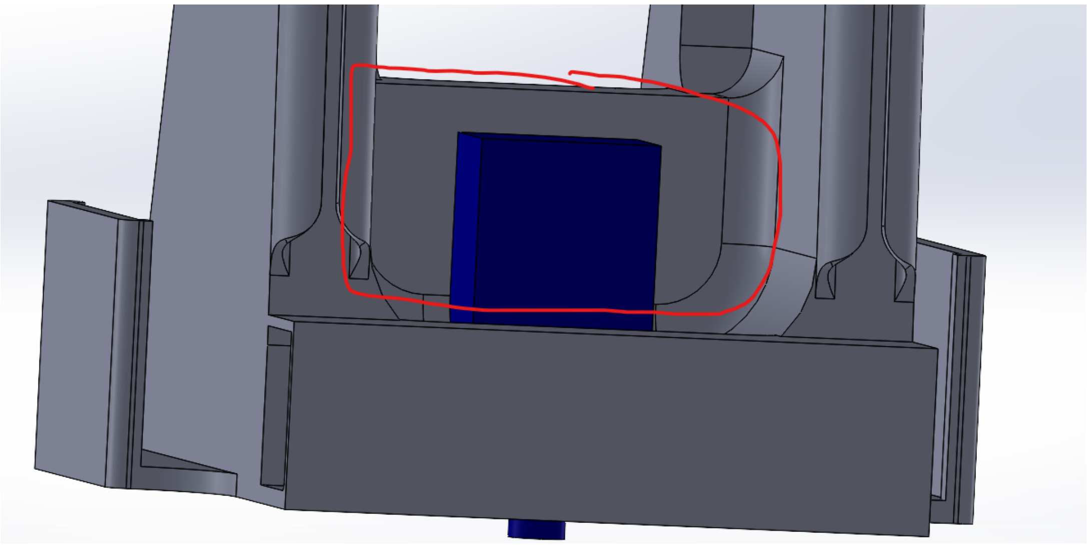
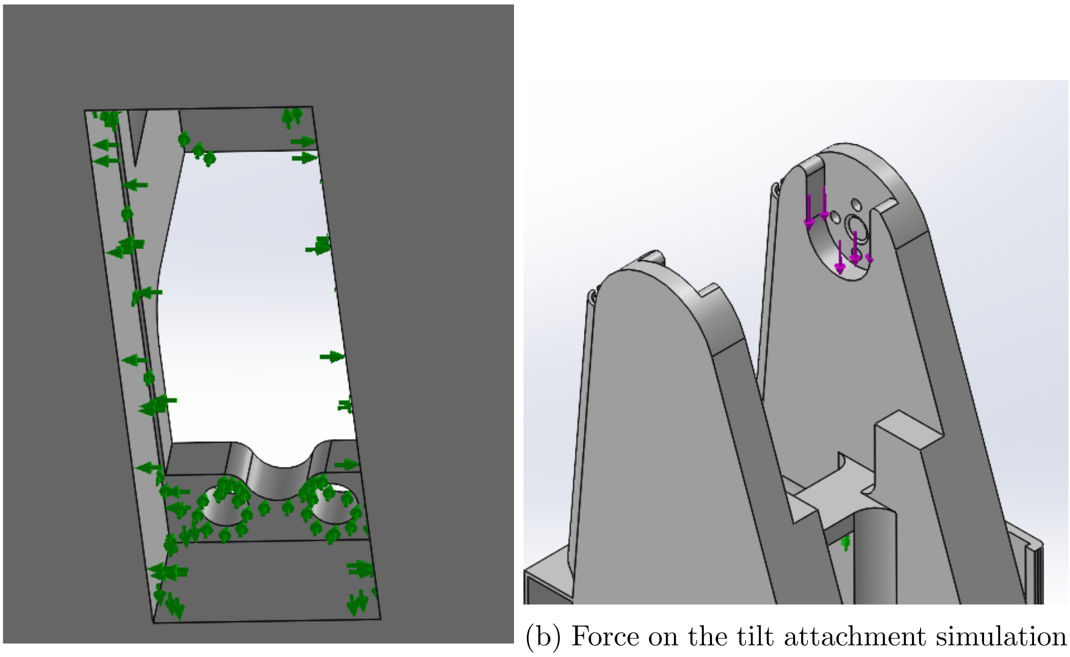


Figure 25: Cross beam.

6.3.1 Simulation

A simulation has been carried out where the choice of material and loads was set as mentioned in chapter 4. The fixtures were set perpendicular to the cross beam and



(a) Fixtures on the tilt attachment simulation

(b) Force on the tilt attachment simulation

Figure 26: Simulation conditions for the tilt attachment.

to the flange where it can be screwed down and directly against the walls the motor rests/works against, as shown in Figure 26. The static load is vertical on both the fixed attachment flanges at the top, as shown in Figure 26.

The result from the simulation shows the greatest stresses occur in the contact surface where the fixed attachment itself will rest, and half is distributed on the crossbeam as from the colour map in Figure 27. The stresses are relatively low, as 0.5MPa << (30MPa the tensile strength for ABS 4). Based on this, topology optimisation was carried out to see where and if any material could be removed.

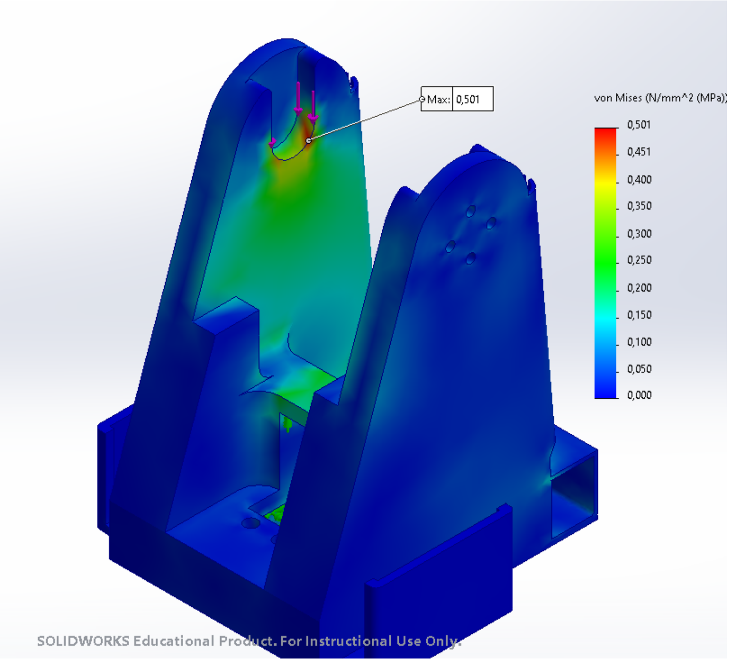


Figure 27: Static stress test.

Before the topological study, areas that were restrained, force applied, and areas used for the purpose were set as restrictions that material shall not be removed from them. Some additional details have not yet been explained, but they will be presented in the construction review 10. The result from the topology study is shown in Figure 28, where 38% of the mass can be reduced while detailed areas remain unchanged.

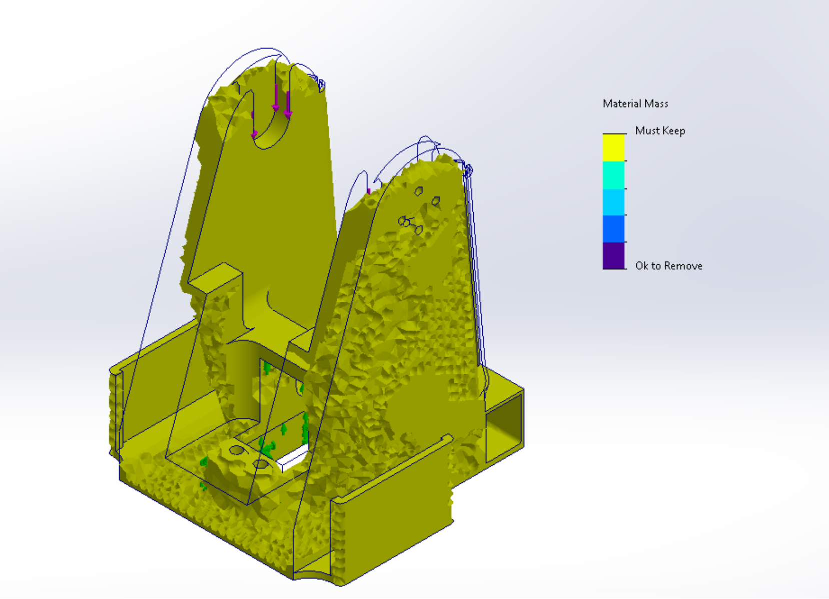


Figure 28: Topology study result.

With regard to the CLT, not all the material will be removed as this is a solid simulation, and the prototype will be in plastic layers with pattern infill. As shown in Figure 29, some changes were made where only the mass in the front has been removed. It should be noted that when this is done and all material is removed, it is assumed by the software that these are the only forces that work. But in reality, other forces may arise that have not been considered. Thus, only minimal material has been removed so that it is not optimal but robustly designed.

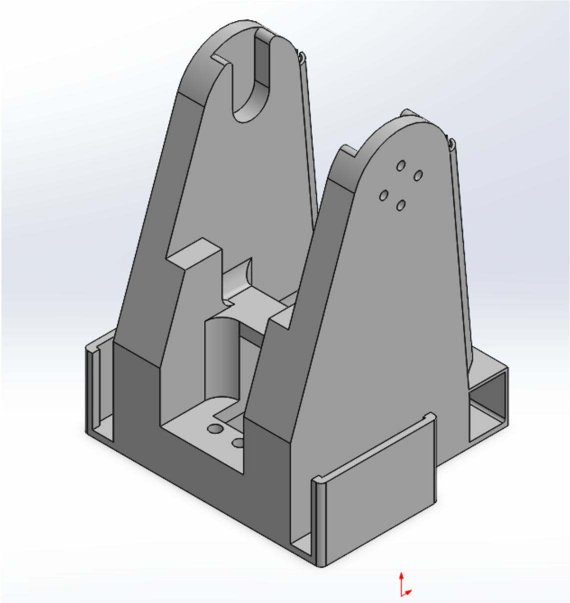
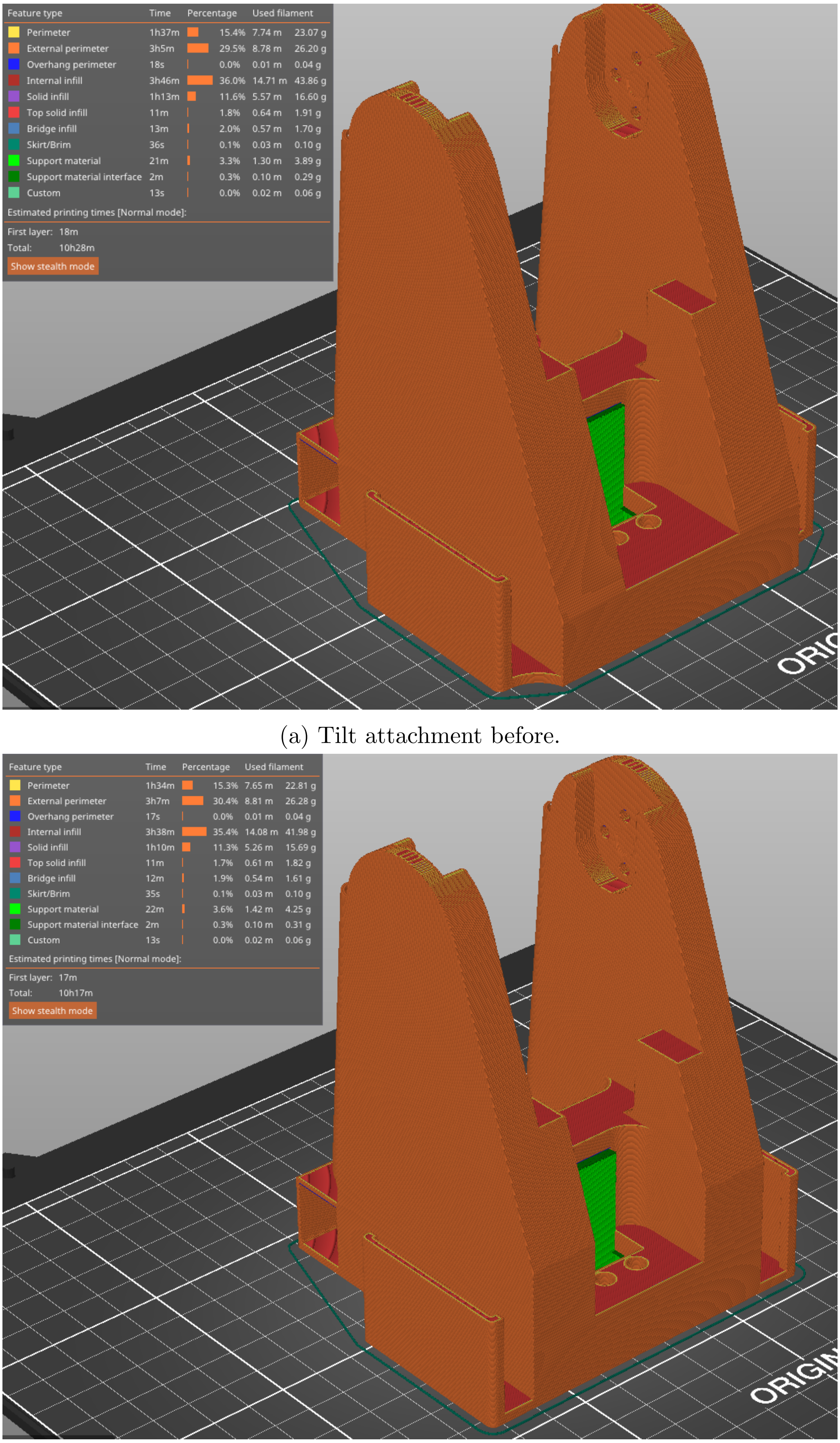


Figure 29: Material removal after topology study.

This has not changed any mechanical properties significantly apart from the centre of gravity. The advantage is that it will take approximately 11 minutes faster to print the part with a 0.4 nozzle, or 7 minutes if a 0.6 mm nozzle is used. Both numbers come from the slicer settings from tab 4, and an example of before and after time in the slicer is shown in Figure 30.



Feature type	Time	Percentage	Used filament	
🟨 Perimeter	1h37m	15.4%	7.74 m	23.07 g
🟧 External perimeter	3h5m	29.5%	8.78 m	26.20 g
🟦 Overhang perimeter	18s	0.0%	0.01 m	0.04 g
🟥 Internal infill	3h46m	36.0%	14.71 m	43.86 g
🟪 Solid infill	1h13m	11.6%	5.57 m	16.60 g
🟥 Top solid infill	11m	1.8%	0.64 m	1.91 g
🟦 Bridge infill	13m	2.0%	0.57 m	1.70 g
🟩 Skirt/Brim	36s	0.1%	0.03 m	0.10 g
🟩 Support material	21m	3.3%	1.30 m	3.89 g
🟩 Support material interface	2m	0.3%	0.10 m	0.29 g
🟩 Custom	13s	0.0%	0.02 m	0.06 g

Estimated printing times [Normal mode]:

First layer: 18m

Total: 10h28m

Show stealth mode

(a) Tilt attachment before.

Feature type	Time	Percentage	Used filament	
🟨 Perimeter	1h34m	15.3%	7.65 m	22.81 g
🟧 External perimeter	3h7m	30.4%	8.81 m	26.28 g
🟦 Overhang perimeter	17s	0.0%	0.01 m	0.04 g
🟥 Internal infill	3h38m	35.4%	14.08 m	41.98 g
🟪 Solid infill	1h10m	11.3%	5.26 m	15.69 g
🟥 Top solid infill	11m	1.7%	0.61 m	1.82 g
🟦 Bridge infill	12m	1.9%	0.54 m	1.61 g
🟩 Skirt/Brim	35s	0.1%	0.03 m	0.10 g
🟩 Support material	22m	3.6%	1.42 m	4.25 g
🟩 Support material interface	2m	0.3%	0.10 m	0.31 g
🟩 Custom	13s	0.0%	0.02 m	0.06 g

Estimated printing times [Normal mode]:

First layer: 17m

Total: 10h17m

Show stealth mode

(b) Tilt attachment after.

Figure 30: Tilt attachment printing comparison.

Following the modification, static simulations were performed on the tilt attachment. The results remained unchanged compared to the initial result.

6.4 Base Attachment

The base attachment is designed to be fixed onto ”something” and remain attached to the tilt attachment servo motor that rotates the SCL. Two configurations have been made, one that can be fixed onto a table with five screws, see Figure 31, and one that is adapted for tripod attachments, see Figure 32. These base attachments have a flange for mounting the servo motor from the tilt attachment and a slot for a standard 51105 trust bearing in common.

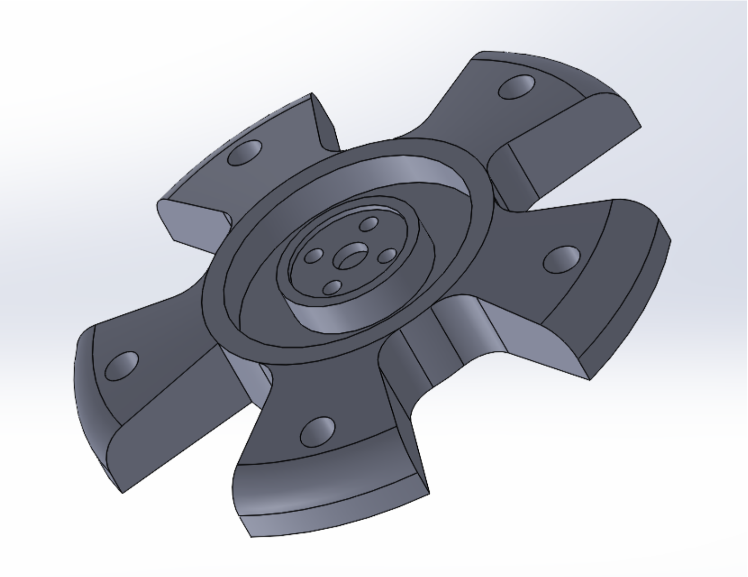


Figure 31: 5 spoke base attachment.

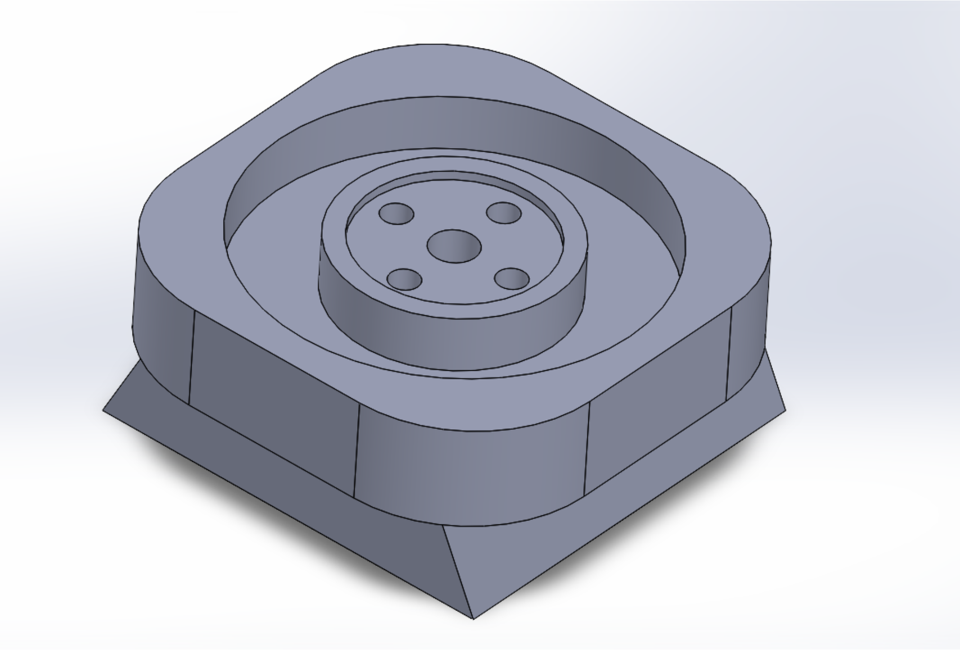


Figure 32: Tripod base attachment.

The principal purpose of using a trust bearing is to relieve the servo motor's load. Without a trust bearing, movements that cause axial deflection may break the motor's axle. Thus, a trust bearing will take up axial forces to prevent this. Furthermore, a trust bearing will also distribute the stresses that arise so that the base attachment receives an evenly distributed pressure instead of concentrated straight from the axle.

An alternative approach could involve adjusting the height of the top surface of the base attachment to establish contact with the tilt attachment. This modification would provide support during movements and alleviate strain on the motor. However, it is important to note that employing plastic against plastic may introduce friction. To mitigate this, applying lubricant becomes essential, although it may be undesirable due to its greasy nature. Trust bearings are relatively affordable, as shown in Table 9 and 51105 trust ball bearings were used in this prototype.

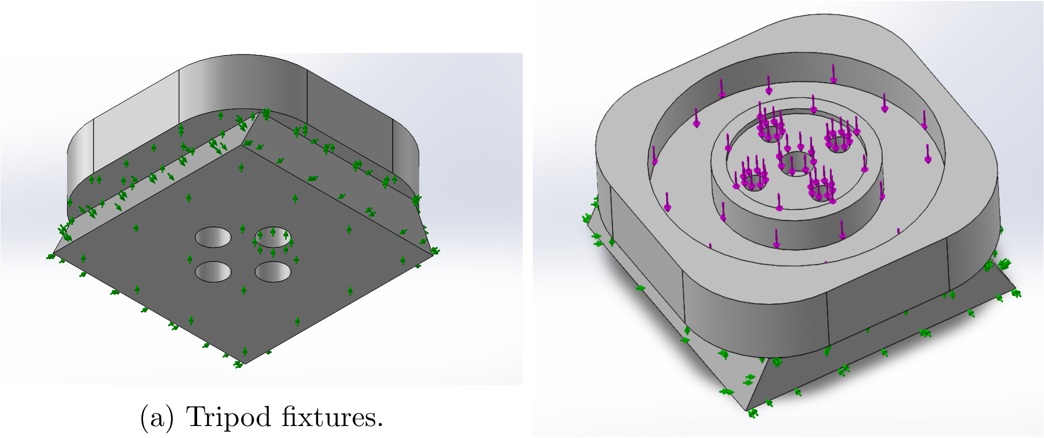
Table 9: Trust bearings price.

From	Price [NOK]	pcs
Verktøy.no	279 [55]	1
Amazon.com	75 [56]	1
aliexpress.com	18 [57]	1

A 51105 thrust bearing by Svenska Kullagerfabriken (SKF) was purchased for the thesis. The specifications say it withstands a dynamic load of 18.2KN, which corresponds to approximately 1855kg, and a maximum speed of 9000 RPM, which is considerably more than expected for the SCL [58].

6.4.1 Simulation

A simulation for the tripod mount was carried out where the choice of material and loads was set, as mentioned in chapter 4. It is assumed that the full load is distributed evenly over the thrust-bearing and flange surface, as shown in Figure 33 (b). It was also assumed that the fixture is perpendicular to the underside surfaces, ensuring that it remains immobilised when securely locked onto the tripod, as shown in Figure 33 (a).

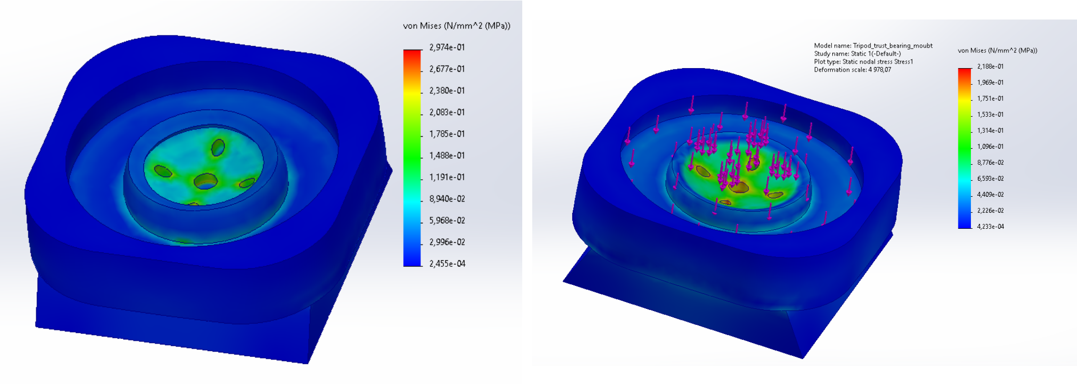


(a) Tripod fixtures.

(b) Tripod load placement.

Figure 33: Conditions before simulation.

The initial simulation revealed minimal stress levels. However, upon printing the test model, a configuration simplification was identified. This modification involved removing the countersinking of the flange hole, allowing for the tripod base attachment to be printed without the need for support. Consequently, a new simulation was conducted to assess the updated design. The results from the initial simulation are depicted in Figure 34a, while the outcomes of the new simulation can be seen in Figure 34b



(a) Simulation result from the first configuration.

(b) Simulation result from the second configuration.

Figure 34: Simulation results of the tripod base attachment.

The construction review 10 provides a complete description of the final results and configurations of the fixed, tilt, and base attachments.

7 Feeding Mechanism

This chapter presents the progress in developing a new feeding mechanism. The chapter begins by introducing the existing feeding mechanism from the pre-project, highlighting its associated problems and limitations. Subsequently, the chapter focuses on developing the new feeding mechanism, covering its conceptualisation, design, prototyping, and testing stages.

7.1 Iris Feeding Mechanism

Two iris shutters were developed in the pre-project, as illustrated in Figure 35. However, friction was observed between the blade surfaces during the testing phase. This friction caused the SG90 motor or equivalent size to be insufficient in power, resulting in an unreliable feeding mechanism.

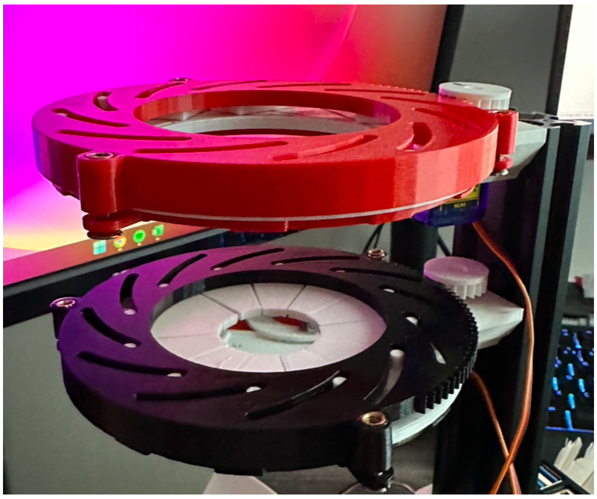


Figure 35: The iris shutter feeding mechanism.

Aligned with the goal stated in section 1.4, an additional criterion for the SCL was defined – the launcher should be easy to assemble. Ease of assembly is essential for potential users to swiftly and accurately assemble the machine without the need for technical knowledge or special tools. A user-friendly assembly process enhances the device's appeal and boosts its practicality.

Recognising the constraints inherent in the existing design of the iris shutter, it became clear that alternate methods for a feeding mechanism were necessary. Such alternative approaches could deliver improved performance in terms of speed, reliability, and ease of assembly. This realisation paved the way for an exploration and refinement process detailed throughout this thesis.

7.2 Arm Prototype Design

Figure 36 illustrates the basic operating principle of the new feeding mechanism, which consists of a tube (shown in dashed lines) and an arm attached to a servo motor. The feed mechanism consists of two steps, each serving distinct functions.

In the first step, known as the starting position, the bottom part of the mechanism is responsible for holding all SCs to prevent them from falling through. Simultaneously, the top part remains free, allowing the tube to be refilled with SCs.

In the second step, the top part of the mechanism takes on a different role. It retains the SCs not intended for feeding, effectively preventing their release. At the same time, it acts as a guider for the feed SC at the bottom, ensuring controlled movement through the feeding process. This two-step operation enables precise control over the SCLs flow.

However, the servo motor is expected to experience greater stress than the iris shutter, depending on the number of SCs it needs to hold and its weight. This aspect will be carefully examined through physical testing.

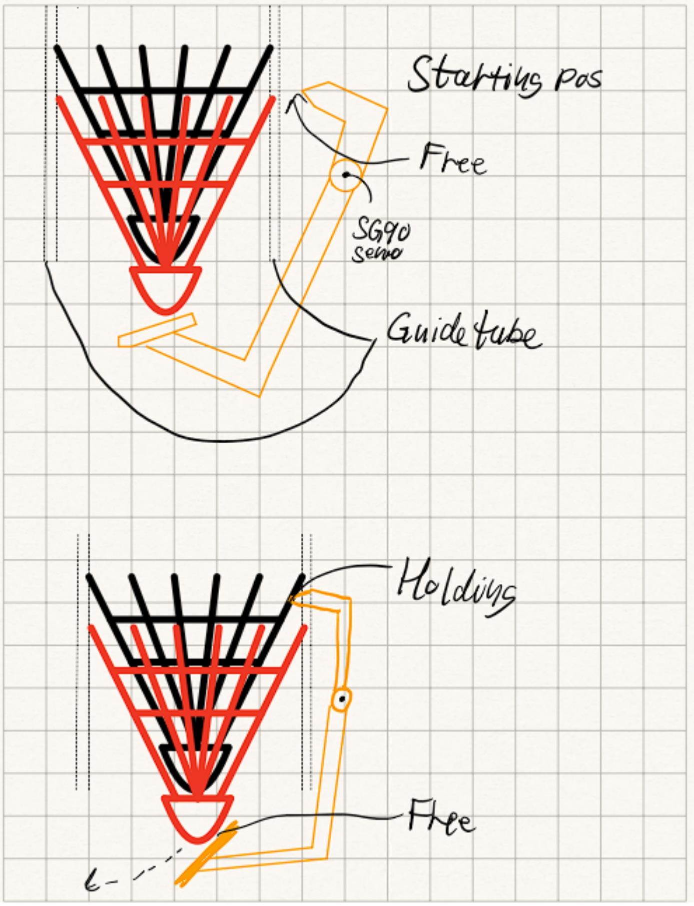


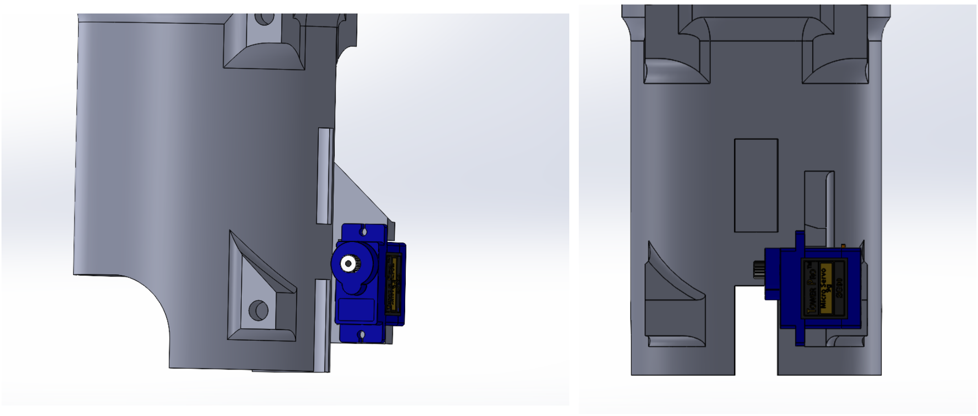
Figure 36: Two steps single servo feeding principle.

The necessary geometries of the tube, including the motor mount, were determined, and the design of the feeding arm commenced. The CAD software SW facilitated precise calculations, adjustments, and the visualisation of the parts and their interactions in a three-dimensional environment, aiding in developing an effective feeding mechanism.

The design process began with the tube, for which the inner diameter was set to 67mm following the standards that define the diameter range of SC feathers (58 to 68mm) [1, 2]. This was done to ensure compatibility with the dimensions of the SCs.

Next, the tube design focused on positioning the SG90 motor shaft at the centre of the tube and securing it to the tube to optimise space utilisation, as depicted in Figure 37. This arrangement allowed for efficient motor integration within the feeding mechanism.

Additionally, the distance from the SG90 axle centre to the bottom of the tube was determined to be 30 mm. This measurement considered that the maximum allowable size of the cork head, according to standards, is 26 mm [1, 2]. An additional 4 mm was included to give the arm's length space on the "holding" side.



(a) CAD seen from above, With the SG90 engine mounted.

(b) CAD seen from side with the SG90 engine in front.

Figure 37: CAD of the tube design.

After all necessary geometry of the tube and motor mount was determined, the design of the feeding arm began. All calculations and adjustments were performed in SW, where the parts could be moved, and their interactions visualised in three dimensions, as demonstrated in Figure 38.

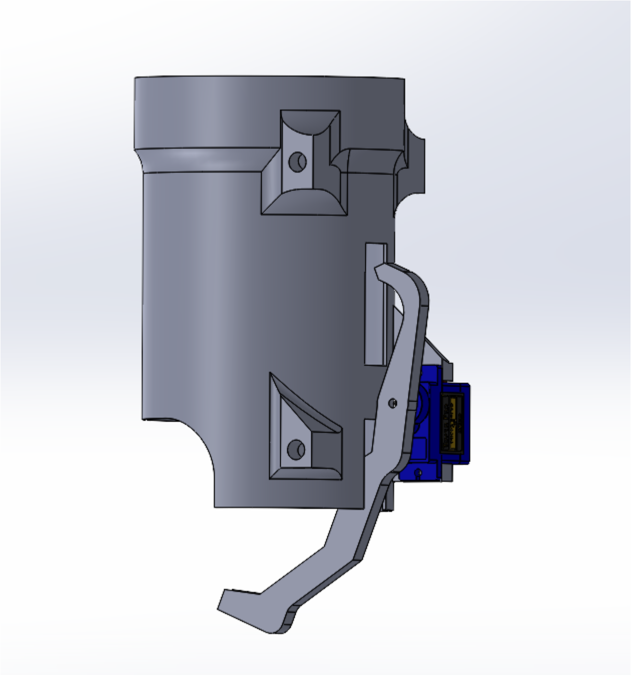


Figure 38: 3D prediction test.

7.2.1 Arm V.1

After finishing the first design, the feeding arm and tube were fabricated using 3D printing technology and tested on the SCL.

During the initial trial, it became evident that the arm had a narrow gap at the bottom, allowing SCs to fall between the arm and the tube. A component called the "Shuttle Retainer" was developed to address this issue. The shuttle retainer is a shell-shaped structure resembling a hollow semicircle. The shuttle retainer was designed to be attached to the arm at the bottom. Its primary function is to hold and guide the SCs during feeding.

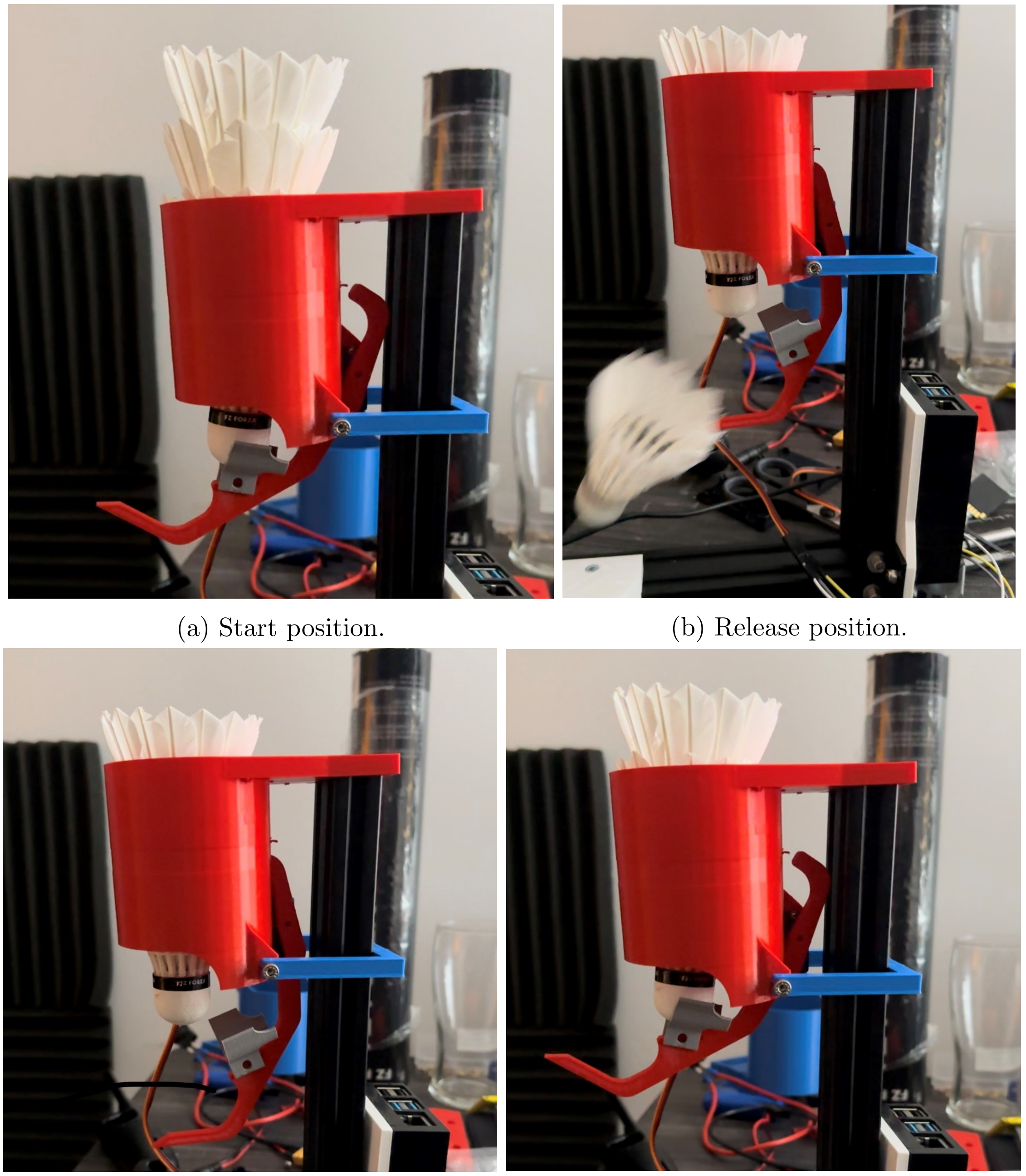
An alternative option would be to design a thicker arm at the base and incorporate the shuttle retainer as a single component. However, this approach was not pursued in the current stage of development due to time efficiency considerations. Printing smaller separate components was preferred over a larger single component, as it allows quicker fabrication without requiring extensive support structures.

The shuttle retainer component was 3D printed and firmly affixed to the arm. The modified arm, now equipped with the shuttle retainer, underwent additional testing, as illustrated in Figure 39 with the grey shuttle retainer mounted on the red arm. Furthermore, the Raspberry pi 4 was used with the code from the pre-project shown in Listing 1.

Initially, the SG90 servo motor was operated without the arm attached. The angle was set to zero before fitting it onto the servo, which was mounted on the tube. Subsequently, the angular endpoints were determined and found to be 0 and 40 degrees by testing.

Listing 1: Python code to control a 50Hz servo

```python
# Import libraries
import RPi.GPIO as GPIO
import time

# Set GPIO numbering mode
GPIO.setmode(GPIO.BOARD)

pwm_pin = 3
GPIO.setup(pwm_pin, GPIO.OUT)
servo = GPIO.PWM(pwm_pin, 50)

servo.start(0)

try:
    while True:
        angle = float(input('Enter␣angle␣between␣0␣&␣180:␣'))
        servo.ChangeDutyCycle(2 + angle / 18)
        time.sleep(0.5)
        servo.ChangeDutyCycle(0)

finally:
    servo.stop()
    GPIO.cleanup()
    print("END")
```



(a) Start position.

(b) Release position.

(c) Release position.

(d) Start position.

Figure 39: Testing of new mechanism.

The mechanism demonstrated functionality in principle; however, an issue arose where occasionally two SCs would fall simultaneously. Additionally, it was observed that the falling direction of the SC was not consistent.

The primary focus then shifted to addressing the issue of two SC falling simultaneously. To resolve this problem, several prototypes were developed and tested. Figure 40 showcases various iterations with different lengths and designs. The initial draft is represented by the blue prototype on the left, followed by the red prototype that demonstrated occasional success. The final prototype, depicted on the far right (the white), represents the most recent iteration.

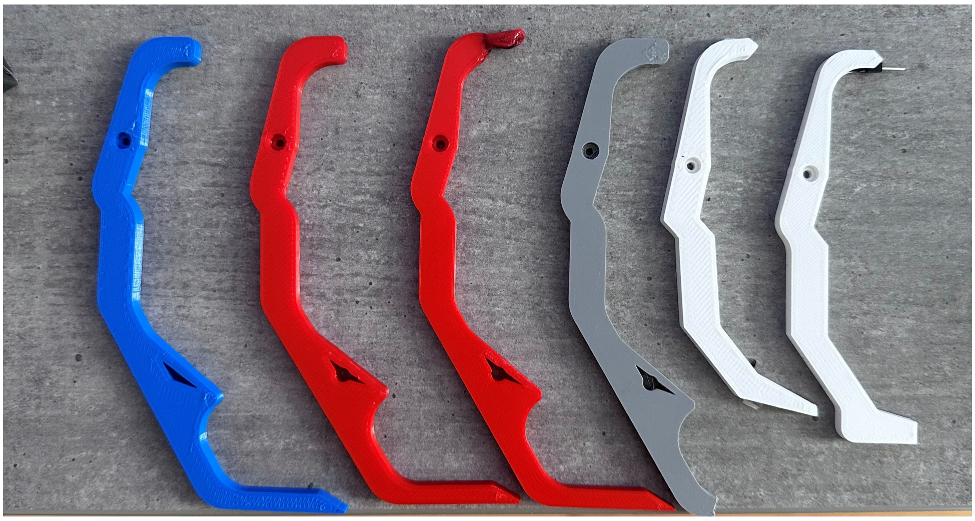


Figure 40: Evolution of feeding arms prototypes.

The difference between the initial prototypes, such as the blue one in Figure 40, and the red one in Figure 39, lies in the length of the arm at the bottom. This extra arm was initially intended to serve as a larger form of shuttle retainer guider. However, it was considered unnecessary since the existing shuttle retainer could perform the same function. Consequently, the additional arm was removed, and the shuttle retainer was redesigned to be larger.

The arm was also designed with a sharp tip at the top to achieve sufficient holding power, but this approach proved unsuccessful. Quite fortuitously, it was discovered that the male pinpoint of a jumper wire from a standard microcontroller cable (22AWG), as depicted in Figure 41, could be attached to the top of the arm. This adaptation led to promising results.

Consequently, an arm equipped with the pinpoint of a 22AWG jumper wire was employed to counteract the slippage of the SC. However, it is necessary to note that whilst it did not entirely address the problem, it represented a significant stride in the correct direction.

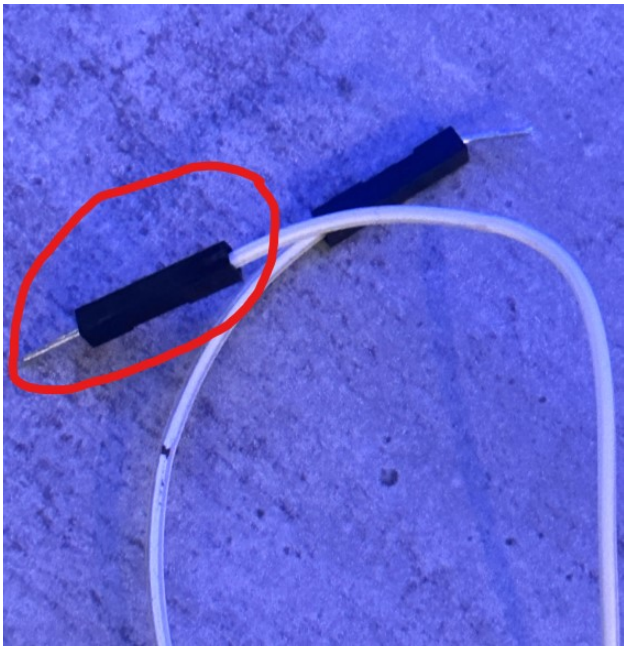


Figure 41: AWG cable tip circled in red.

7.2.2 Arm V1.2

Even with a sharp AWG tip at the bend's front, there was still a chance of simultaneously feeding two SCs. It was also noted that some SCs were missing feathers, leaving the arm with nothing to grasp.

One hypothesis suggested that the top bend might damage or wear out the feathers. As a result, new prototypes were designed without this bend, allowing for the attachment of up to four AWG tips.

Figure 42 presents several new prototypes called arm V1.2. Each prototype has a 5-degree difference in the angle of the AWG tip to see if this change has any impact.

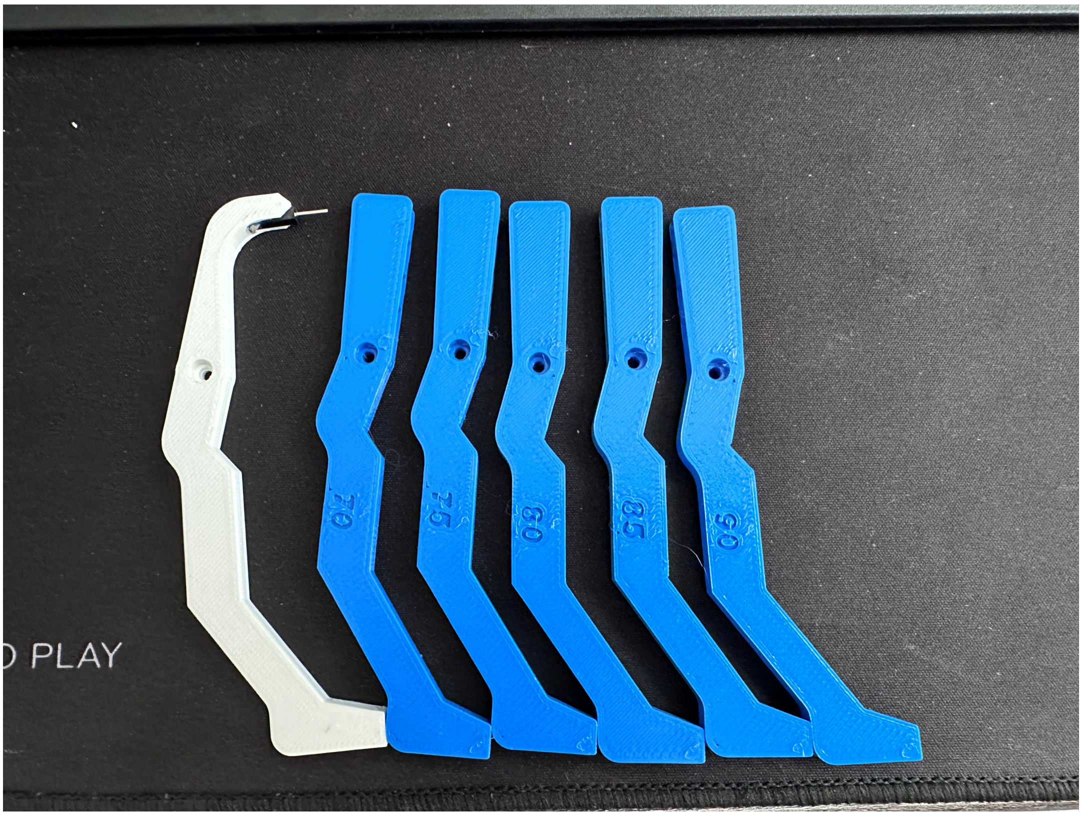


Figure 42: Arm prototype V1.2.

Each new prototype was fitted with 4 AWG tips and subsequently tested. The angles of 70, 75, and 80 degrees proved ineffective, and several SCs could slip through, and the final results showed that these prototypes did not provide any improvements. One of them broke During the dismantling process, prompting an experiment with a zip tie instead of an AWG tip.

As depicted in Figure 43, a zip tie was attached to provide additional length at the lowest possible point. Interestingly, this seemed to function as well as the original arm design with the bend, but without causing any damage to the SC feathers. This finding was further investigated for the development of arm V.2.

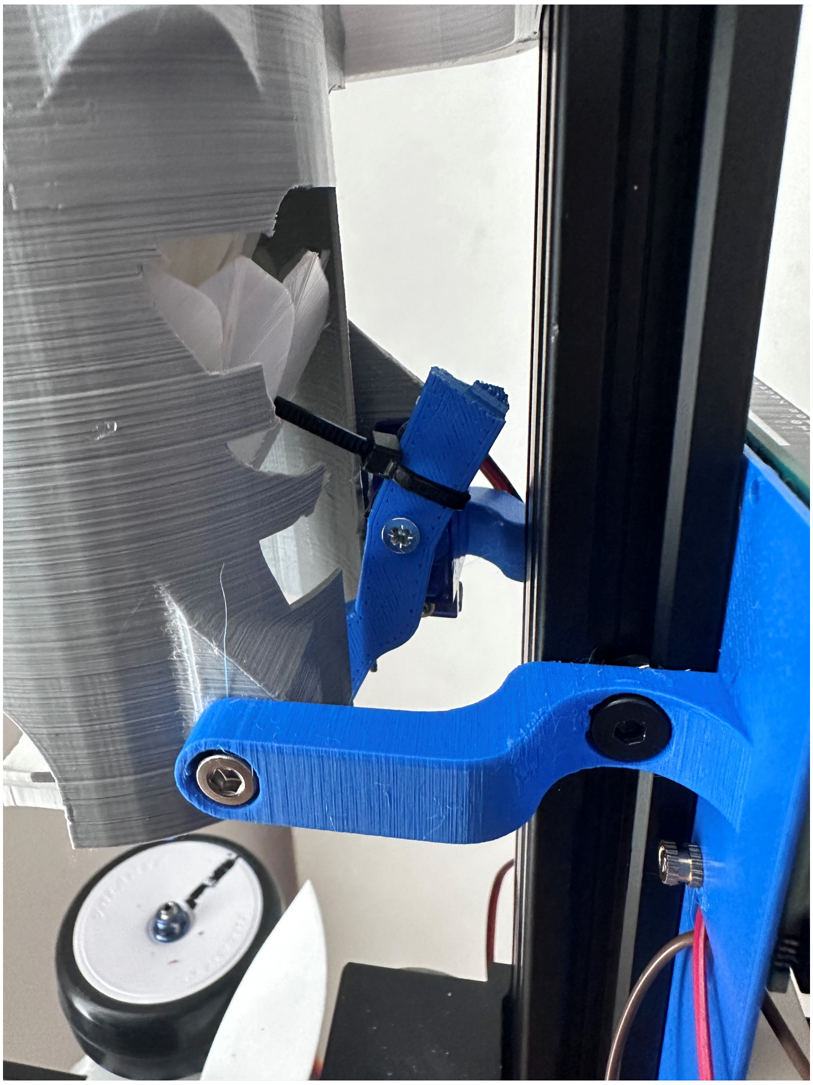


Figure 43: Arm with a zip tie.

7.2.3 Arm V.2

The zip tie's discovery led to a completely new CAD design. Some measurements were taken between two SCs this time. Figure 44 shows the length from the tip of the cork on the (SC below) to the lower string (on the SC above) is 62mm. And from the top string of the (SC above) to the tip of the cork on (SC below) is 56mm. These measurements were taken on to SW for drawing.

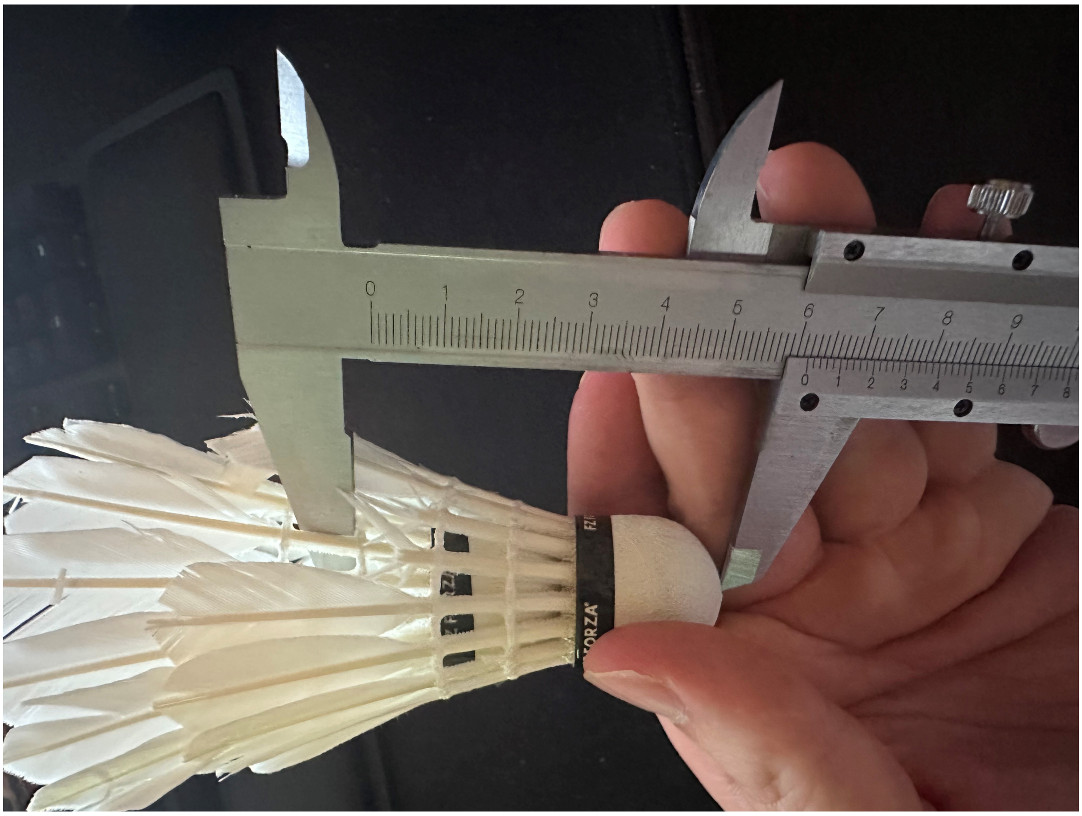


Figure 44: Measure between two SCs.

Figure 45 shows the CAD drawing where the red line indicates a tip that should hit between the two measurement points that were taken, and the green area is the resting point for the SC when it is in the starting position and also where the shuttle retainer will be mounted.

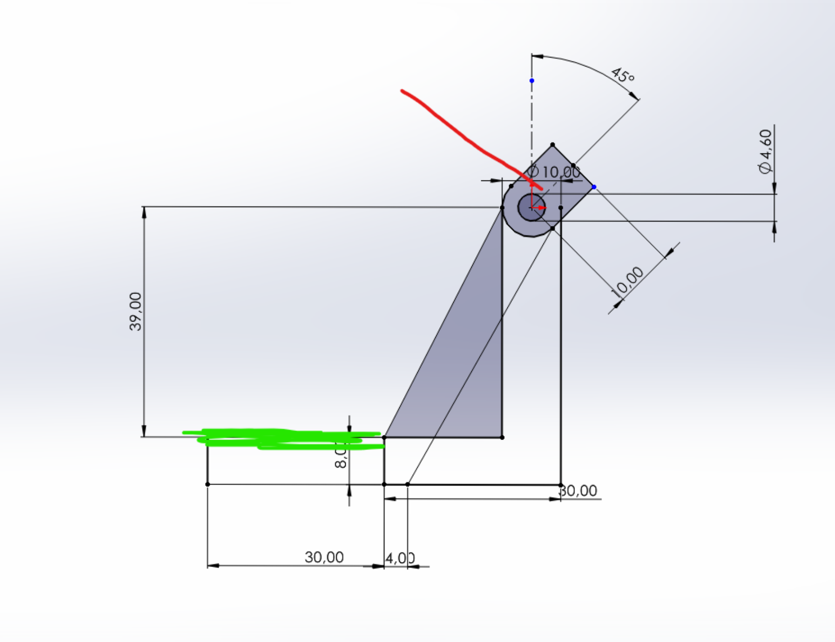


Figure 45: CAD design of the arm V.2.

The total length from the centre of rotation to the top of the green line is 39mm, where the shuttle retainer is mounted. Furthermore, two holes are located 6.5 and 8.5mm from the centre of rotation. to fit a 1.6 mm thick paper clip as the needle. The measurement from the needle's base starting point to the shuttle retainer totals between 45.5 and 47.5mm. It is hypothesised that the journey of the mechanism from its starting to ending position would extend the needle to a length between 56 and 62mm. As a result, the needle should be able to grasp the SC.

However, this is an assumption rather than a precise calculation. This is because calculating the fall time, speed, and the additional push from the needle at a certain point in the movement would be fairly inaccurate. While it's not impossible to perform such a calculation, the decision was made not to pursue it due to the substantial time commitment required for a mechanism whose efficacy was not yet confirmed.

The result of the CAD design is shown in Figure 45, where a slot has now also been added to fit the included attachments from SG90.

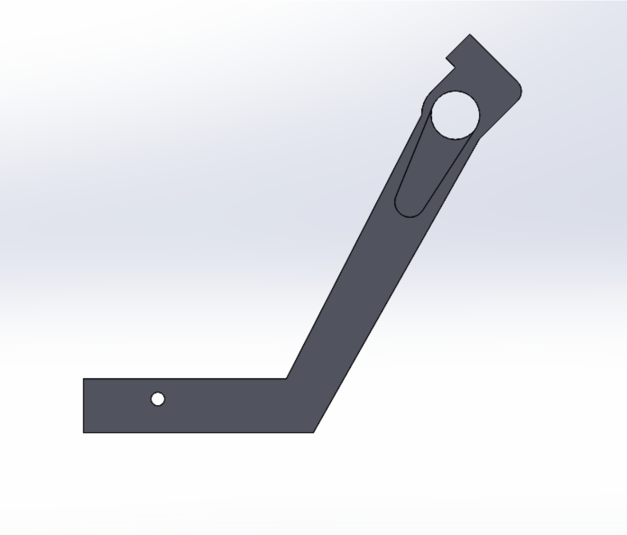
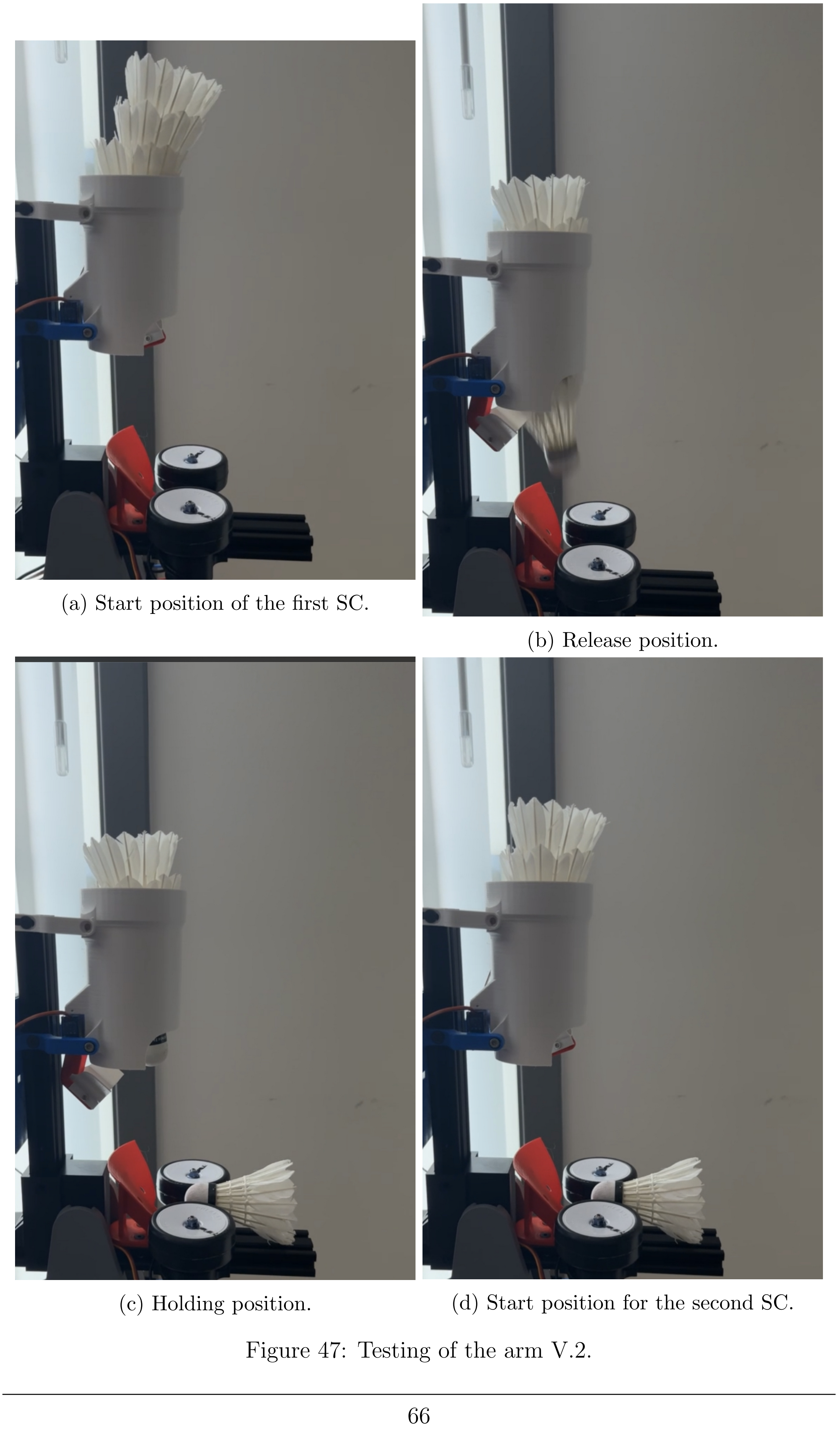


Figure 46: The Arm prototype V.2.

Once the design phase was complete, the arm was printed and prepared for testing. Figure 47 displays the sequence of feeding five SC within 10 seconds. For the initial test, the BLDC was not operational to observe the landing point of the SC.

Figure 47a shows the starting position for the feeding mechanism. When the arm starts to rotate, as shown in Figure 47b, the needle catches the SC above the falling one. The needle thus carries all the weight of all the SC above as shown in Figure 47c and returns to the start point in 47d.



(a) Start position of the first SC.

(b) Release position.

(c) Holding position.

(d) Start position for the second SC.

Figure 47: Testing of the arm V.2.

The mechanism worked. However, the last SC struggled to drop. This can be due to two assumed reasons from observing. The first observation was the edge between the arm and the tube, as shown in Figure 48 rounded in red. The edge becomes sharp when printed, making the cork of the SC get stuck.

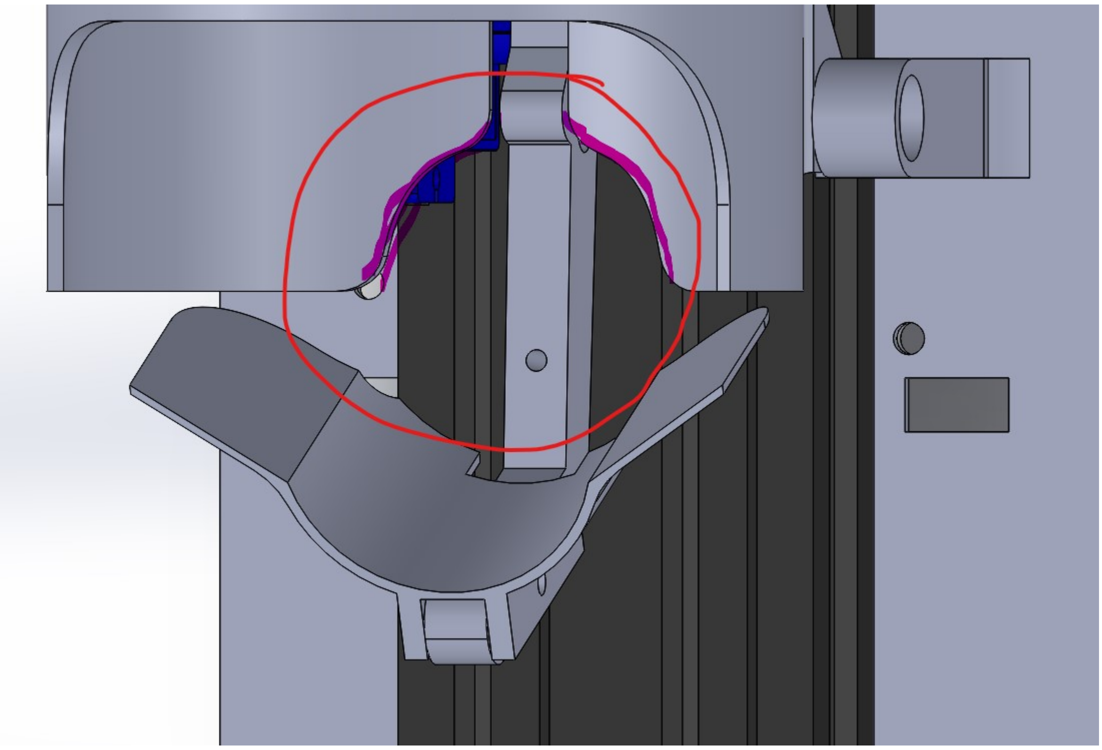


Figure 48: Jamming spot between the tube and the arm.

Having an SC become stuck is not ideal. According to the performance goals outlined in section 1.4, the SCL should be free of jams to avoid interruptions during training. Therefore, a notch was incorporated into the arm, as illustrated in Figure 49, and a ”pusher” was created to remove any space where the cork might become stuck between the arm and the tube. The pusher is depicted as the red part in Figure 50.

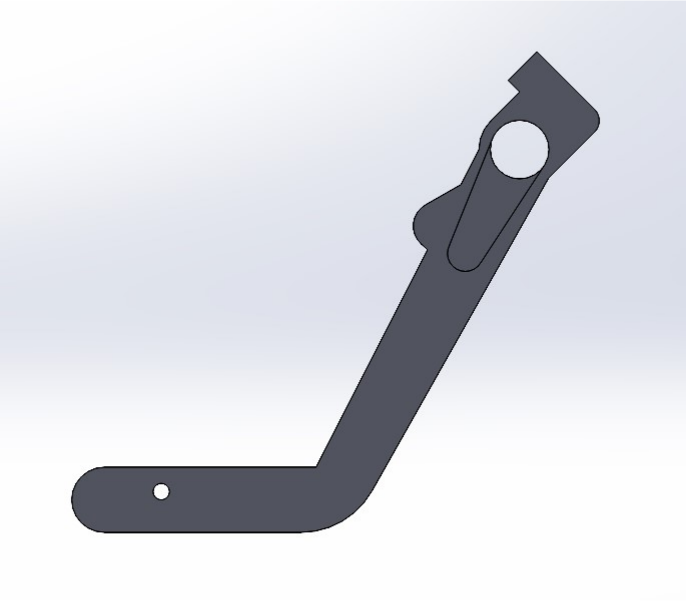


Figure 49: The arm V.2 with the notch added.

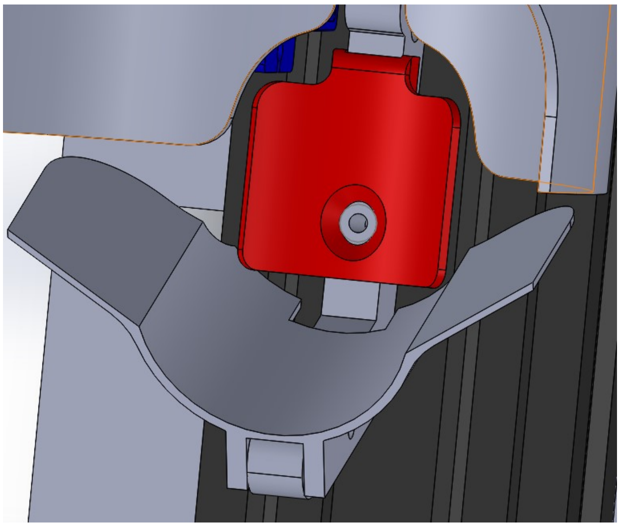


Figure 50: The pusher.

The second reason could be the rough surface on the tube that comes from the 3D printing. It has been observed that the tips of the feathers tend to get stuck in the roughness of the tube surface.

The rough surface on the tube is assumed to be something called z-wobbling, which was mentioned as one of the pitfalls in 3D printing. Wobbling occurs when the z-axis moves. By experience, the fastest and easiest way to fix the problem is to turn off the z-hop function. All slicers have this option. The z-hop function slightly lifts the z-axis during infill change and then lowers itself at the same layer for the next infill area. If the precision is not 100% correct, it will start printing with an offset that never recovers. Thus, The Z-hop function was switched off, and a new one was printed.

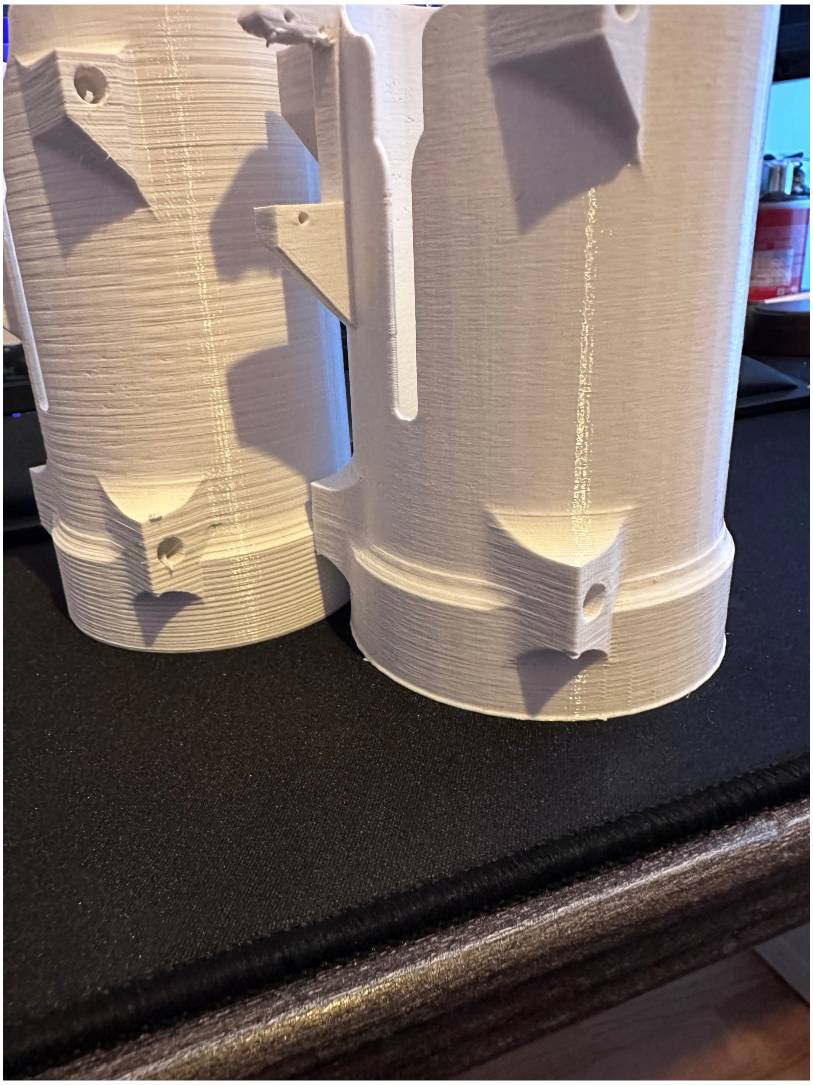


Figure 51: Tube surface roughness.

As demonstrated in Figure 51, the initial assumption was accurate. It appears that the 3D printers in use have not been properly maintained. By disabling z-hop, the results improved. Maintenance must be performed on the 3D printer's axes to achieve better results, such as tightening belts or applying lubrication. Another solution if this doesn't work involves using sandpaper to smooth the inside of the tube, removing any edges.

After addressing these two assumptions, the feeding mechanism was tested again. It functioned effectively and rarely dropped two items at once. However, it was observed that an SG90 motor might be too weak when transitioning from the drop to the start position. This issue can be resolved by using a rubber band between the top of the arm and the vertical aluminium profile or employing a stronger servo motor. The decision was made to test a more powerful motor alternative.

The newly selected servo motor was a 17-gram Metal Gear Digital servo manufactured by FMS. This motor was chosen because the SG90 motor frequently appeared weak for the task, and its gears tended to sustain damage during extensive testing. It was also considered that the repeated mounting and dismounting of numerous arm prototypes could have contributed to the wear and tear of the gears. Regardless, the metal gear of the new servo motor, with its additional 1kg*cm of torque, is expected to be a more reliable solution.

Upon the arrival of the new servo motor, the tube was redesigned to accommodate the FMS Servo. Due to the larger motor chassis, it had to be flipped, as illustrated in Figure 52. Moreover, the larger axle required a bigger arm to fit. The new arm maintains the same dimensions as the previous version, only thicker around the servo motor connection, as depicted in Figure 53.

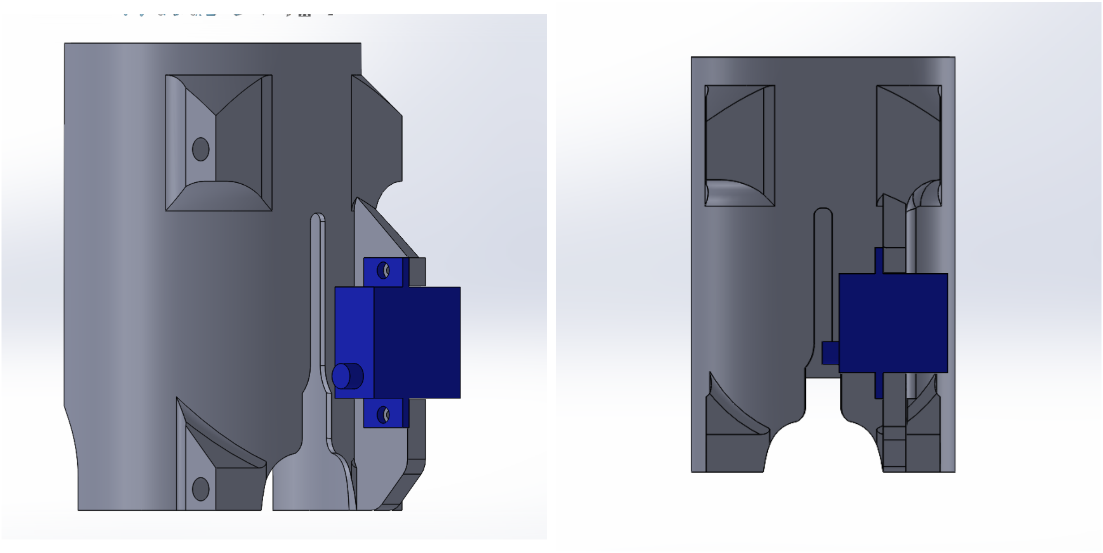


Figure 52: The tube with FMS 17g servo.

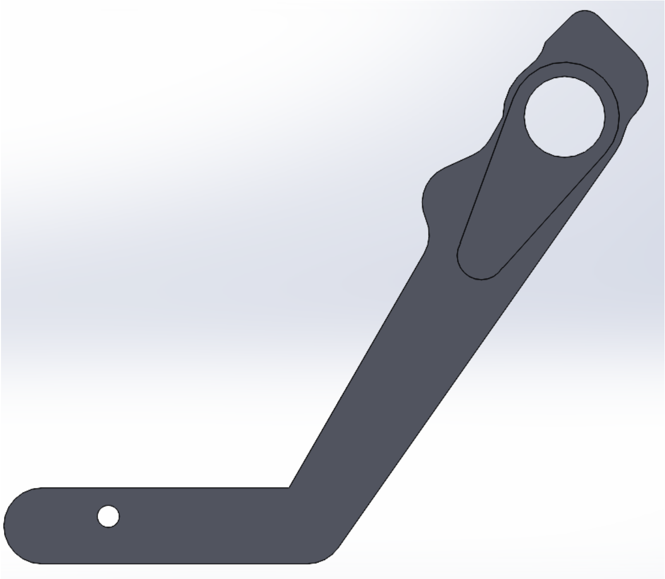


Figure 53: The arm made to fit FMS 17g servo.

After the redesign, the components were printed and subjected to preliminary testing. Everything operated as anticipated and was prepared for more detailed testing. This would occur after all the electronic components had been integrated with a program, allowing for simultaneous testing of all parts. This was the next stage in the process. Chapter 10 construction review will provide a detailed overview of all the parts from this chapter.

8 Transition from Raspberry Pi to Arduino

This chapter clarifies the process of transferring the existing program designed for testing mechanisms by a Raspberry Pi to an Arduino platform. The reasons for the transition will be discussed, along with the advantages and challenges associated with the conversion. Moreover, a detailed explanation of wiring and connecting the components will be provided, accompanied by circuit diagrams and codes. The presented information intends to provide a thorough understanding of the transfer process and contribute to the existing literature on the subject.

8.1 Microcontroller

The main reasons for transitioning to Arduino are cost-effectiveness and availability on all general Arduino microcontrollers compared to the Raspberry Pi 4 model B. Additionally, the Arduino is a microcontroller, not a single-board computer [18], making it more suitable for projects requiring simpler functionality and lower power consumption. It has thus been chosen to use an Arduino UNO R3. NTNU has multiple for students to use. However, the smaller and more compact versions, such as a raspberry pi Pico or Arduino Nano, can be used for the prototype but were unavailable.

Considering the advantages and disadvantages, the Raspberry pi 4 model B is relatively expensive on the Norwegian market and costs approximately NOK 2,900 [59]. This is not usually the valid price to state since they are obtainable for around NOK 800-1,100. However, in the current situation with stock status and demand, NOK 2,900 will be the price if it is wanted immediately.

In contrast, an Arduino UNO R3 costs approximately NOK 360 [60]. The smaller models, such as Raspberry Pi Pico and Arduino Nano, cost approximately NOK 100 [61], and NOK 150 [62], respectively. Ultimately, the choice between Raspberry Pi and Arduino will depend on the specific requirements of a project. In this case, the need for the computing power of a Raspberry Pi 4 model b is not required, though all the options that have been stated work for the SCL.

To program a microcontroller to control the BLDC, ESC, and servos, the following conditions should be met:

1. Use a programming language such as C++ or Python, depending on the microcontroller.

2. Program the microcontroller to control the ESC, BLDC, and servos.
3. Connect the ESC, BLDC, and servos to the microcontroller using the appropriate wires and connectors.
4. Ensure proper power supply for the components.
5. Additional hardware such as motor drivers or power regulators may be required depending on the specific components used.

The list was followed from point 5 to point 1, and a hardware summary was made after choosing a microcontroller.

Regarding the power supply, the SCL current peak draw is 0.5 Amps when both MSA and MSB are started up simultaneously. However, they got a maximum average draw of 0.2 Amps when both ran smoothly at a constant velocity. Moreover, the battery used is labelled 11.7-volt, but from the measurements of 12.7-volt. However, the size of the battery will be discussed in the cost calculation 10.

8.2 Hardware

The hardware components required for the SCL currently include:

- A 11.7V battery power supply with a charger
- 2 ESC
- 2 BLDC
- 3 Digital Servos
- 1 Microcontroller

Chapter 5 includes code (A) utilised for controlling the ESC. However, it was discovered during this chapter that the two BLDC motors did not exhibit the same speeds at identical pulse width. One option to address this issue is to adjust the value on one of the ESC units by the number of pulse widths, either positively or negatively. Alternatively, both motors can be operated using separate potentiometers. Employing one or two potentiometers makes it feasible to adjust the speeds of the SCL without needing a connected computer. Additionally, a code (1) has been employed to control servos, including the feed mechanism (7), which can also be applied to the pan/tilt servos.

All the codes mentioned are in Python and must be rewritten into C++ for the Arduino IDE or use ready-made codes to rewrite those. However, before code writing, the machine control method must be determined. For testing purposes, finding a method to adjust and use all the mechanisms without being connected to a computer would be reasonable. A joystick was thus added to control the pan/tilt function, with a button that triggers the feed mechanism and two potentiometers to adjust the speeds of the BLDC.

The new list of hardware components is now listed as follows:

- Power supply
- 2 ESC
- 2 BLDC
- 2 Potentiometers
- 3 Digital Servos
- 1 Joystick
- 1 Microcontroller

After all, the parts had been determined. A wiring diagram was made.

8.3 Wiring

Ensure that the components receive an appropriate power, as mentioned in the list at 8.1. Figure 54 demonstrates two alternative wiring configurations designed in Fritzing[5]. The battery size has not yet been calculated; however, the required voltage is typically indicated on the ESC, usually ranging from 11-22 volts. It is usually stated on the ESC with 3-6s. Where s stands for cells and a li-ion cell is often at 3.6 or 3.7 volts [63]. Three cells will therefore be around 11 volts, and six cells will be around 22 volts

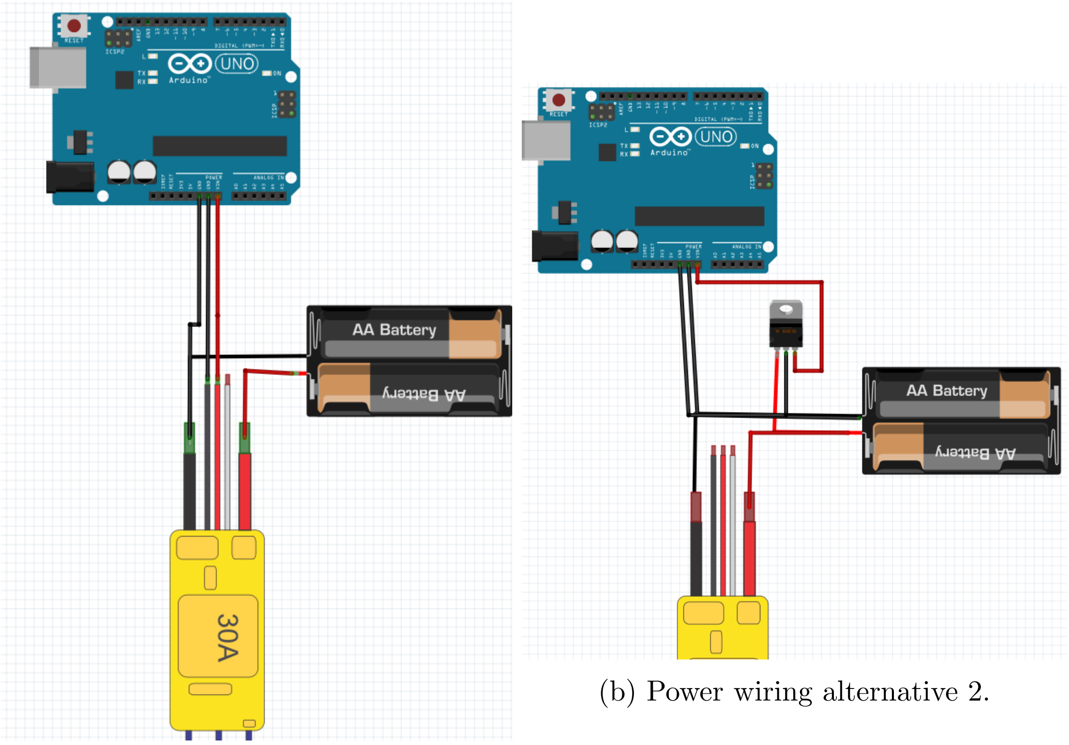
Alternative 1 uses the built-in voltage reducer within the ESC to power the microcontroller. This is achieved by connecting the red wire from the ESC to the VIN on

[5] Fritzing is a free and open-source software tool used for designing and prototyping electronic circuits. It provides a user-friendly interface for creating and editing schematics, PCB layouts, and interactive views of the designed circuits. https://fritzing.org/.

the Arduino. Additionally, the ground connections from the battery, microcontroller, and ESC are linked to the other electronics powered by the current.

Alternative 2 involves connecting the current from the battery to a separate voltage reducer, which can be purchased as either adjustable or fixed. In this alternative, the voltage regulator must be grounded to the battery and microcontroller.

Alternative 1 was selected since the Dual Sky ESC provides a 5-volt output through its 3-pin connection. The next step from the list 8.1 was 3; Connect the ESC, BLDC, and servos to the microcontroller using the appropriate wires and connectors.



(a) Power wiring alternative 1.

(b) Power wiring alternative 2.

Figure 54: Power wiring alternatives.

After establishing a method for power supply, a wiring diagram was created using Fritzing, as depicted in Figure 55. The colour coding is as follows: red represents power, black represents the ground, green is the signal dedicated to the feeding mechanism, and yellow and grey can be either analog or PWM signals.

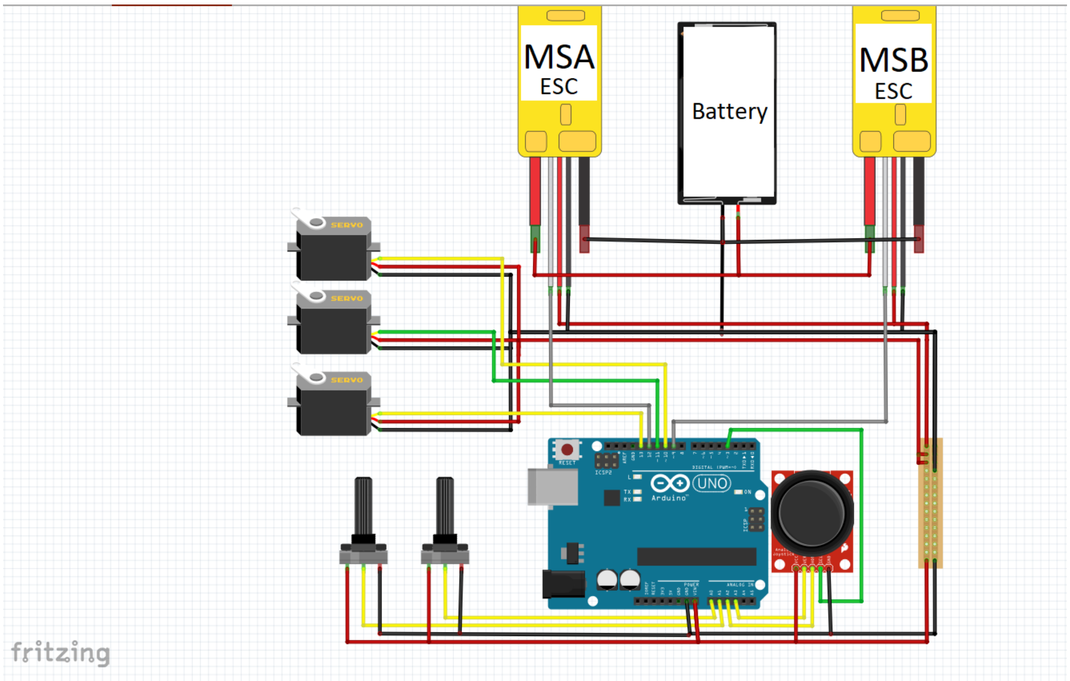


Figure 55: Wiring diagram for the SCL

At the top of the diagram from Figure 55, MSA and MSB ESC are positioned with the battery in between. As illustrated, the thick red cables represent the main current and are connected only to the battery's positive terminal to each ESC. Additionally, the ground connections for MSA and MSB are linked to the battery and extended to a breadboard. The breadboard receives ground and power from both the 3pin connections from ESC of MSA and MSB. This setup enables the distribution of current from the breadboard to components, such as a PCB. Power and ground connections from the breadboard are directed to the feeding, pan and tilt servos, joystick, potentiometers and microcontroller. Meanwhile, individual cables run from the microcontroller to each module, sending or receiving signals.

Based on the parts and wiring used for the SCL, a mounting plate was constructed to hold the microcontroller, potentiometer and cable management. The constructed Arduino mounting plate is shown in Figure 56.

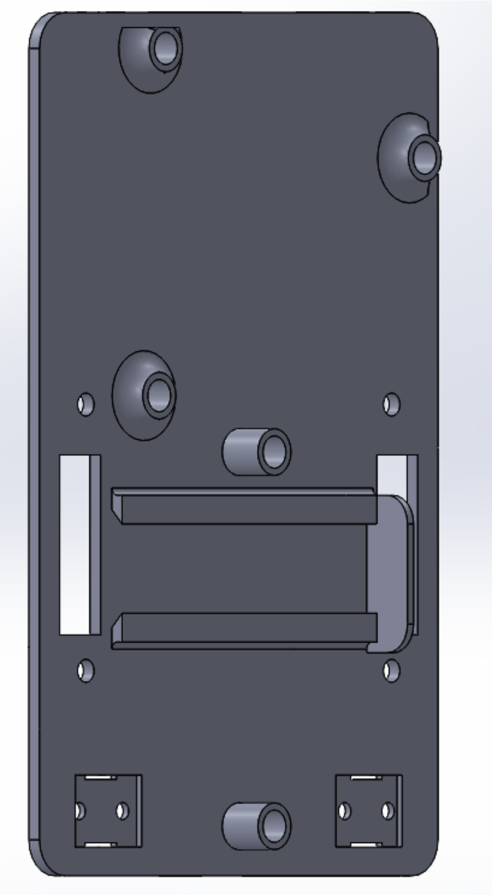
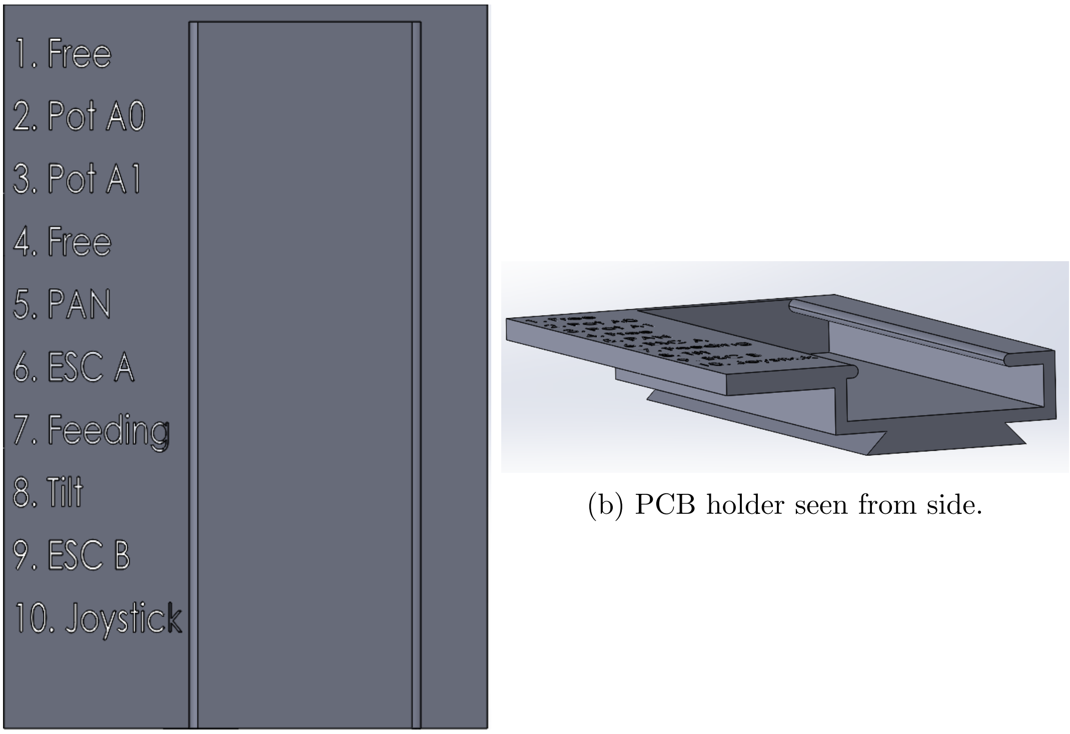


Figure 56: The Arduino UNO R3 mounting plate.

On the bottom of the Arduino plate are two slots for the 10K OHM TYPE potentiometers to be attached. The potentiometer attached to the plate is shown in Figure 59, and two potentiometers of the same type can be bought on eBay for NOK 50 [64]. In the centre of the mounting plate, two rails with a plate at the end, are placed between two square holes. These rails are made to fit a PCB holder that was designed as shown in Figure 57, while the square holes are for pulling cables in and out of the connection box. The PCB holder has numbers that describe where the connection points are in sequence.



(a) PCB holder seen from above.

(b) PCB holder seen from side.

Figure 57: The PCB holder.

Furthermore was, a design to hold the joystick as a remote control found and printed. The design and files were found on Thingiverse with the hyperlink in the figure description and citation to the creator.

Figure 58: Printed and mounted joystick housing from Thingiverse [65].

After all the parts were printed, it was easier to set up wiring in a physical model, and the soldering phase of the breadboard PCB, hereby called PCB, could start.

8.4 Soldering

Before starting the soldering process, a list of the required equipment was prepared, as shown in Table 10. This table outlines the necessary items for creating a PCB.

Table 10: List of equipment for soldering PCB.

No.	Item	Description
1	6x14 Soldering breadboard	Soldering breadboard are where the different headers will be soldered on and also used as a distribution for power, the ground and common connection point for all servos and ESC
2	9x3 header	A connector with nine pins arranged in three rows will be used for potentiometers, servos and ESC connections.
3	1x5 header	A single-row connector with five pins, used for Joystick.
4	1x5 bended header	A 1x5 header with a right-angle bend are useful for space-constrained designs or different orientations and used for PWM pin connectors.
5	1x4 bended header	A 1x4 header with a right-angle bend, commonly used for compact connections. Used for the analog inputs
6	1x2 bended header	A 1x2 header with a right-angle bend, suitable for power or signal connections in limited spaces. Used for ground and power to the microcontroller
7	2 1x3 male header	Two single-row connectors with three male pins each are used for component connections. Onto the potentiometer to get a secure connection
8	2 1x3 female headers	Two single-row connectors with three female receptacles each, for component or wire connections. used to connect the potentiometers onto the PCB
9	Wires and accessories	Wires of different gauges (e.g., 22 AWG, 24 AWG) for connecting components and headers. Accessories such as shrinking tubing

Once all the required equipment was gathered, the soldering process began.

The first step involved soldering the male header (Item No. 7 in Table 10) onto the potentiometers. This step ensures a secure connection between the potentiometers and the headers, as shown in Figure 59.

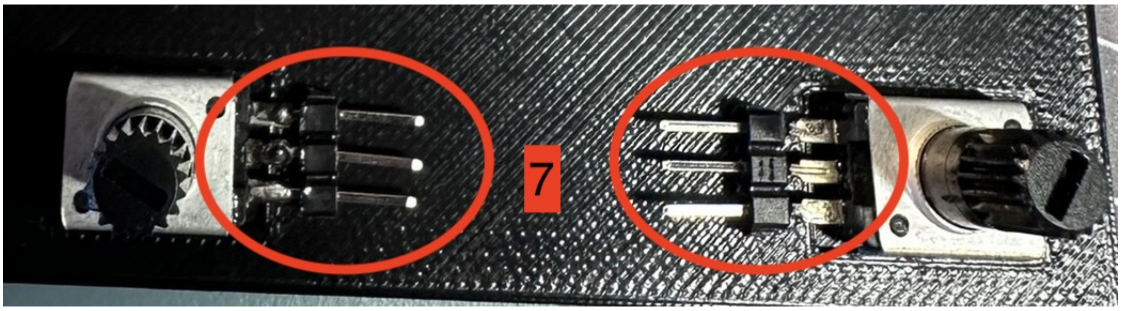


Figure 59: Headers soldered onto the potentiometers.

Following the wire connections, a soldering breadboard (Item No. 1) was prepared by cutting it to a size of 5 rows and 16 columns. Subsequently, the 9x3 header (Item No. 2) and the 1x5 header (Item No. 3) were soldered onto the breadboard. It's important to note that the 5V input/output row is positioned between the ground and signal pins and shorted along the entire column. Similarly, the grounding connections are established similarly, as depicted at the top of Figure 60. This arrangement ensures consistent and efficient power distribution and ground connections throughout the breadboard.

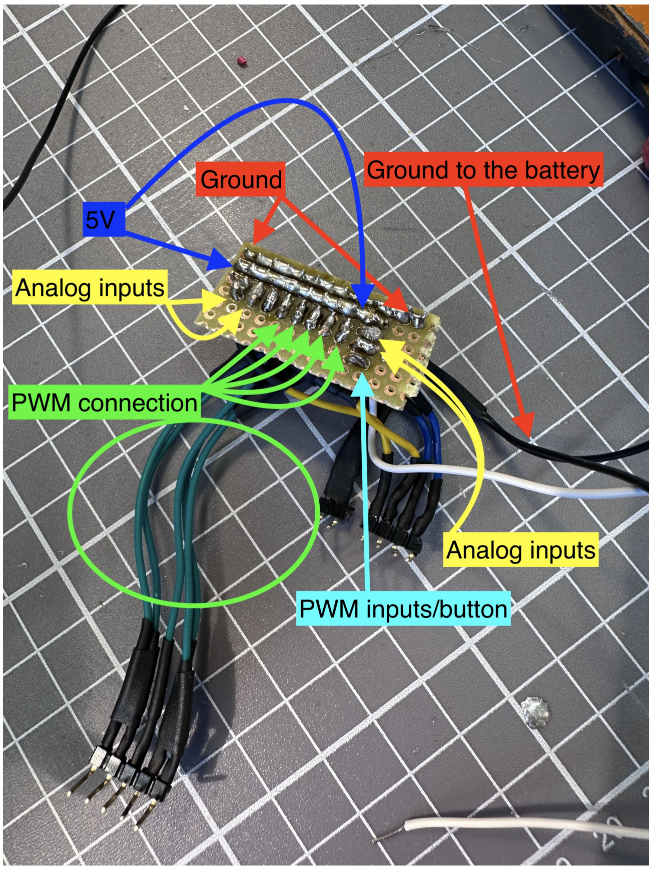


Figure 60: Underneath the breadboard PCB.

After preparing the soldering breadboard (Item No. 1) and soldering the headers (Item No. 2 and 3), a variety of 0.6mm equipment wires in different colours (Item No. 9) were cut and sorted. These wires were then carefully connected to their respective header pins, as shown in Figure 61. A colour-coding system was employed to improve the clarity and organisation of the connections. Specifically, the green wires were dedicated to the header pins from (Item No. 2) to the output pins (Item No. 4). The yellow wires were used for the potentiometer connection pins. In contrast, the blue wires were designated for the joystick's x and y signals, connected to (Item No. 5). This strategic use of colour enables better visualisation and comprehension of the wiring arrangement, facilitating the distinction of specific connections made within the breadboard, as illustrated in Figure 60.

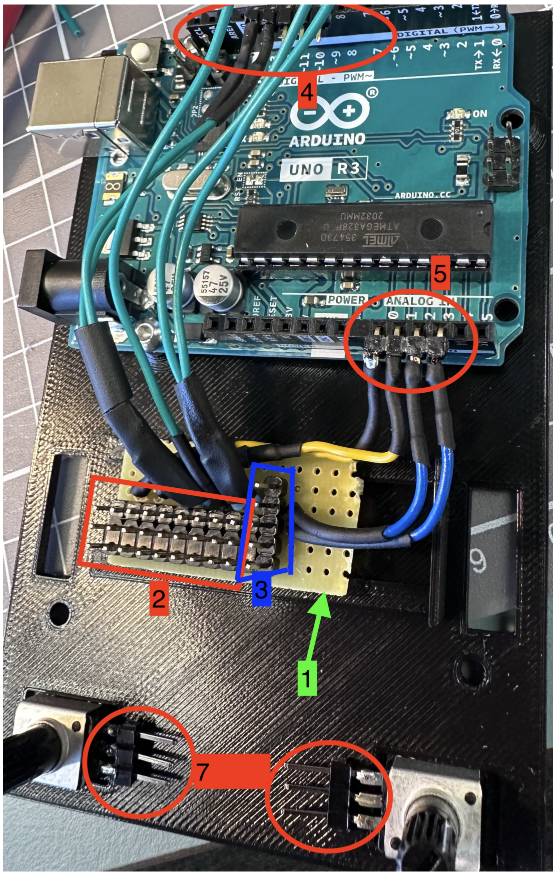


Figure 61: Soldering and wiring progress.

The remaining wires were then soldered individually, ensuring they were placed to reach their designated positions without excessive stretching. The potentiometers were also soldered with their cables and secured against (Item No. 8). This was done because the original headers did not fit the female headers in the mechatronics lab. And afterwards, it turned out that the female header was too large, so cables were directly soldered onto the headers.

The finalised PCB, showcasing the completed soldering and wiring, is shown in Figure 62. Notably, the white cable represents the signal from the push button on the joystick, which serves as an input and is connected directly to PWM pin 3. This cable is not soldered onto a header but can be easily inserted into the PWM pin.

Furthermore, two additional cables were utilised to establish a secure grounding connection between the PCB and the battery. This redundancy was implemented to ensure sufficient grounding and that at least one cable will always function reliably [66].

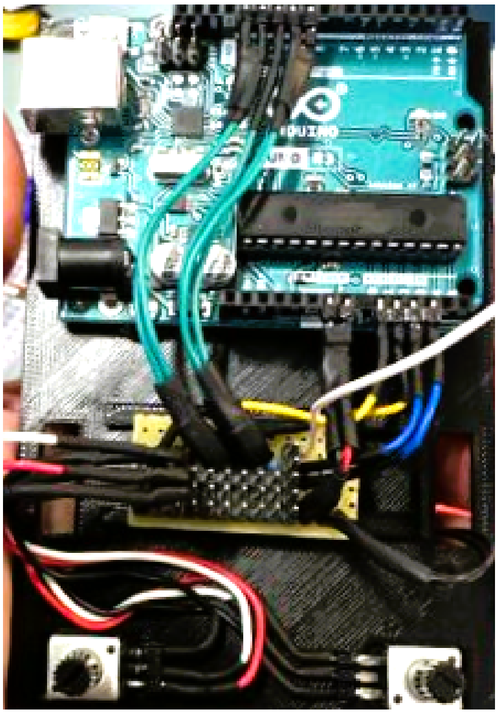


Figure 62: Finalised PCB with soldering and wiring completed.

Figure 63 visually represents the fully connected PCB with all components in place. This comprehensive illustration provides an overview of the interconnected components and their connections.

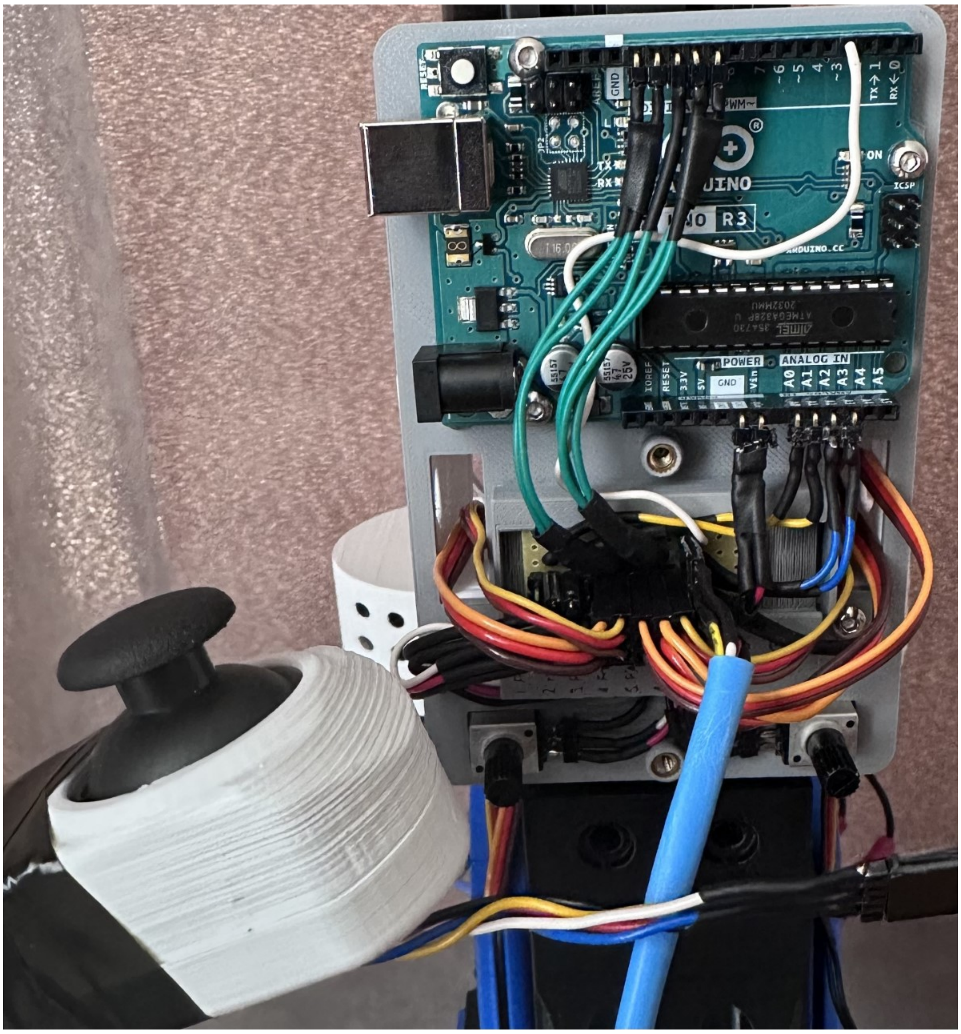


Figure 63: Fully connected PCB with all components.

In addition, Figure 64 showcases the connection of the main current using WAGO connectors. This method ensures a secure and reliable connection for the main current flow.

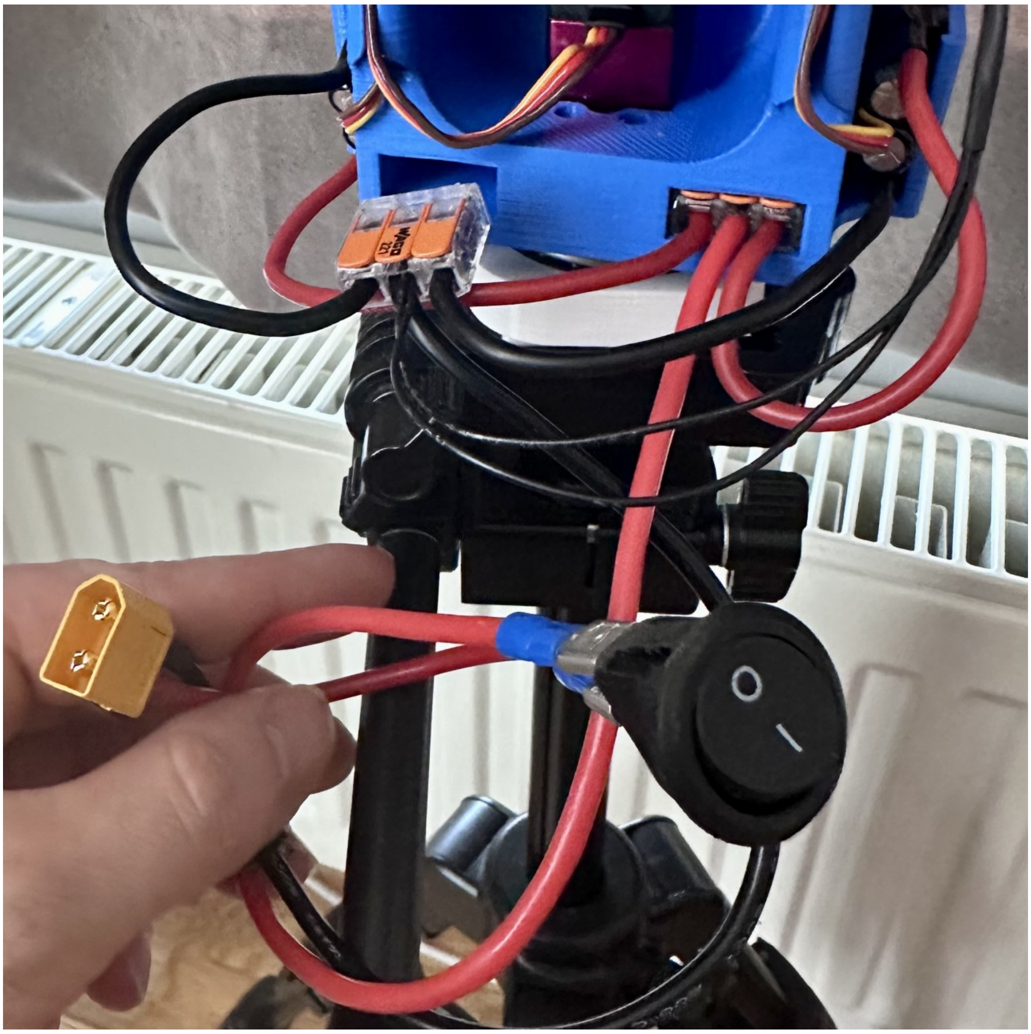


Figure 64: Connection of main current using WAGO connectors.

These Figures illustrate the completed wiring and connections, offering a visual reference for the successful wiring and soldering of the PCB. The PCB makes the objective ease of assembly more relevant as it avoids many loose wires going in all directions and everything collected on a board.

After soldering each connection point, a multimeter with a ”short-circuit” function was utilised to verify the presence of a connection. This crucial step involved checking each solder point individually and ensuring no unintended short circuits were between different points on the PCB.

By carefully examining each connection with the multimeter, the continuity of the electrical paths was confirmed, and any potential issues, such as short circuits, were identified and fixed. Before proceeding with the programming phase, this process ensures the PCB connections’ integrity and reliability.

With the completed thorough examination and verification of the soldered points, the next step involved commencing the programming process.

8.5 Programming

The final step in the process, as outlined in the list in 8.1, involved programming all the components.

Each component was coded individually during the programming phase according to the list outlined in 8.1. This approach allowed for a better understanding of the components and simplified the troubleshooting process. Two separate codes were then developed and subsequently merged into a single program. These codes included ESC control with a potentiometer, push-button control for the feeding mechanism and joystick control for the pan/tilt mechanism.

An existing program was found as a starting point from the Arduino IDE library. The code was modified to suit the thesis's specific requirements for controlling two ESC. The code Listing 2 provided a foundation for controlling a single ESC using a potentiometer [49].

Listing 2: ESC Control with Potentiometer [49].

```
// PWM pin for the ESC
int throttlePin = 9;

// Analog input pin for the potentiometer
int potPin = A0;

// maximum throttle value (corresponding to full speed)
int maxThrottleValue = 1200;
// minimum throttle value (corresponding to stopped motor)
int minThrottleValue = 1100;
// initial throttle value
int throttleValue = 0;

void setup() {
  pinMode(throttlePin, OUTPUT);
}

void loop() {
    // read the value of the potentiometer
  int potValue = analogRead(potPin);
  // map the potentiometer value to a throttle value
  throttleValue = map(potValue, 0, 1023, minThrottleValue,
      maxThrottleValue);
  // send the throttle value to the ESC
  analogWrite(throttlePin, throttleValue);
  // wait for a short delay
  delay(20);
}
```

After successfully testing the code on both MSA and MSB, the next step was to modify the code to allow each potentiometer to control its respective ESC independently. Rewriting the code assigned separate control signals to each potentiometer and its corresponding ESC. This modification enabled individual control over the speed and direction of each motor, as shown in Listing 3.

Listing 3: ESC Control with Separate Potentiometer.

```
// PWM pins for the ESCs
int throttlePin1 = 9;
int throttlePin2 = 12;
// analog input pins for the potentiometers
int potPin1 = A0;
int potPin2 = A1;
// maximum and minimum throttle values for each ESC
// ESC A
int maxThrottleValue1 = 1180;
int minThrottleValue1 = 1170;
// ESC B
int maxThrottleValue2 = 1253;
int minThrottleValue2 = 1130;

// initial throttle values
int throttleValue1 = 0;
int throttleValue2 = 0;

void setup() {
pinMode(throttlePin1, OUTPUT);
pinMode(throttlePin2, OUTPUT);
}

void loop() {

// read the values of the potentiometers
int potValue1 = analogRead(potPin1);
int potValue2 = analogRead(potPin2);

// map the potentiometer values to throttle values between the min and max values for each ESC
// ESC A:
throttleValue1 = map(potValue1, 0, 1023, minThrottleValue1, maxThrottleValue1);
// ESC B:
throttleValue2 = map(potValue2, 0, 1023, minThrottleValue2, maxThrottleValue2);

// send the throttle values to the ESCs
analogWrite(throttlePin1, throttleValue1);
analogWrite(throttlePin2, throttleValue2);

// wait for a short delay
delay(20);
}
```

The code was successfully tested, and the throttle values for ESC A and B were verified using a tachometer. Without a tachometer, it is possible to drive the BLDC manually s up until the wheel's physical shake is observed, then adjust the throttle down from that point and mark it as the maximum. Connecting the system to a PC via USB can make monitoring and analysing the values easier through a serial plotter.

Furthermore, it was discovered that manual arming of the ESC had to be done every startup. To address this, the code was modified to include automatic arming. The latest version of the code, which includes automatic arming and serial plotting functionality, is shown in Listing 4.

Listing 4: Code for controlling ESCs with potentiometers.

```
// PWM pins for the ESCs
int throttlePin1 = 9;
int throttlePin2 = 13;

// analog input pins for the potentiometers
int potPin1 = A0;
int potPin2 = A1;
// maximum and minimum throttle values for each ESC
// ESC A
int maxThrottleValue1 = 1180;
int minThrottleValue1 = 1170;
// ESC B
int maxThrottleValue2 = 1253;
int minThrottleValue2 = 1130;
// initial throttle values
int throttleValue1 = 0;
int throttleValue2 = 0;

void setup() {
pinMode(throttlePin1, OUTPUT);
pinMode(throttlePin2, OUTPUT);

// Start serial communication at 9600 baud
Serial.begin(9600);

// Arm the ESCs
armESC(throttlePin1);
armESC(throttlePin2);
}

void loop() {
// read the values of the potentiometers
int potValue1 = analogRead(potPin1);
int potValue2 = analogRead(potPin2);

// map the potentiometer values to throttle values between the min and max values for each ESC
// ESC A:
throttleValue1 = map(potValue1, 0, 1023, minThrottleValue1,
    maxThrottleValue1);
// ESC B:
throttleValue2 = map(potValue2, 0, 1023, minThrottleValue2,
    maxThrottleValue2);

// send the throttle values to the ESCs
analogWrite(throttlePin1, throttleValue1);
analogWrite(throttlePin2, throttleValue2);

// send throttle values over serial
Serial.print("Throttle1:␣");
Serial.print(throttleValue1);
Serial.print("␣Throttle2:␣");
Serial.println(throttleValue2);

// wait for a short delay
delay(20);
}

void armESC(int pin) {
// Send maximum throttle value to arm the ESC
analogWrite(pin, maxThrottleValue1);
delay(2000);
// Send minimum throttle value to disarm the ESC
analogWrite(pin, minThrottleValue1);
delay(1000);
}
```

The push button functionality was programmed next. An example program for buttons and digital servos was found in the Arduino IDE library and combined to test the feeding mechanism. Subsequently, the joystick control for pan/tilt was added by rewriting and testing the joystick signals through the serial plotter. Both functionalities were integrated into a single program, as shown in Listing 5.

Listing 5: Code for Servo and Joystick Control.

```
#include <Servo.h>

// Button and first servo configuration
const int buttonPin = 3;
const int servoPin = 11;
const unsigned long debounceDelay = 50;
int startPosition = 66;
int targetPosition = 0;

// Joystick and tilt/pan servos configuration
const int tiltJoystickPin = A2;
const int panJoystickPin = A3;
const int tiltServoPin = 10;
const int panServoPin = 13;

// Constants for tilt
const int minTiltAngle = 0;
const int maxTiltAngle = 44;
const int startTiltAngle = 5;

// Constants for pan
const int minPanAngle = 0;
const int maxPanAngle = 180;
const int startPanAngle = 90;

// Servo speed
const unsigned long updateInterval = 20; // 20 milliseconds
const int angleChange = 2;

// Create servo objects
Servo myservo;
Servo tiltServo;
Servo panServo;

// Variables for button press handling
volatile bool buttonPressed = false;
unsigned long lastDebounceTime = 0;

// Variables to store the last servo angles
int lastTiltAngle = startTiltAngle;
int lastPanAngle = startPanAngle;

// Variable to store the last update time
unsigned long lastUpdateTime = 0;

void setup() {
  pinMode(buttonPin, INPUT_PULLUP);
  attachInterrupt(digitalPinToInterrupt(buttonPin), onButtonPress,
      FALLING);
  myservo.attach(servoPin);
  myservo.write(startPosition);

  tiltServo.attach(tiltServoPin);
  panServo.attach(panServoPin);

  tiltServo.write(startTiltAngle);
  panServo.write(startPanAngle);

  Serial.begin(9600);
}

void loop() {
  if (buttonPressed) {
    if ((millis() - lastDebounceTime) > debounceDelay) {
      buttonPressed = false;
      myservo.write(targetPosition);
      delay(1500);
      myservo.write(startPosition);
    }
  }

  int tiltJoystickValue = analogRead(tiltJoystickPin);
  int panJoystickValue = analogRead(panJoystickPin);

  unsigned long currentTime = millis();

  if (currentTime - lastUpdateTime >= updateInterval) {
    if (tiltJoystickValue > 650) {
      lastTiltAngle = min(lastTiltAngle + angleChange, maxTiltAngle);
    } else if (tiltJoystickValue < 300) {
      lastTiltAngle = max(lastTiltAngle - angleChange, minTiltAngle);
    }
    tiltServo.write(lastTiltAngle);

    if (panJoystickValue > 650) {
      lastPanAngle = min(lastPanAngle + angleChange, maxPanAngle);
    } else if (panJoystickValue < 300) {
      lastPanAngle = max(lastPanAngle - angleChange, minPanAngle);
    }
    panServo.write(lastPanAngle);

    lastUpdateTime = currentTime;

    Serial.print("Tilt␣angle:␣");
    Serial.print(lastTiltAngle);
    Serial.print(",␣Pan␣angle:␣");
    Serial.println(lastPanAngle);
  }

  delay(15);
}

void onButtonPress() {
  if ((millis() - lastDebounceTime) > debounceDelay) {
    lastDebounceTime = millis();
    buttonPressed = true;
  }
}
```

In the void loop() function, the code is designed to handle the joystick input. The program does nothing when the joystick rests and sends a signal value of 512. However, when the joystick receives a value greater than 650 or less than 300, it triggers the servo to move one increment at a time. This means that for each movement of the joystick beyond the threshold values, the servo will increase or decrease its position by one step. If the joystick is released and returns to the resting position, the servo will remain in its current position without further movement. Furthermore, response and speed control were added for the servo, which can be adjusted under (// Servo speed).

After testing both programs, they were compiled into one program to control the entire SCL from the joystick. The merged code is shown in Listing 6.

Listing 6: SCL joystick controlled program.

```
#include <Servo.h>
// Button and feeding servo configuration
const int buttonPin = 3;
const int feedingServoPin = 11;
const unsigned long debounceDelay = 50;
int startPosition = 0;
int targetPosition = 85;

// Joystick and tilt/pan servos configuration
// Constants for tilt
const int tiltServoPin = 10;
const int tiltJoystickPin = A3;

const int minTiltAngle = 0;
const int maxTiltAngle = 44;
const int startTiltAngle = 5;

// Constants for pan
const int panServoPin = 13;
const int panJoystickPin = A2;

const int minPanAngle = 0;
const int maxPanAngle = 180;
const int startPanAngle = 90;

// ESC configuration
const int ESC_A_Pin = 12;
const int ESC_B_Pin = 9;

const int potAPin = A1;
const int potBPin = A0;

const int ESC_A_Min = 1080; // Min pulse width
const int ESC_A_Max = 1170; // Max pulse width

const int ESC_B_Min = 1130; // Min pulse width
const int ESC_B_Max = 1253; // Max pulse width
const int ESC_Arm_Delay = 2000;

// Servo speed
const unsigned long updateInterval = 20; // 20 milliseconds
const int angleChange = 2;

// Create servo objects
Servo feedingServo;
Servo tiltServo;
Servo panServo;
Servo ESC_A;
Servo ESC_B;

// Variables for button press handling
volatile bool buttonPressed = false;
unsigned long lastDebounceTime = 0;

```

```
// Variables to store the last servo angles
int lastTiltAngle = startTiltAngle;
int lastPanAngle = startPanAngle;

// Variable to store the last update time
unsigned long lastUpdateTime = 0;

void setup() {
  pinMode(buttonPin, INPUT_PULLUP);
  attachInterrupt(digitalPinToInterrupt(buttonPin), onButtonPress,
      FALLING);
  feedingServo.attach(feedingServoPin);
  feedingServo.write(startPosition);

  tiltServo.attach(tiltServoPin);
  panServo.attach(panServoPin);

  tiltServo.write(startTiltAngle);
  panServo.write(startPanAngle);

  ESC_A.attach(ESC_A_Pin);
  ESC_B.attach(ESC_B_Pin);

  // Arm the ESCs
  ESC_A.writeMicroseconds(ESC_A_Min);
  ESC_B.writeMicroseconds(ESC_B_Min);
  delay(ESC_Arm_Delay);

  Serial.begin(9600);
}

void loop() {
  if (buttonPressed) {
    if ((millis() - lastDebounceTime) > debounceDelay) {
      buttonPressed = false;
      feedingServo.write(targetPosition);
      delay(1500);
      feedingServo.write(startPosition);
    }
  }

  int tiltJoystickValue = analogRead(tiltJoystickPin);
  int panJoystickValue = analogRead(panJoystickPin);

  unsigned long currentTime = millis();

  if (currentTime - lastUpdateTime >= updateInterval) {
    if (tiltJoystickValue > 650) {
      lastTiltAngle = min(lastTiltAngle + angleChange, maxTiltAngle);
    } else if (tiltJoystickValue < 250) {
      lastTiltAngle = max(lastTiltAngle - angleChange, minTiltAngle);
    }
    tiltServo.write(lastTiltAngle);

        if (panJoystickValue > 650) {
      lastPanAngle = min(lastPanAngle + angleChange, maxPanAngle);
    } else if (panJoystickValue < 250) {
      lastPanAngle = max(lastPanAngle - angleChange, minPanAngle);
    }
    panServo.write(lastPanAngle);

    lastUpdateTime = currentTime;

    Serial.print("Tilt␣angle:␣");
    Serial.print(lastTiltAngle);
    Serial.print(",␣Pan␣angle:␣");
    Serial.println(lastPanAngle);
  }

  // Read potentiometer values
  int potAValue = analogRead(potAPin);
  int potBValue = analogRead(potBPin);

  // Map potentiometer values to ESC range
  int ESC_A_Value = map(potAValue, 0, 1023, ESC_A_Min, ESC_A_Max);
  int ESC_B_Value = map(potBValue, 0, 1023, ESC_B_Min, ESC_B_Max);

  // Write the mapped values to the ESCs
  ESC_A.writeMicroseconds(ESC_A_Value);
  ESC_B.writeMicroseconds(ESC_B_Value);

  // Print the mapped values for debugging purposes
  Serial.print("ESC␣A␣Value:␣");
  Serial.print(ESC_A_Value);
  Serial.print(",␣ESC␣B␣Value:␣");
  Serial.println(ESC_B_Value);
  delay(15);
}
void onButtonPress() {
  if ((millis() - lastDebounceTime) > debounceDelay) {
    lastDebounceTime = millis();
    buttonPressed = true;
  }
}
```

This program enables joystick-based control of a SCL system. Utilising an Arduino board, it operates various servos for managing tilt, pan, and feeding mechanisms.

Furthermore, two ESCs with potentiometers included to regulate the BLDC speed. The program establishes the necessary components, sets their initial positions, and arms the ESCs. Within the main loop, the code addresses button presses for the feeding mechanism, interprets joystick values to manipulate the tilt and pan servos and maps potentiometer values to the ESCs for speed control, as well as the positions of the pan and tilt servo angles.

With the program to control the SCL in place, was it ready to be tested.

9 Adjustments and Testing

This chapter tests, and adjustments are done on the SCL. The primary goals are to improve the SCL with regards to the objectives 1.3 and goals 1.4.

The process begins with examining the feeding mechanism, highlighting any encountered issues and discussing necessary modifications. As the chapter proceeds, any changes to the design are thoroughly documented, eventually leading to the final presentation of results. These adjustments are crucial to enhancing the overall functionality of the SCL, setting the stage for experimental testing.

9.1 HSE Wheel Cover Implementation

As a cautious step before the testing phase, wheel covers were developed to serve as safety measures. The wheels' potential to reach rotations as high as 10,000 RPM presents a potential risk when in contact with fingers or other objects. Figure 65 displays the design of the wheel cover with ventilation holes.

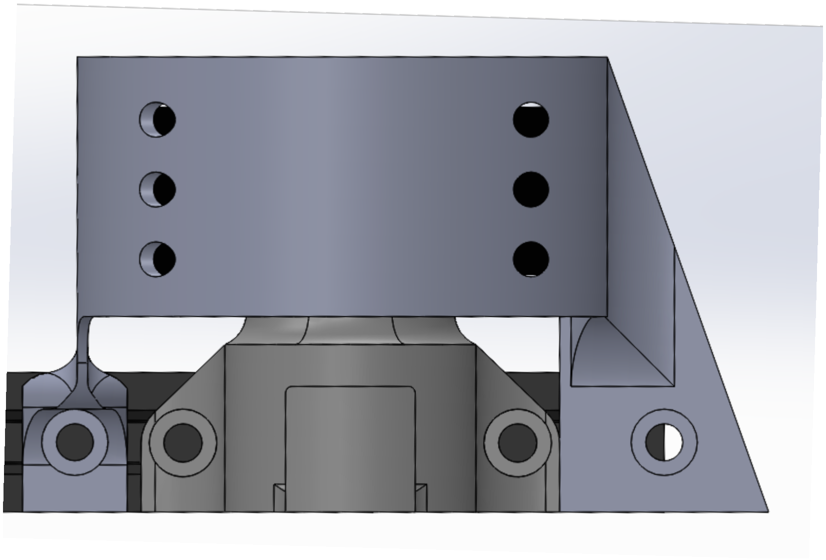


Figure 65: The HSE wheel cover, fitted with BLDC mount at the centre.

It was hypothesised that a cover mounted on a fast-rotating object could cause turbulence. Consequently, the holes were designed to ventilate the air turbulence caused by high wheel rotation. The following section will examine their effectiveness and determine any noticeable impact before further adjustments.

The initial design, as illustrated in Figure 65, included a set of three vertically aligned holes on the side of the cover. These holes were cut in a designed direction to follow the wheel's rotation direction. The design was assumed to minimise the turbulence, despite limited aerodynamics experience.

Furthermore, the holes were also set to a diameter of approximately 5mm, to avoid fingers going through. Moreover, an inclined edge, depicted in Figure 66, was designed along the wheel edge to ensure that the SC remains within the launcher, not on the wheel covers.

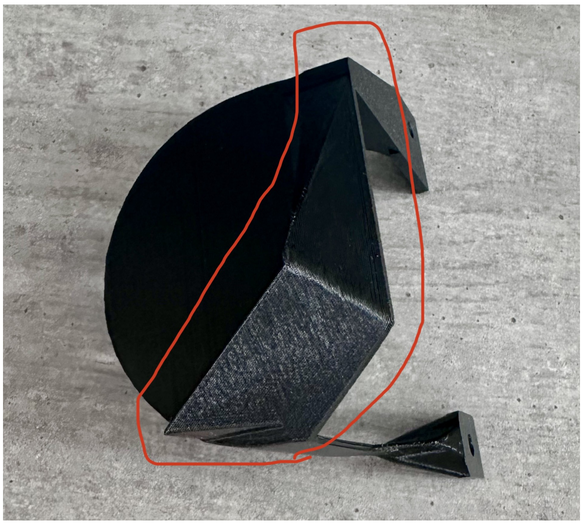


Figure 66: 3D printed HSE wheel cover showcasing a guiding edge.

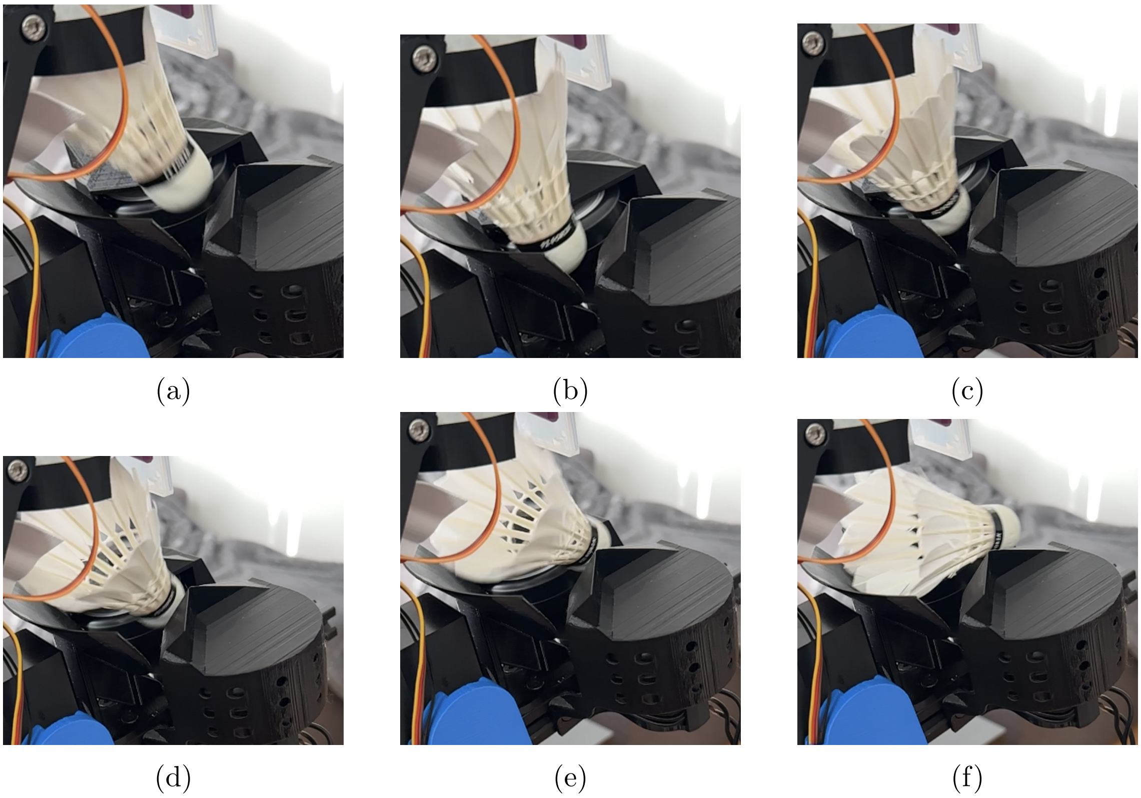
9.2 HSE Wheel Cover Testing and Adjustments

During the initial testing phase, the holes were temporarily sealed with tape to remove their effect. To estimate turbulence, the concept of tufts[6] in aerodynamics was employed. Paper was used as the tuft to evaluate the turbulence at the front, where the SC is launched, and the entrance, where the SC is fed into the wheels. Strong turbulence was indicated when the tuft could not hang straight down but lay vertically.

The method was considered somewhat ineffective when using paper. However, it provided a sufficient indication of air pressure. Upon removing the tape, a decrease in turbulence was observed at the entrance, while the front remained unaffected. As a result, additional holes were added to minimise turbulence as much as possible. The extra pair of holes led to observable turbulence reduction at the entrance.

[6] Tufts in aerodynamics are small streamers or yarns attached to objects or surfaces to visualise the airflow patterns. They are used in wind tunnel testing and experimental aerodynamics to study the behaviour of air around aerodynamic bodies. Tufts provide insights into airflow direction, speed, and stability, aiding design optimisation for improved performance and efficiency [67].

Following SC shots testing, it became evident that the added edge illustrated in Figure 66 was insufficient. Figure 67 shows a slow-motion recording 4.2 of a failed shot. This link shows the video LINK:(slow-motion fail). The SC lands roughly on the wheels, rebounds to the guider (designed to guide the SC from the tube to the wheels), and ricochets over the wheels. The slow-motion analysis also revealed that when the SC enters straight, the feathers could collide with the edge, potentially causing damage or leading to an incorrect trajectory. Therefore, when gripped by the wheels, the HSE wheel cover edge was relocated and flattened to prevent SC contact.



(a) (b) (c)

(d) (e) (f)

Figure 67: Incomplete shot: Missing the wheels.

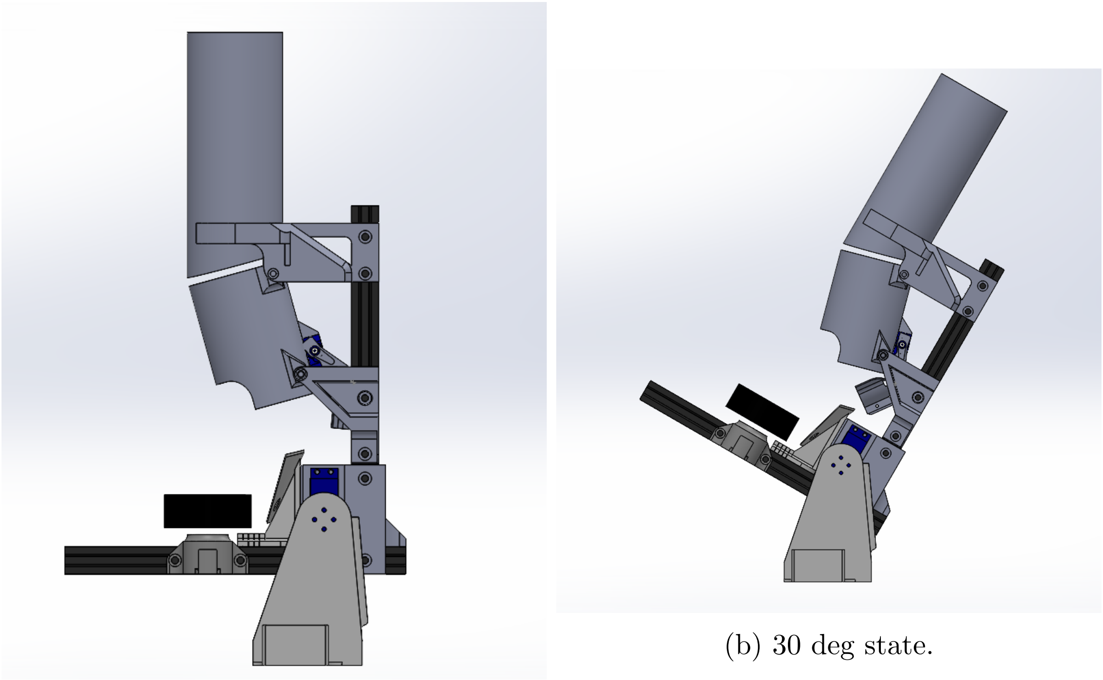
Following the redesign and retesting, the SCL adjustments could commence without the risk of hands getting caught or injured by the wheels. A more comprehensive review of the HSE cover is presented in the construction review section 10.

9.3 Preparations and Initial Adjustments

Before commencing any experimental process, all components of the SCL must be properly set, and initial adjustments must be made. The feeding mechanism, one of the main mechanisms under testing, will be evaluated based on its efficiency in launching SCs without causing double shots, jams, or damage. Based on the objectives 1.3 and goals 1.4 is it undesirable to have jams that stop the SCL. The tests and experimental tests will be carried out across various tilt angles, documenting any variations observed.

However, during the tests with the HSE wheel cover, the FMS digital servo encountered a short circuit. With the procurement of a replacement posing a significant delay, all subsequent testing will proceed using only the SG90 digital servo in the feeding mechanism.

The initial adjustment to the feeding mechanism involved modifying the tube holders to position the tube at a 10-degree angle in the resting state of the SCL. This adjustment aimed to obtain a smoother drop of the SC from the tube to the guider, and eventually to the wheels driven by the BLDC. At a tilt angle of 0 degrees for the SCL, the tube is at an angle of -10 degrees. Conversely, at a 30-degree tilt, the tube adjusts to an additional angle of 20 degrees, as illustrated in Figure 68.



(b) 30 deg state.

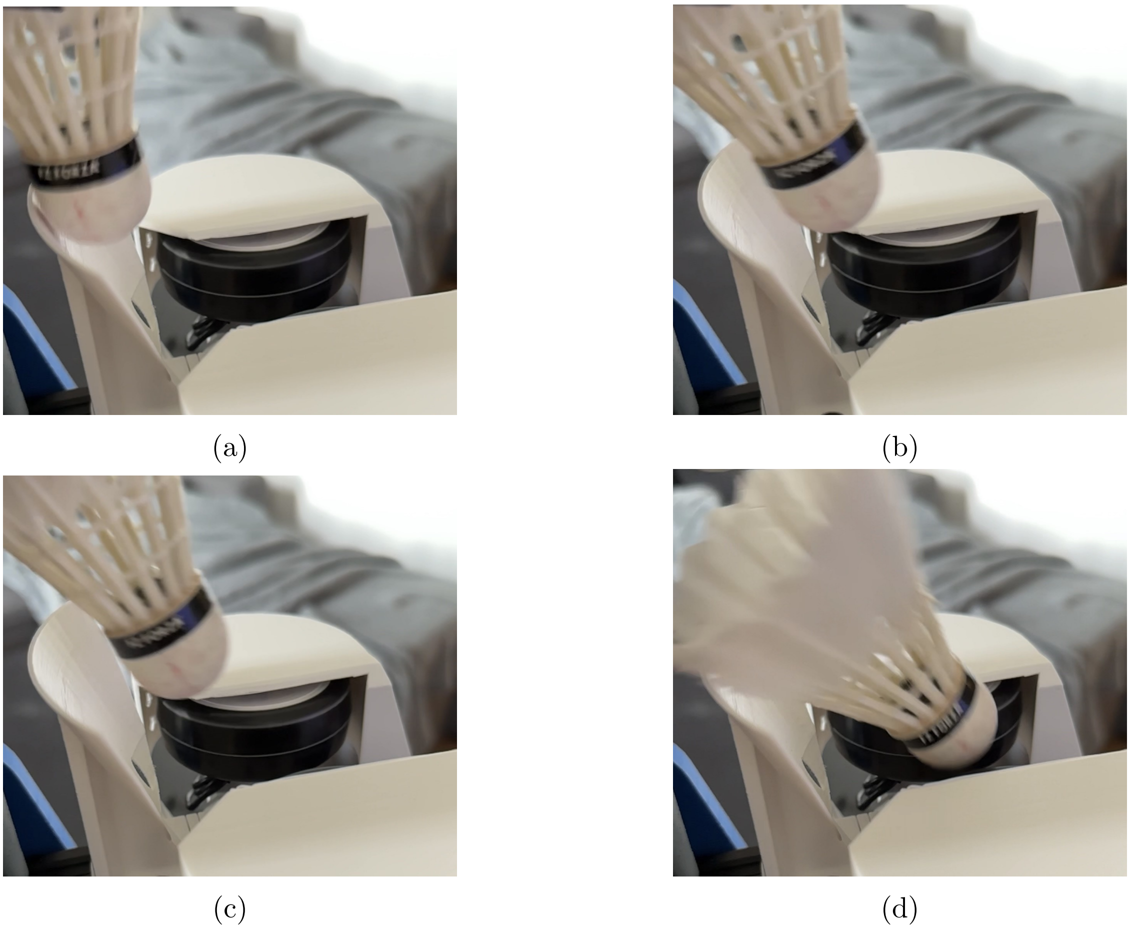
(a) 0 deg state.

Figure 68: The SCL with adjusted tube angle.

Following the tube angle adjustment, a decrease in the instances of jams and mishaps was observed, enabling the continuation of further testing.

9.4 Additional Adjustments and Evaluations

The feeding mechanism was tested similarly with the installed SG90 servo and the re-positioned tube. It was observed during these tests that the drop length might be long in relation to the guider. As illustrated in Figure 69, the SC tends to strike the edge of the guider. Given its velocity, resulting from the drop length, it possesses sufficient energy to bounce over the wheels, landing ahead of the SCL. Such occurrences were repeatedly recorded. Additionally, the guiders mounting position was suspected to be too distant from the wheels, affording the SC space to bounce.



(a)

(b)

(c)

(d)

Figure 69: Fail Shot: Jumps over the wheels.

To address these issues, the bottom tube holders highlighted in Figure 70 were redesigned to reduce the height of the feeding mechanism. Additionally, break-off slots were integrated into the guider to allow adjustments in the space between its end and the wheels without having to print different configurations, as illustrated in Figure 71.

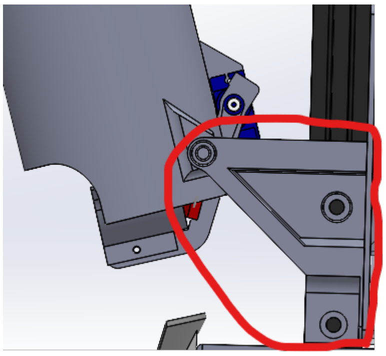


Figure 70: Bottom tube holder.

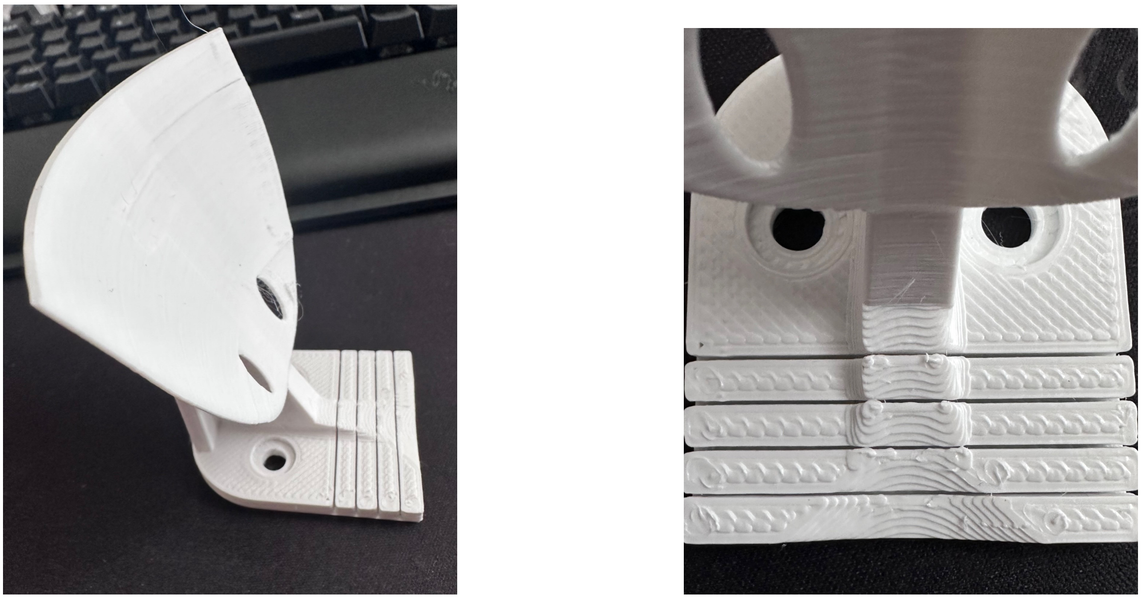
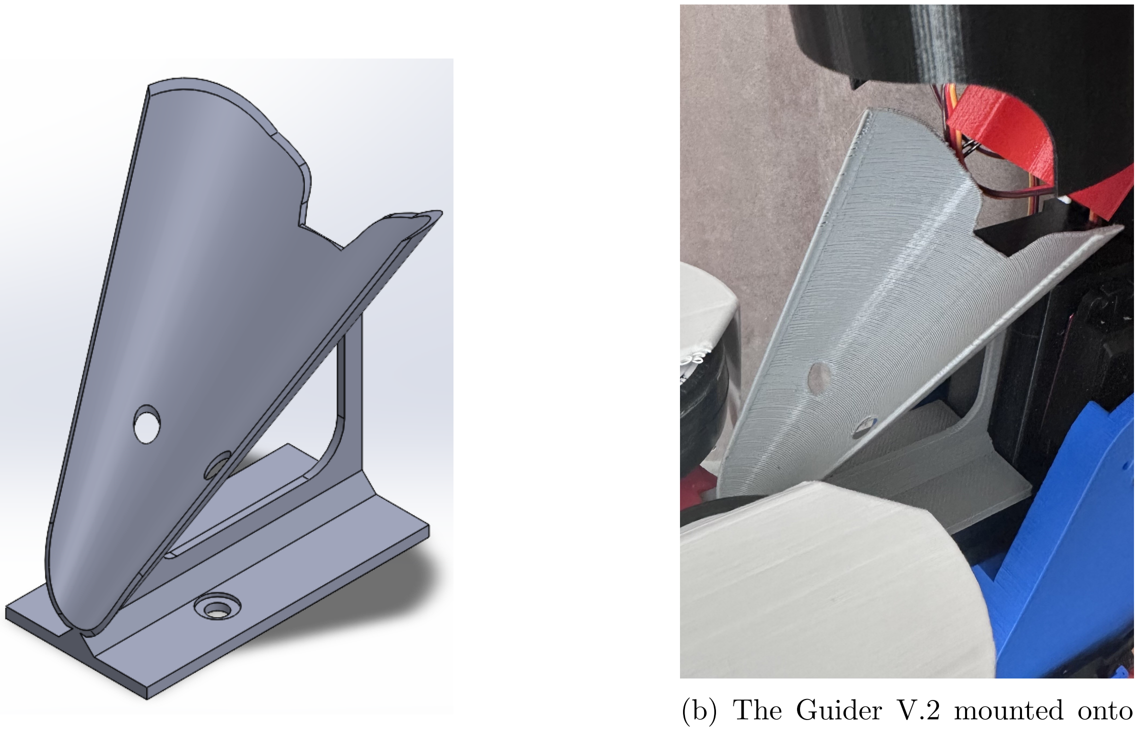


Figure 71: Guider with break-off slots.

No improvement was noticeable after testing from the first to the last break-off slot. Thus, a new design was developed in SW. Three configurations were created, tested, and iterative redesigned until the SC consistently landed accurately on the guider. The final configuration is illustrated in Figure 72, both as a CAD model and installed on the SCL. A in-depth design description and discussion can be found in the construction review section 10.



(a) The Guider V.2.

(b) The Guider V.2 mounted onto the SCL

Figure 72: The Guider V.2.

The revised configurations, which involved lowering and angling the tube and a longer angled guider contributed to a significant reduction in general errors. One final alteration was made in the void loop part of the code, as demonstrated in Listing 7. (This link shows a video).

The modification in the code results in the arm initiating movement from the starting position, proceeding to a 5-degree position throughout 800 milliseconds, and then transitioning to the target position, which prompts the automatic release of the SC. Given that the SC can occasionally hang, a ”bump function” was devised to lift the arm slightly, approximately by 10-15 degrees, ensuring that the needle retains its grip on the SC it is holding. The arm returns to the target position within 1.2 seconds before reverting to its starting position. This alteration promotes a smoother operation, significantly minimising the potential for failures or jams.

The latest version of the program, including recent changes, can be found in the appendix B.

Listing 7: Change in the code for the feed mechanism

```
#include <Servo.h>

// Button and feeding servo configuration
const int buttonPin = 3;
const int feedingServoPin = 11;

const unsigned long debounceDelay = 40;
int startPosition = 0;
int targetPosition = 75;

//The rest is the same
void loop() {
  if (buttonPressed) {
    if ((millis() - lastDebounceTime) > debounceDelay) {
      buttonPressed = false;
      feedingServo.write(startPosition+5);
      delay(800);
      feedingServo.write(targetPosition);
      delay(200);
      feedingServo.write(targetPosition-12);
      delay(200);
      feedingServo.write(targetPosition);
      delay(1200);
      feedingServo.write(startPosition);
    }
  }

// The rest is the same:
```

9.5 Wheel and BLDC Mounts

During each round of testing, alterations were made to the BLDC mounts, primarily to examine the effects of incrementally reducing the distance between the wheels and adjusting the BLDC flang angle. Figure 73 depicts the configurations trialled with the BLDC.

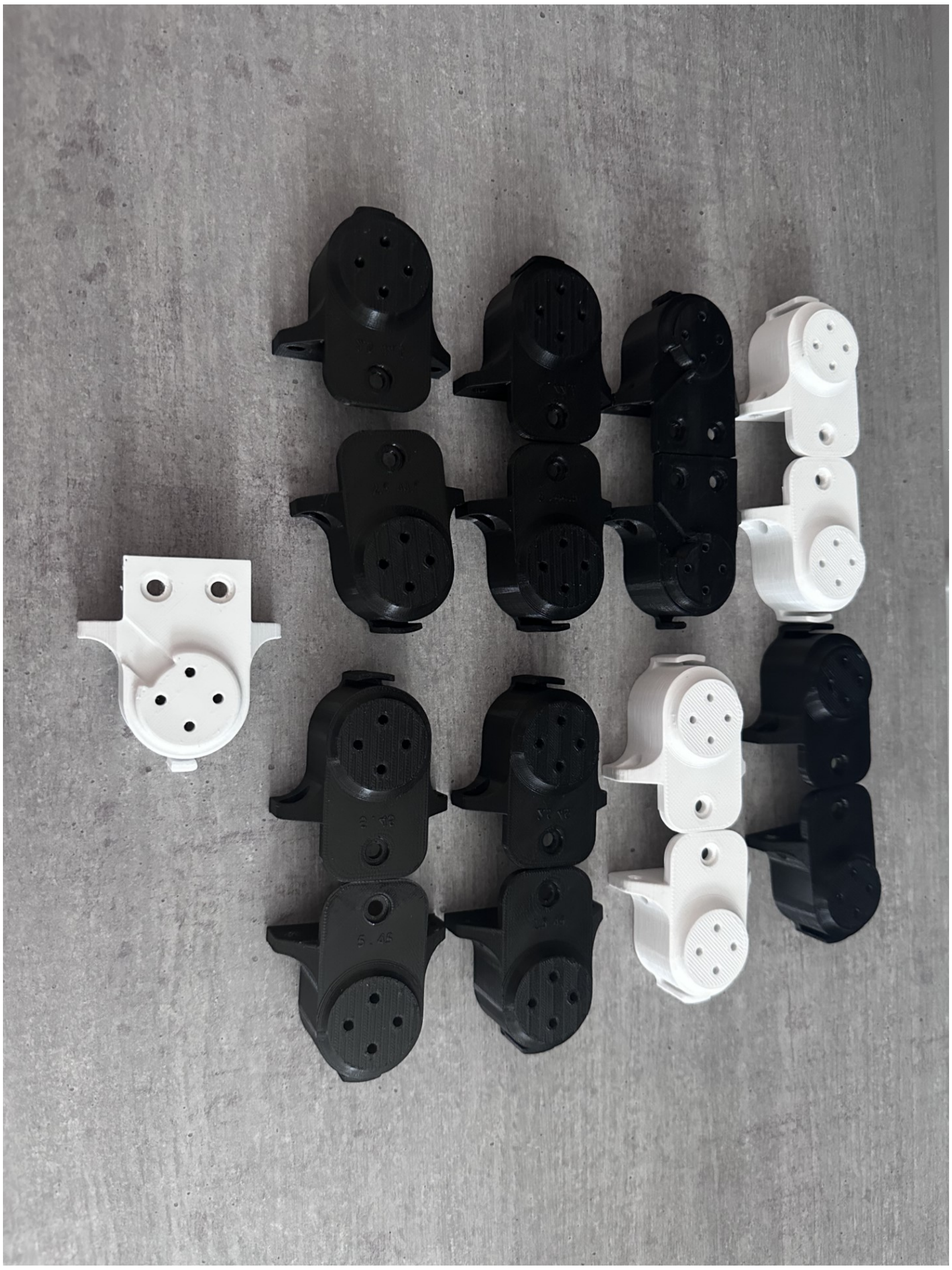


Figure 73: Collection of different configurations.

The initial mounts had a total distance of 45mm between the centre of the aluminium profile and the centre of the BLDC mount. This measurement was derived from the size of a SCs cork head, which is 26mm, and the wheel's diameter of 64mm. Therefore, to maintain a grip on the SC cork, the centre of the wheel and the centre of the BLDC must be positioned such that $\frac{26+64}{2} = 45$ The cork size can range between 25 to 28mm according to [1]. The BLDC was also tested with up to a 5-degree angle. However, this testing was not completed due to necessary modifications to the HSE wheel covers and time constraints.

Through subsequent testing, it was determined that a distance of 43.8 mm was the optimal position to maintain a grip on the SC without causing any damage or deformation to the SC cork. Adjustments were made using the CAD software, with Figure 74 illustrating the parameters that could be changed.

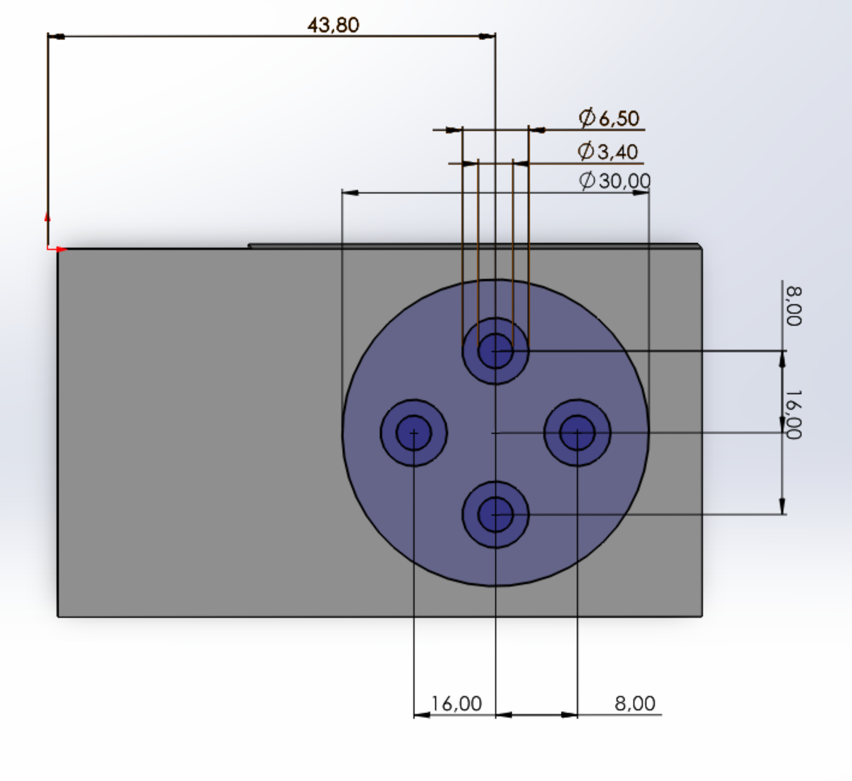


Figure 74: CAD of the BLDC mount parameters in mm.

After all the testing and adjustments had been completed, the construction review could commence once all the parts were specified.

10 Construction review

Before conducting experimental testing, it is necessary to ensure that all design components are specified. This chapter will detail the different components designed to be produced using a 3D printer. Figure 75 provides an overview of the developed parts (references are clickable), which will now be explained in terms of their function, design, and the cause for their specific 3D printing orientations. Figure 76 shows the fully assembled SCL before experimental testing and has been added for a comparison of CAD in Figure 75.

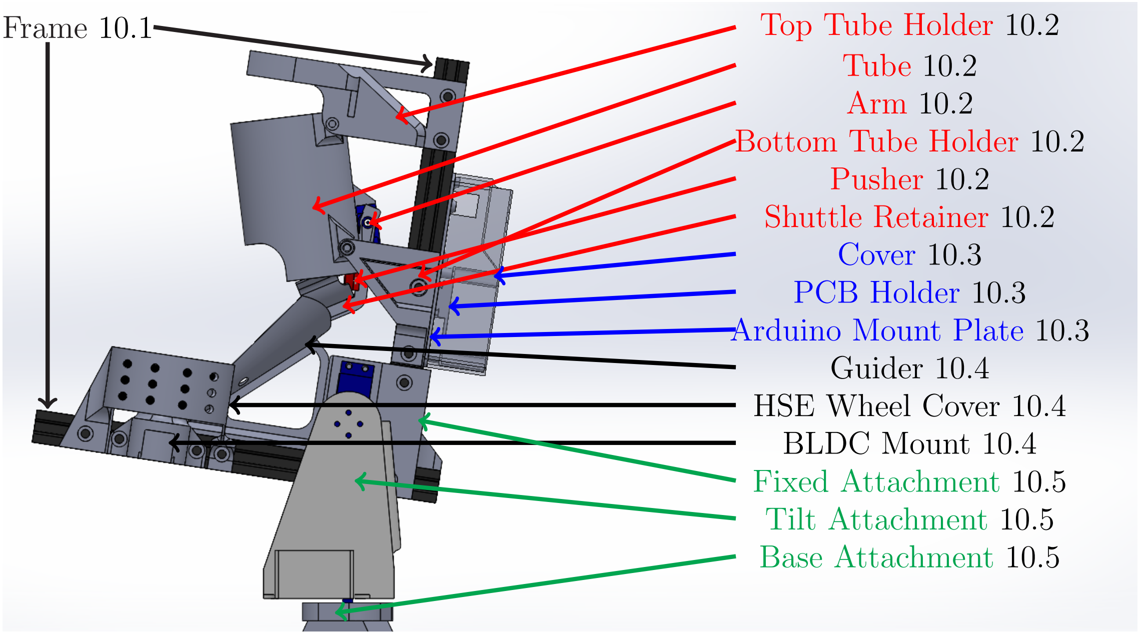


Figure 75: Overview of all printable parts for the SCL.

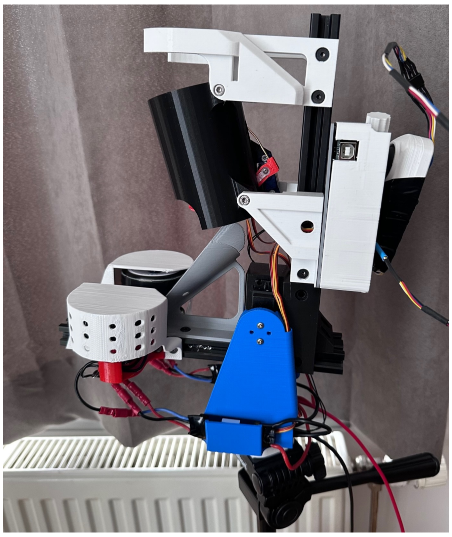


Figure 76: Overview of the SCL assembled.

The design of each component is closely linked to the suggested 3D printing direction, which is essential for optimising the final product's structural integrity and functional efficiency. Various factors, such as mechanical stress, and surface finish requirements, are determined by the specific 3D printing directions.

Subsequently, considering total printing time and required materials, an estimated cost for producing the SCL will be presented. After this chapter, the turn towards to the experimental testing phase, followed by a discussion and conclusions related to the SCL.

10.1 Frame

The SCL frame consists of two V-slot 2040x250mm extruded aluminium profiles provided by RatRig. These profiles have a cross-section of 20x40mm, and the slot feature allows for attachment using specialised nuts. These profiles were selected based on their widespread usage in CNC machines and suitability for the prototype in the pre-project stage [8].

The fixed attachment restrains together both aluminium profiles. Besides the Cover, PCB Holder, and Arduino Mount Plate, most components are attached to the frame using one type of bolt and T-slot nuts. The vertical profile holds the top tube holder, tube, arm, bottom tube holder, and fixed attachment. Meanwhile, the horizontal profile is designed to hold the BLDC Mount, HSE wheel cover, and Guider.

To ensure compatibility and standardising, standard parts from RatRig were used. RatRig also supplies low-profile T-slot screws in M5 size, which have been adopted as the standard for the dimension of all the parts. The M5 screws, specifically M5x8mm in size, were purchased from a supplier to fulfil the thesis's requirements.

A summary of these parts and their associated cost will be presented in Cost Estimation 10.8.

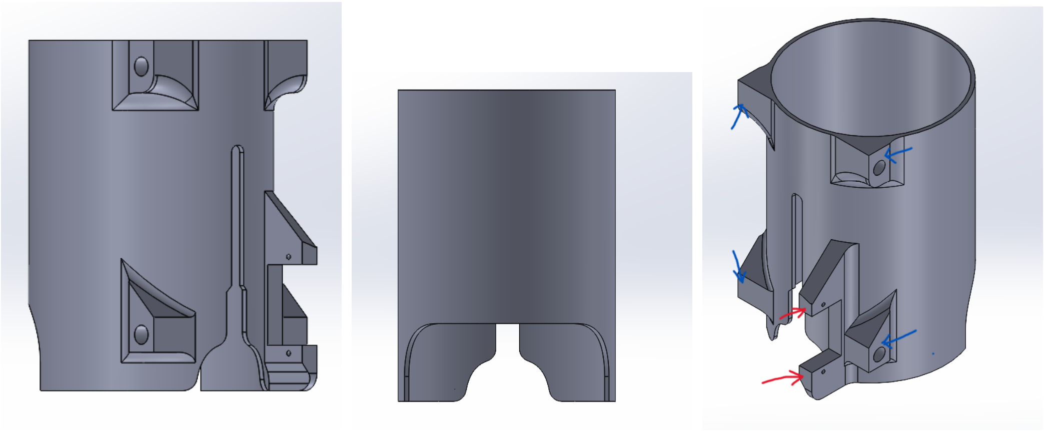
10.2 Feeding Mechanism

Tube

The tube is shaped like a hollow cylinder. The inner diameter of 67mm, and was chosen to provide some friction when the SCs slides down. A SC typically have diameters ranging from 62 to 70mm [1], measured on the outer edge of the feathers.

A larger tube diameter would introduce more slack and inaccuracy, while a narrower tube could cause excessive friction and stick. Therefore, an inner diameter of 67 mm was selected. The outer diameter of the tube is set to 70mm, resulting in a wall thickness of 1.5mm. It is 100mm long to house roughly three SCs.

Figure 77 illustrates the tube in a CAD representation from three perspectives: front, right rear side, and left rear side. The top of the tube, as defined in this context, corresponds to the uppermost part of the figures, while the bottom refers to the lowermost part.



(a) Rear left view.

(b) Front view.

(c) Rear right view.

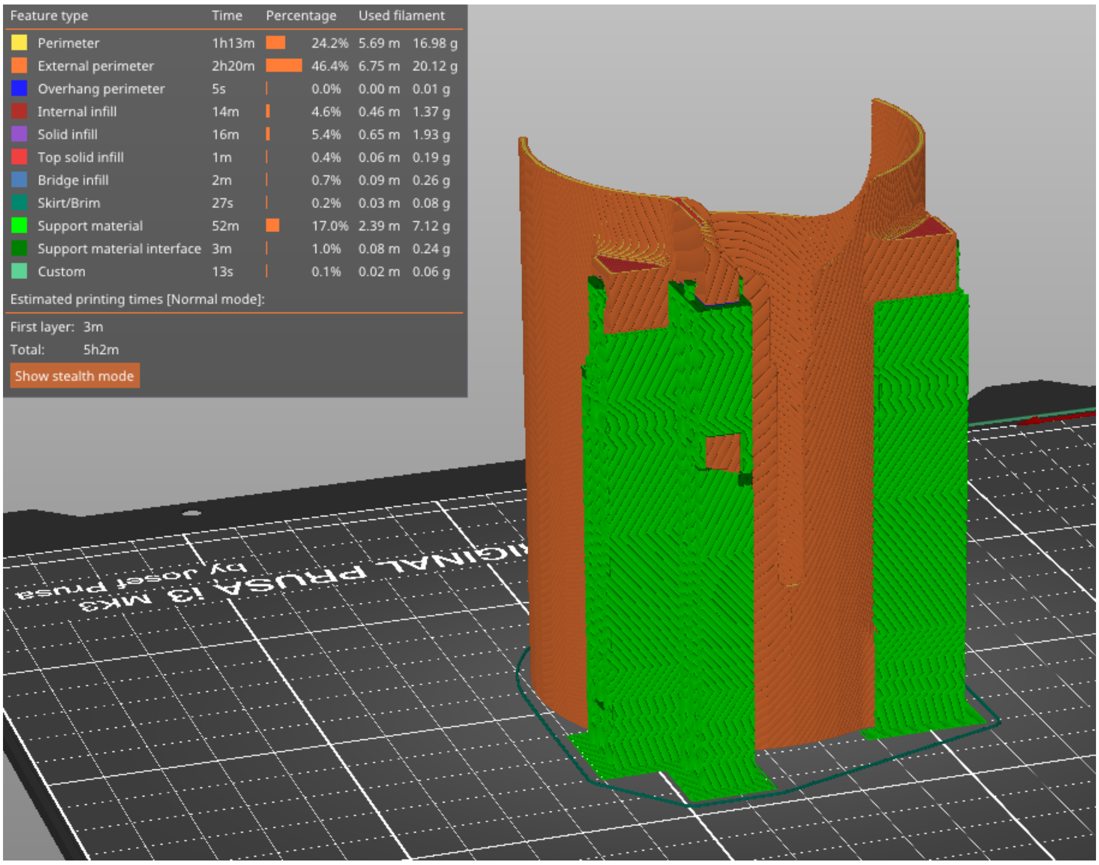
Figure 77: The Tube for the feeding mechanism.

A large cut-out is visible in the front view, allowing the SCs to pass through when the tube is angled and mounted onto the SCL. The smaller cut-outs in the background, also visible in the rear view, are for the shuttle retainer and the arm needle to pass through, and their sizes were determined through testing with the arm.

There are also four protrusions with holes, two visible on the left and right marked with a blue arrow. These protrusions are designed for M4 threaded inserts, which are melted in and used to mount the top and bottom tube holders. These protrusions are symmetrically placed to avoid collisions with the engine mount at the bottom.

The motor mount, marked in red at the bottom tube holding attachment point in the rear right view, has been designed to fit a 9g electric servo motor (commonly known as an SG90 motor). It is positioned so that the arm is centred with respect to the tube's centre when everything is assembled.

Regarding strength and 3D printing, it was designed so that the top should be printed on the bottom. Figure 78 shows how it will look in PrusaSlicer. It is estimated that support must be used due to the motor mount, which will hang 90 degrees out from the tube. No simulations have been made regarding the strength as all weight from the SC is negligible.



Feature type	Time	Percentage	Used filament	
🟨 Perimeter	1h13m	24.2%	5.69 m	16.98 g
🟧 External perimeter	2h20m	46.4%	6.75 m	20.12 g
🟦 Overhang perimeter	5s	0.0%	0.00 m	0.01 g
🟥 Internal infill	14m	4.6%	0.46 m	1.37 g
🟪 Solid infill	16m	5.4%	0.65 m	1.93 g
🟥 Top solid infill	1m	0.4%	0.06 m	0.19 g
🟦 Bridge infill	2m	0.7%	0.09 m	0.26 g
🟩 Skirt/Brim	27s	0.2%	0.03 m	0.08 g
🟩 Support material	52m	17.0%	2.39 m	7.12 g
🟩 Support material interface	3m	1.0%	0.08 m	0.24 g
🟩 Custom	13s	0.1%	0.02 m	0.06 g

Estimated printing times [Normal mode]:

First layer: 3m

Total: 5h2m

Figure 78: The Tube sliced in PrusaSlicer.

However, there is a requirement that the surface on the inside is ”smooth” so that a SC don’t stick when it moves up and down. To achieve this surface finish, the layer height can be lowered, the z-hop can be turned off, or surface treatment techniques such as sandpaper can be used.

Top Tube Holder

The top tube holder is mounted between the vertical aluminium profile and at the top of the Tube. The blue arrows shown in Figure 79 (a) represent the M5 screw holes from RatRig, securely holding the top tube holder onto the vertical aluminium profile. The red arrow in the same figure indicates the mounting point for an M4 screw on the top of the Tube. All screw holes are countersunk for a finer finish. The green arrow in Figure 79 (b) indicates the surface where the vertical aluminium profile is supported against when mounted.

The top tube holder has a geometric height of 43mm, a total width of 80.7mm, and a total length of 133.9mm. A truss structure has been used to provide the strength needed to hold the tube and a magazine housing SCs by passing it through the hole depicted in Figure 79 (c).

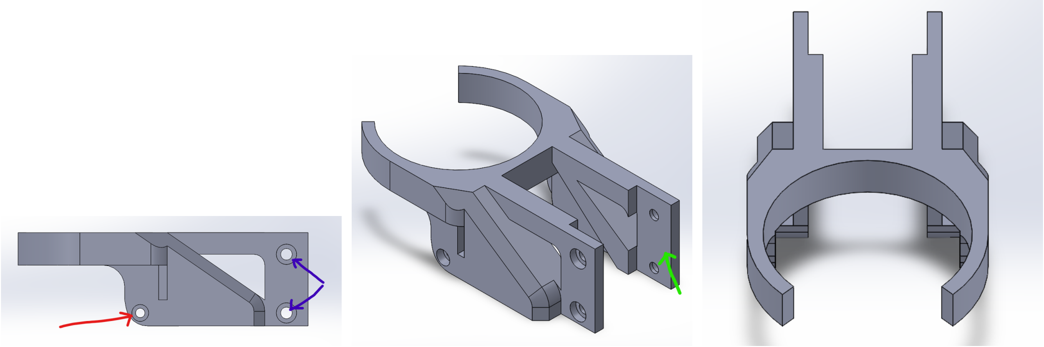
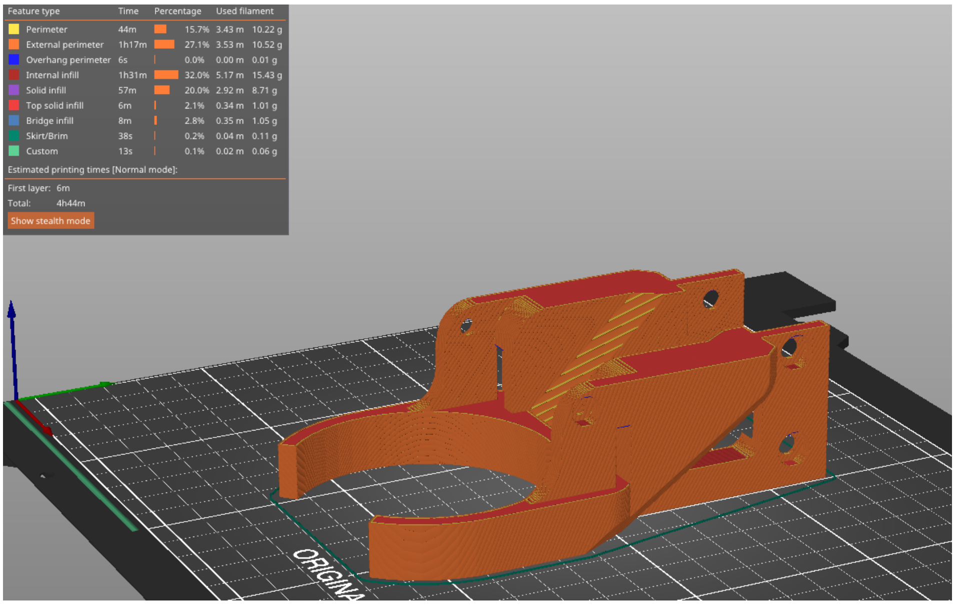

(a) Side view.
(b) Side rear view.
(c) Top front view.

Figure 79: The top tube holder.

It should be noted that the top tube holder served as an inspiration for how a larger magazine could be placed on top, but no further development was pursued in that direction.

No strength test simulations were conducted. However, the design of the top tube holder is intended to withstand pressure stress along the truss structure resulting from forces applied from the top. It is important to note that the 3D-printed lamina is the weakest matter in strength, particularly in a stretch tensile context. To ensure optimal functionality and structural integrity, it is recommended to 3D print the top tube holder as depicted in Figure 80, with the top facing down on the printing plate. This orientation allows the lamina to lie horizontally, distributing pressure evenly across all the lamina until it reaches the vertical profile.



Feature type	Time	Percentage		Used filament	
🟨 Perimeter	44m	▬	15.7%	3.43 m	10.22 g
🟧 External perimeter	1h17m	▬	27.1%	3.53 m	10.52 g
🟦 Overhang perimeter	6s	|	0.0%	0.00 m	0.01 g
🟥 Internal infill	1h31m	▬	32.0%	5.17 m	15.43 g
🟪 Solid infill	57m	▬	20.0%	2.92 m	8.71 g
🟥 Top solid infill	6m	|	2.1%	0.34 m	1.01 g
🟦 Bridge infill	8m	|	2.8%	0.35 m	1.05 g
🟩 Skirt/Brim	38s	|	0.2%	0.04 m	0.11 g
🟩 Custom	13s	|	0.1%	0.02 m	0.06 g

Estimated printing times [Normal mode]:

First layer: 6m

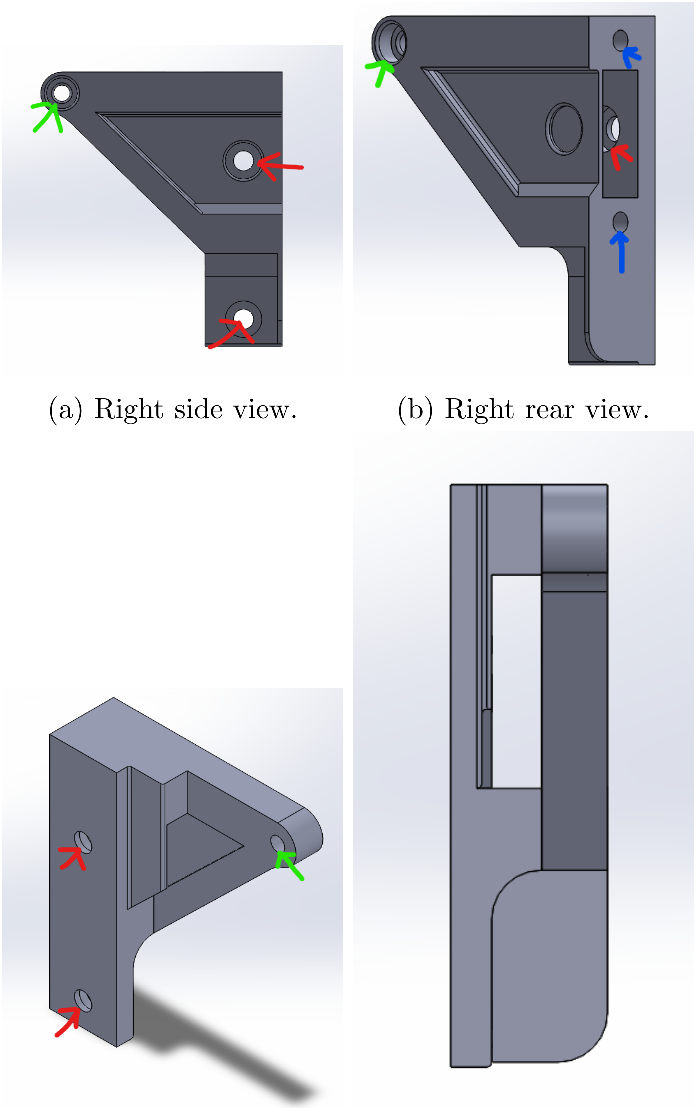
Total: 4h44m

Figure 80: The top tube holder sliced in PrusaSlicer.

Bottom Tube Holder

The bottom tube holder is mounted between the vertical aluminium profile and the bottom of the Tube. Its primary function is to support the tube from the bottom, hold the Arduino mount plate at the rear, and serve as a cable channel. Geometrically, it has a total height of 71mm, a total length of 64.4mm, and a thickness of 19.1mm. Its design uses trusses, with cover-to-cover cables passing through, as depicted in Figure 81 (a) (d).

Figure 81 illustrates the bottom tube holder from different perspectives. The red arrows indicate the mounting points for M5 screws, while the green arrow represents the placement of an M4 screw for the Tube. All screw holes are countersunk for a smoother finish. Additionally, the blue arrows indicate where M3 thread inserts will be mounted. The thread inserts are positioned on the same surface where the Arduino Mount plate lies flat and will be parallel to the rear of the vertical aluminium profile.



(a) Right side view.

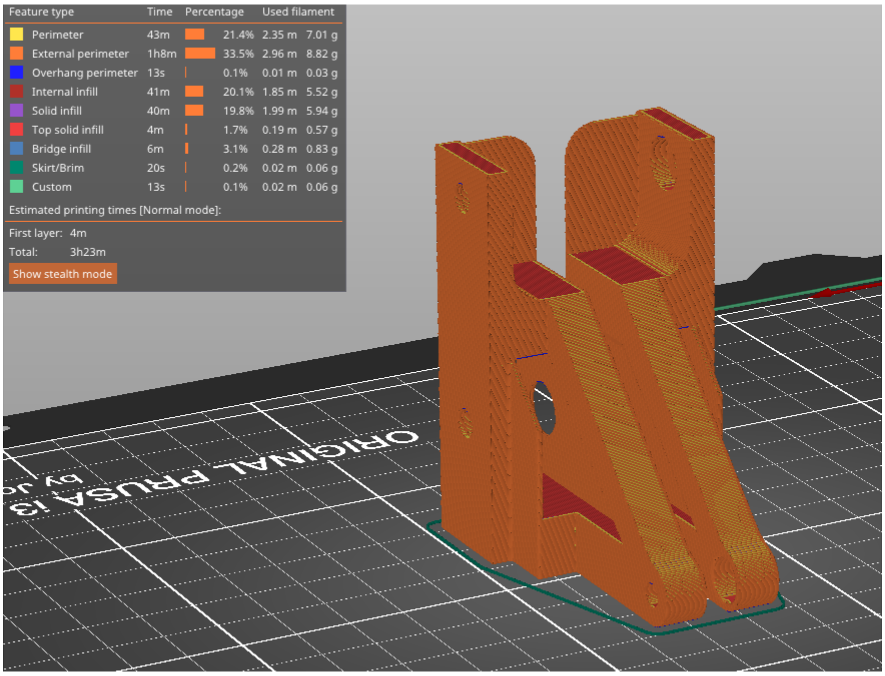
(b) Right rear view.

(c) Left front view.

(d) Front view.

Figure 81: The Bottom Tube Holder.

Like the top tube holder, the bottom tube holder should be printed in a specific direction to achieve optimal strength. No simulations have been conducted on this component, as negligible loads from the tube are assumed. As shown in Figure 82, two components must be printed for each side of the vertical aluminium profile. In PrusaSlicer, the model can be mirrored by right-clicking and selecting the option to mirror it about the axis from which it was inserted. Additionally, Figure 82 clarifies the recommended printing orientation to maximise the lamina's pressure force on the lamina.



Feature type	Time	Percentage	Used filament	
🟨 Perimeter	43m	21.4%	2.35 m	7.01 g
🟧 External perimeter	1h8m	33.5%	2.96 m	8.82 g
🟦 Overhang perimeter	13s	0.1%	0.01 m	0.03 g
🟥 Internal infill	41m	20.1%	1.85 m	5.52 g
🟪 Solid infill	40m	19.8%	1.99 m	5.94 g
🟥 Top solid infill	4m	1.7%	0.19 m	0.57 g
🟦 Bridge infill	6m	3.1%	0.28 m	0.83 g
🟩 Skirt/Brim	20s	0.2%	0.02 m	0.06 g
🟩 Custom	13s	0.1%	0.02 m	0.06 g

Estimated printing times [Normal mode]:

First layer: 4m

Total: 3h23m

Figure 82: The bottom tube holder sliced in PrusaSlicer.

The bottom tube holder can be printed with the rear side facing down; however, in this instance, the orientation of the lamina would be vertical, as represented by the thick red line in Figure 83. This orientation might cause the lamina to tear under load and potentially break if the adhesion between lamina layers is poor, given that the lamina would be positioned along the assumed direction of the force.

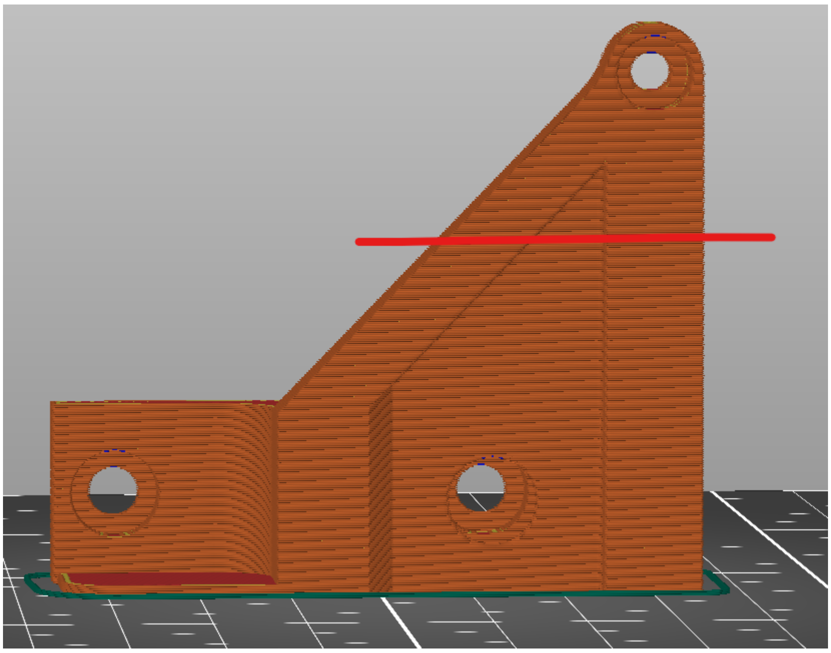


Figure 83: Example of the bottom tube holder sliced in PrusaSlicer

Arm

The arm is mounted on the servo motor located on the tube. It has a J-shaped design with a total length of approximately 80mm, a thickness of 6mm, and a width of approximately 8mm. The arm comprises four functional surfaces, each serving a specific purpose. They are described as follows:

- Top Surface (marked with a red arrow in Figure 84): This surface features a hole where the needle is melted and secured. It provides a stable attachment point for the needle, ensuring proper positioning during operation.
- Slot Surface (marked in orange in Figure 84): The arm incorporates a slotted housing designed to accommodate single-arm push-on accessories for SG90 servo motors.
- Pusher Surface (marked in blue in Figure 84): Positioned before the bend in the bottom, this surface includes a hole for mounting the pusher and a notch that serves the same function as a pusher. The pusher plays a critical role in pushing the SC away from the cut-out in the Tube.
- Shuttle Retainer Hole Surface (marked in green in Figure 84): This surface contains a hole designed to secure the shuttle retainer. The shuttle retainer can be firmly tightened onto the surface and a screw into the hole, ensuring a secure and stable connection.

These functional surfaces enable the arm to handle and precisely manipulate the SC. The arm and shuttle retainer will be presented in more detail later.

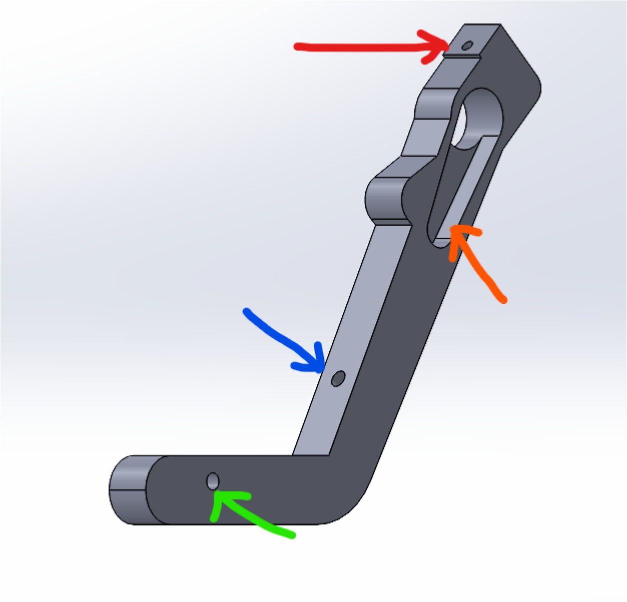
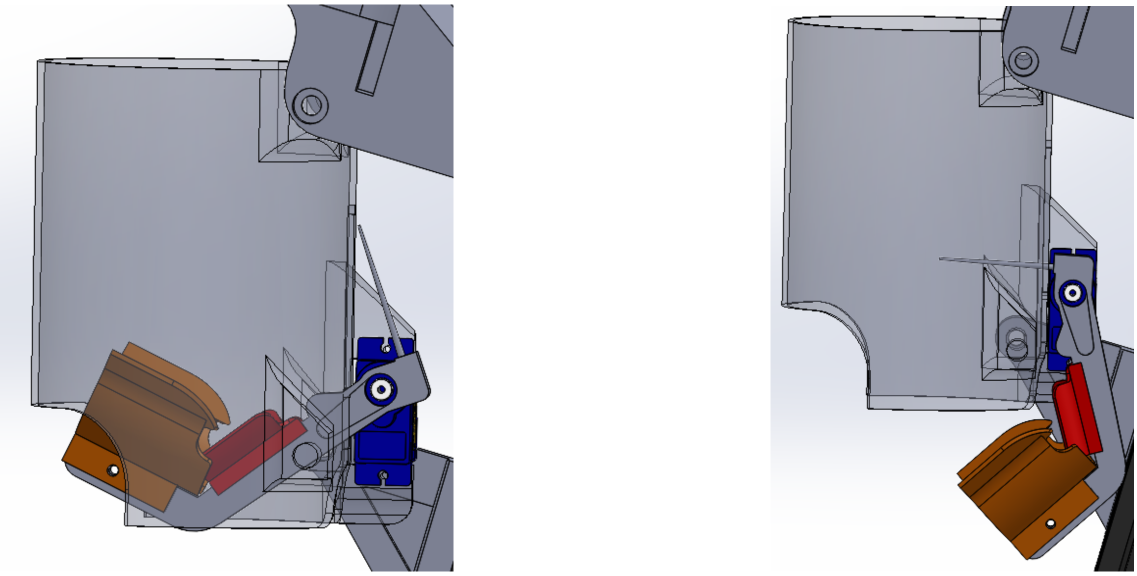


Figure 84: The Arm.

The arm is a dynamic component with two positions: the start and end positions, as illustrated in Figure 85.



(a) Start position.

(b) End position.

Figure 85: Two stages of the arm.

In the start position, the arm holds the SC, preventing it from falling through the tube, with the assistance of the shuttle retainer.

In the end position, the needle penetrates between the SC feathers to be fed. The needle hits between the stings on the SC that are not supposed to fall and bears the weight of the remaining SCs above, while the shuttle retainer guides the SC in the desired direction.

The needle is made from a 1.6mm thick binder's steel string. It is cut to a length of 35mm, with 5mm of the end heated and melted into the hole on the top surface. An additional 18mm extends perpendicularly from the top surface. At the same time, the remaining portion is bent at a 5-degree angle in the direction of the shuttle retainer and pusher, as shown in Figure 86.

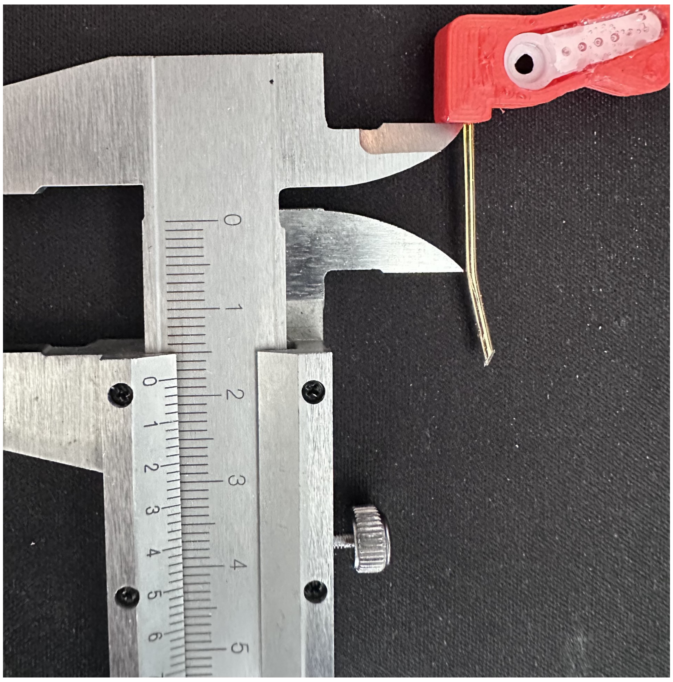
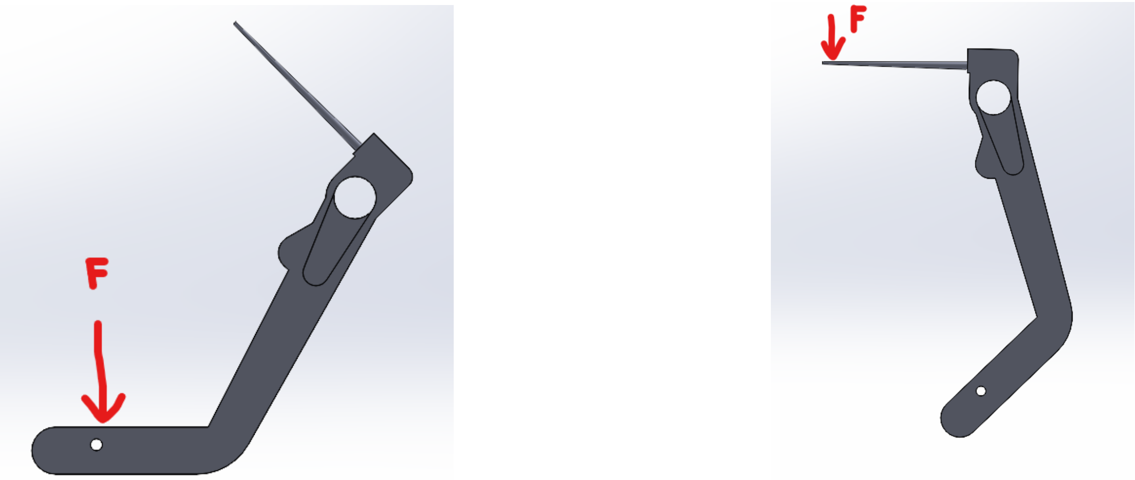


Figure 86: Length of the needle on the Arm.

During the testing phase, it was observed that the bent angle on the needle allowed for smoother movement of the arm and the SC when transitioning from the end position back to the start position.

Figure 87 illustrates a simplified representation of the forces involved in the start and end positions. In the start position, it is crucial to note that the lamina's weakest point would be if it were oriented perpendicular to the force within the same plane and in the same direction as the applied force.

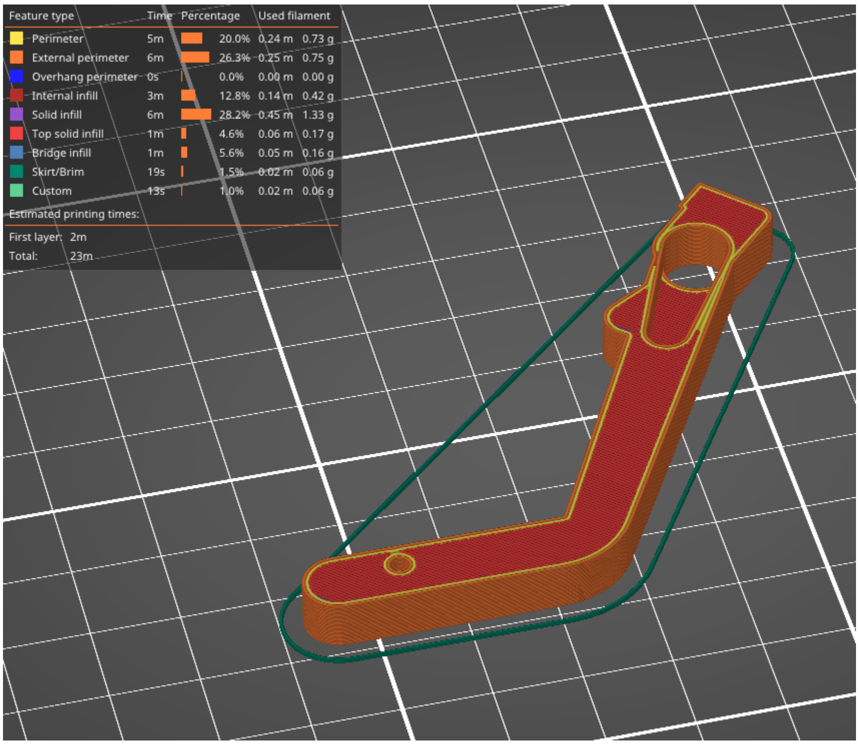
To ensure optimal strength, it is recommended to print the arm in alignment with the indicated direction shown in Figure 88. This orientation will maximise the lamina's structural integrity. No simulations were conducted for this component as the testing phase provided satisfactory results and necessary information.



(a) Start position.

(b) End position.

Figure 87: Two stages of the arm with force simplification.



Feature type	Time	Percentage		Used filament	
🟨 Perimeter	5m	▬	20.0%	0.24 m	0.73 g
🟧 External perimeter	6m	▬	26.3%	0.25 m	0.75 g
🟦 Overhang perimeter	0s			0.0%	0.00 m	0.00 g
🟥 Internal infill	3m	▬	12.8%	0.14 m	0.42 g
🟪 Solid infill	6m	▬	28.2%	0.45 m	1.33 g
🟥 Top solid infill	1m	▬	4.6%	0.06 m	0.17 g
🟦 Bridge infill	1m	▬	5.6%	0.05 m	0.16 g
🟩 Skirt/Brim	19s			1.5%	0.02 m	0.06 g
🟩 Custom	13s			1.0%	0.02 m	0.06 g

Estimated printing times:

First layer: 2m

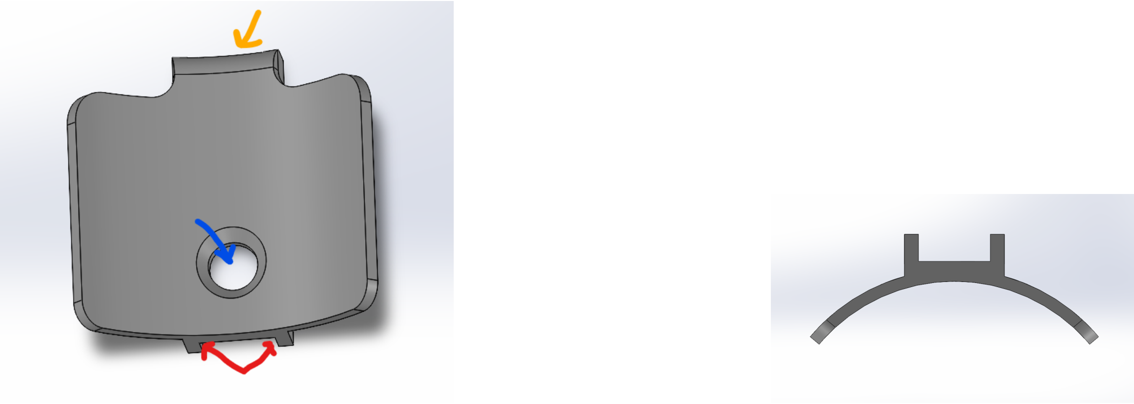
Total: 23m

Figure 88: Example of the arm sliced in PrusaSlicer.

Pusher

The Pusher's role is to push the SC from the tube's cut-out. It is approximately 24mm high, 25mm wide, and 3mm thick.

Figure 89a shows the pusher, where a red arrow highlights the section that should be mounted against the arm. The blue arrow indicates the location of the screw hole, and an orange arrow points towards the transition point from the pusher to the notch, which lies on the same surface on the arm. This transition is seamless, and the pusher itself is designed with a curved shape, featuring two slots that fit the thickness of the arm, as shown in Figure 89b.



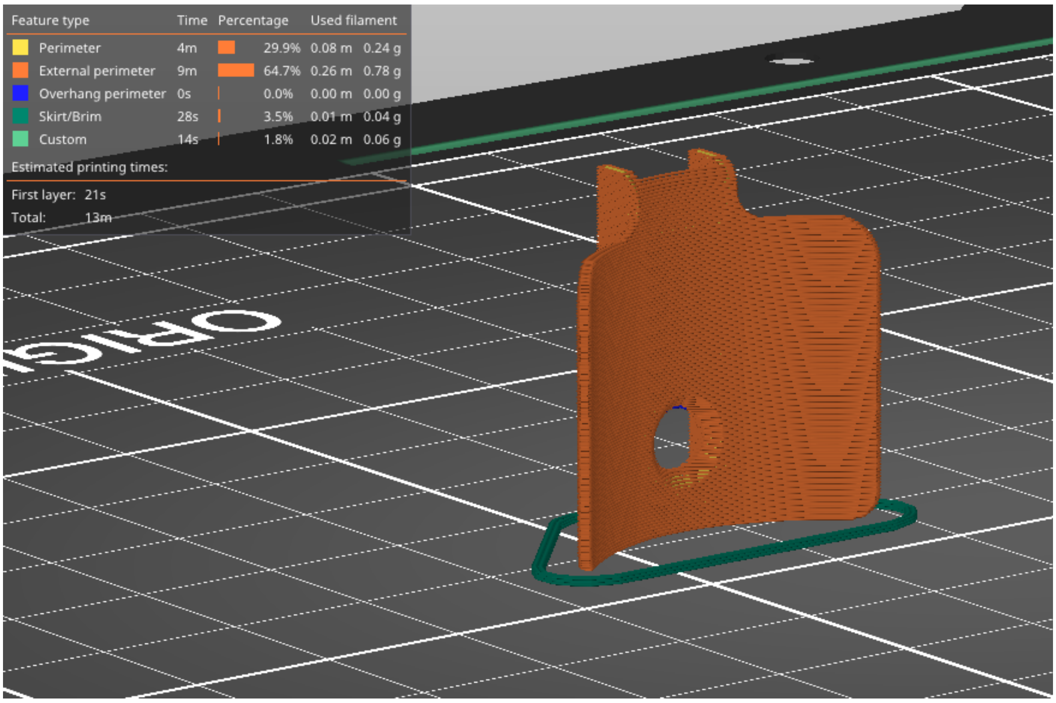
(a) Front view.

(b) Side view.

Figure 89: The pusher.

No simulations were conducted for this component as the testing phase provided satisfactory results and necessary information.

As for 3D printing, although there is no specific required direction, the most efficient way is illustrated in Figure 90. This orientation avoids the need for a support structure, reducing material waste.



	Feature type	Time	Percentage		Used filament	
🟨	Perimeter	4m	▬	29.9%	0.08 m	0.24 g
🟧	External perimeter	9m	▬	64.7%	0.26 m	0.78 g
🟦	Overhang perimeter	0s	ǀ	0.0%	0.00 m	0.00 g
🟩	Skirt/Brim	28s	ǀ	3.5%	0.01 m	0.04 g
🟩	Custom	14s	ǀ	1.8%	0.02 m	0.06 g

Estimated printing times:

First layer: 21s

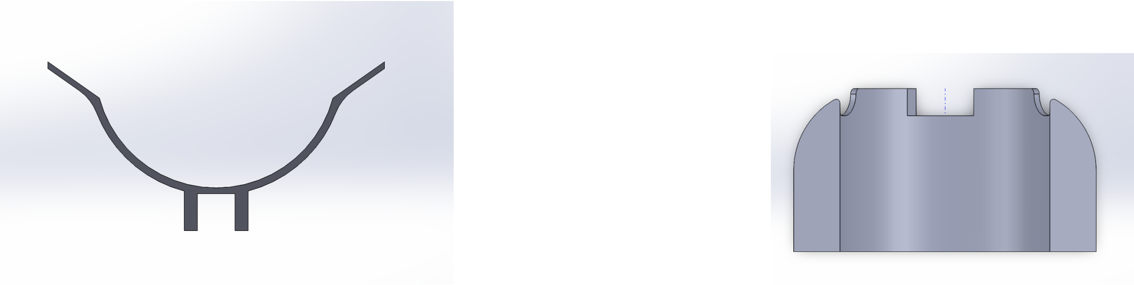
Total: 13m

Figure 90: Example of the pusher sliced in PrusaSlicer.

Shuttle Retainer

The shutter retainer, placed at the bottom of the arm mechanism, is responsible for retaining the SC and guiding it appropriately towards the wheels as the arm rotates. It achieves this through a semi-circular design of uniform thickness, as shown in Figure 91a. The design features an inner radius of 20mm to hold the largest corks and extends outwards to a wingspan of 10.4mm. The extended wings offer a larger coverage area within the tube, ensuring that the descending SC is centralised at the bottom of the shutter retainer.

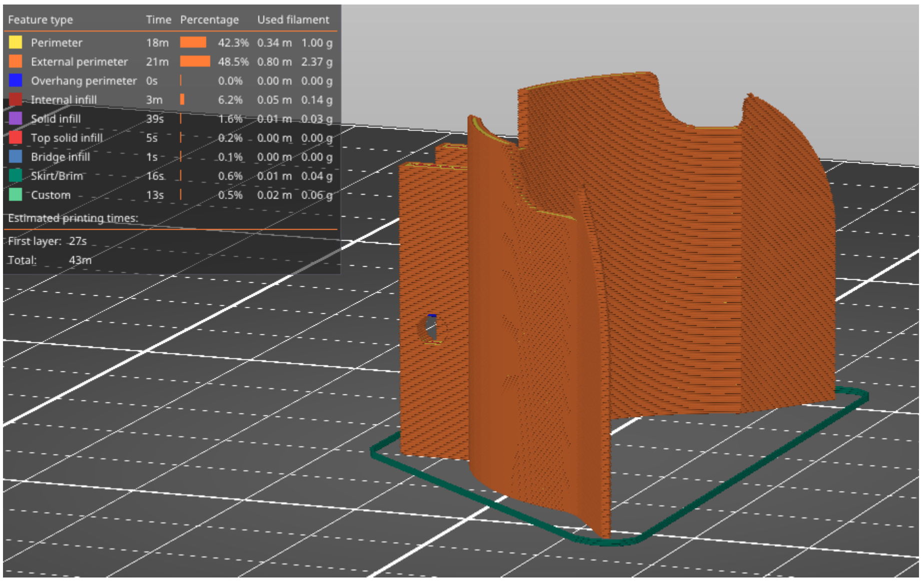
Figure 91b highlights a cutout in the shutter retainer's centre, designed to prevent collision with the pusher. The retainer's sides are curved inward, providing a recess that enables the retainer to pass under the tube without collision.



(a) (b)

Figure 91: The shuttle retainer.

No simulations were conducted because the shutter retainer, like the pusher, is not subjected to significant forces. However, a preferred 3D printing direction is proposed in Figure 92 to avoid the use of additional support material.



Feature type		Time	Percentage	Used filament	
🟨	Perimeter	18m	42.3%	0.34 m	1.00 g
🟧	External perimeter	21m	48.5%	0.80 m	2.37 g
🟦	Overhang perimeter	0s	0.0%	0.00 m	0.00 g
🟫	Internal infill	3m	6.2%	0.05 m	0.14 g
🟪	Solid infill	39s	1.6%	0.01 m	0.03 g
🟥	Top solid infill	5s	0.2%	0.00 m	0.00 g
🟦	Bridge infill	1s	0.1%	0.00 m	0.00 g
🟩	Skirt/Brim	16s	0.6%	0.01 m	0.04 g
🟩	Custom	13s	0.5%	0.02 m	0.06 g

Estimated printing times:

First layer: 27s

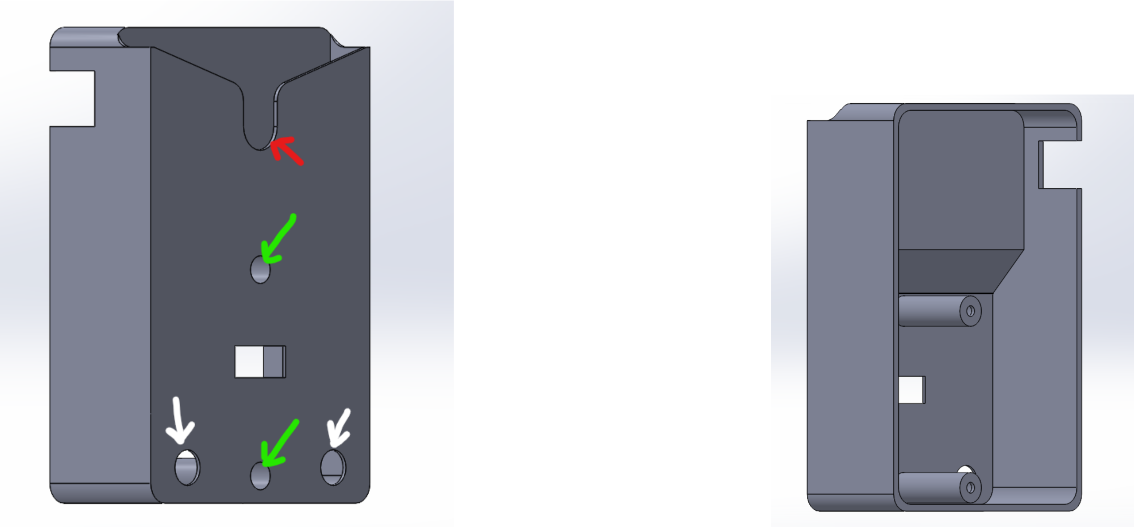
Total: 43m

Figure 92: Example of the shuttle retainer sliced in PrusaSlicer.

10.3 Microcontroller Holder

Cover

The cover is intended to shield the PCB, potentiometer, and microcontroller. Its dimensions are 120mm tall, 78.2mm wide, and 36mm thick. Figure 93a shows the front view of the cover, with the green arrows indicating where the M3 screws are inserted to secure the cover to the Arduino mount plate. Additional openings are provided to adjust speeds using a flathead screwdriver. At the top, a pocket (indicated by the red arrow) is designed to house a joystick and a screwdriver, as shown to the right in Figure 76. Figure 93b shows the interior of the cover.

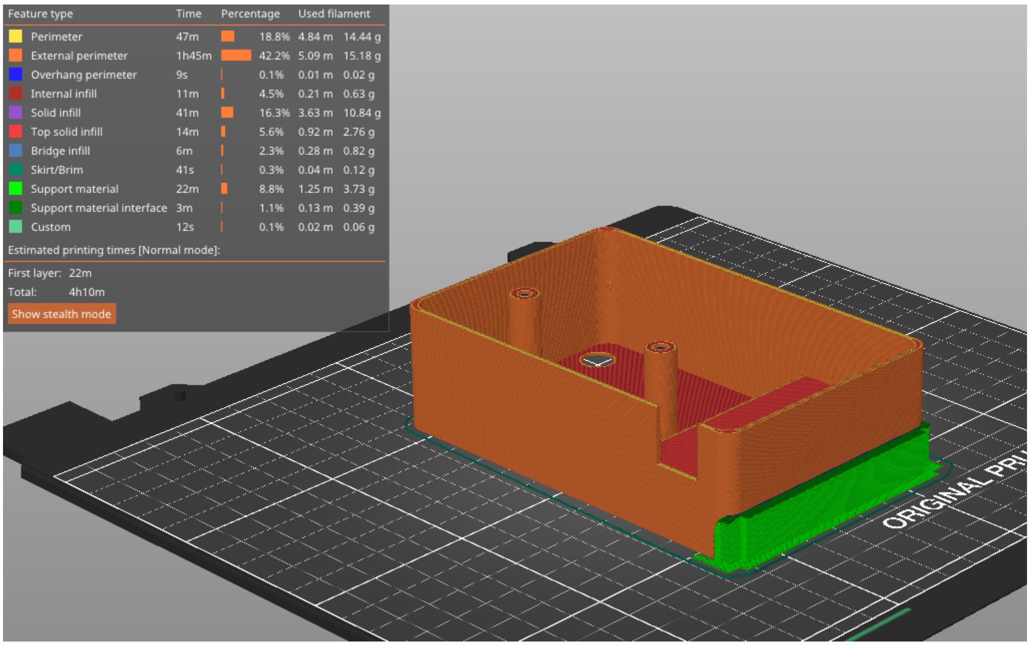


(a) Front View.

(b) Inside View.

Figure 93: The cover.

The cover is designed to be 3D printed with the front side placed against the printer bed, as shown in Figure 94, requiring support material from the build surface.



Feature type	Time	Percentage		Used filament	
🟨 Perimeter	47m	▬	18.8%	4.84 m	14.44 g
🟧 External perimeter	1h45m	▬	42.2%	5.09 m	15.18 g
🟦 Overhang perimeter	9s	▏	0.1%	0.01 m	0.02 g
🟥 Internal infill	11m	▏	4.5%	0.21 m	0.63 g
🟪 Solid infill	41m	▬	16.3%	3.63 m	10.84 g
🟥 Top solid infill	14m	▏	5.6%	0.92 m	2.76 g
🟦 Bridge infill	6m	▏	2.3%	0.28 m	0.82 g
🟩 Skirt/Brim	41s	▏	0.3%	0.04 m	0.12 g
🟩 Support material	22m	▏	8.8%	1.25 m	3.73 g
🟩 Support material interface	3m	▏	1.1%	0.13 m	0.39 g
🟩 Custom	12s	▏	0.1%	0.02 m	0.06 g

Estimated printing times [Normal mode]:

First layer: 22m

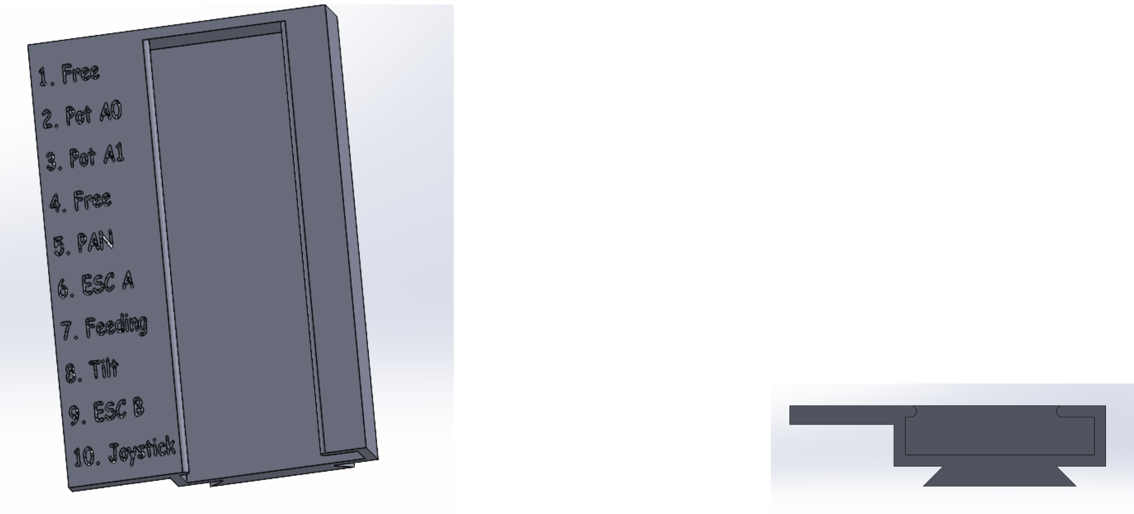
Total: 4h10m

Show stealth mode

Figure 94: Example of the cover sliced in PrusaSlicer.

Pcb Holder

The PCB holder is designed to secure the PCB. As depicted in Figure 95, the holder is 50mm long, 33.4mm wide, and 8.5mm thick. It features a wall at one end and an opening at the other for PCB insertion. The holder is meant to slide into the Arduino mount plate using the lower part of the triangle shown in Figure 95b. Figure 95a shows the PCB holder with marked inputs for the PCB made in this thesis.

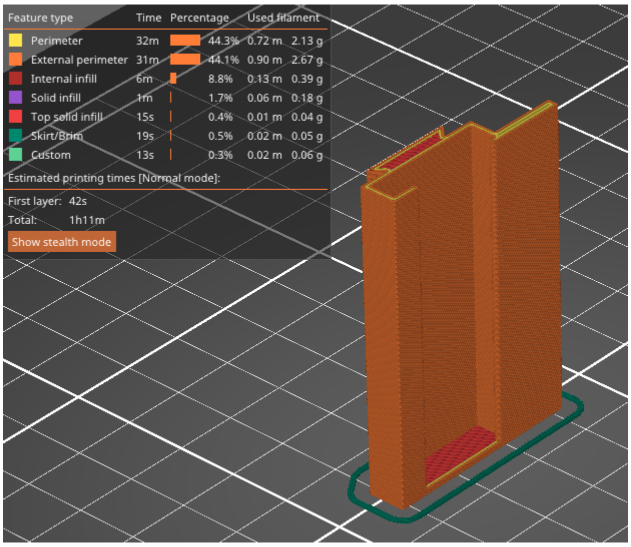


(a) Front view.

(b) Side view.

Figure 95: The PCB holder.

For 3D printing, text integration requires a 0.4 mm nozzle or lower for the 3D printer. To avoid this requirement, the uploaded CAD and STL file does not include the text, which can be printed and affixed separately later. Figure 96 illustrates a potential print setup for the PCB holder. Another option is to print with the surface shown in Figure 95a against the printer bed.



	Feature type	Time	Percentage	Used filament	
🟨	Perimeter	32m	44.3%	0.72 m	2.13 g
🟧	External perimeter	31m	44.1%	0.90 m	2.67 g
🟥	Internal infill	6m	8.8%	0.13 m	0.39 g
🟪	Solid infill	1m	1.7%	0.06 m	0.18 g
🟥	Top solid infill	15s	0.4%	0.01 m	0.04 g
🟩	Skirt/Brim	19s	0.5%	0.02 m	0.05 g
🟩	Custom	13s	0.3%	0.02 m	0.06 g

Estimated printing times [Normal mode]:

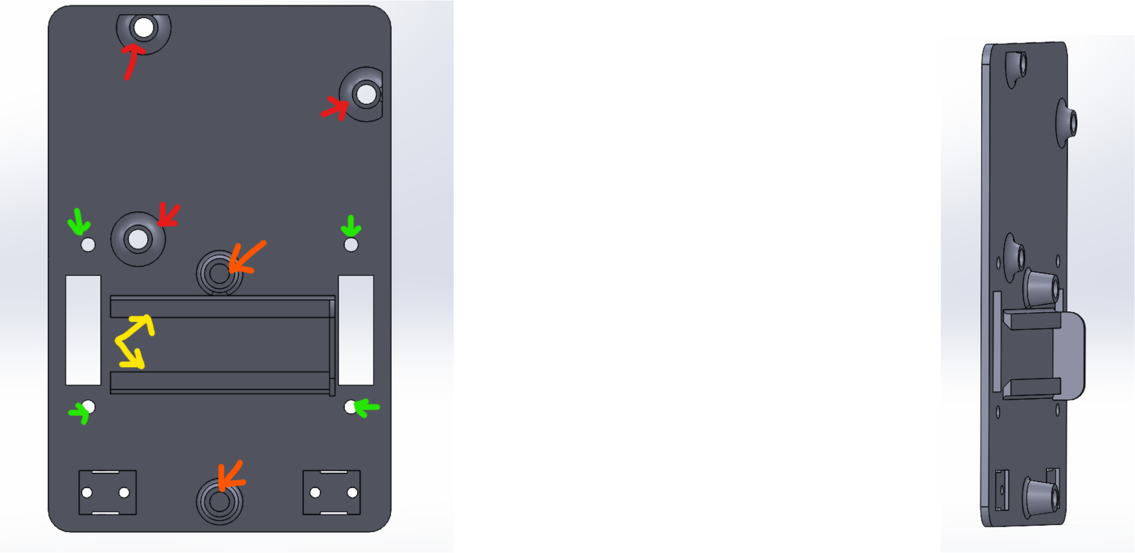
First layer: 42s

Total: 1h11m

Figure 96: Example of the PCB holder sliced in PrusaSlicer.

Arduino Mount Plate

The Arduino mount plate is fixed onto the bottom tube holder and placed on the rear side of the vertical aluminium profile of the SCL. Its dimensions are 120mm in height, 78.2mm in width, and 10.1mm in thickness at the highest point. Figure 97a shows the front view of the mounting plate. The green arrows point to the holes for the M3 screws to secure the plate to the bottom tube holder. The red arrows indicate the locations for the M3 threaded inserts to fasten the Arduino UNO R3, whereas the orange arrows point to the spots for the M3 threaded inserts for the cover. The yellow arrows point where the PCB holder should be inserted. Beneath the green arrows at the bottom and adjacent to the orange arrow on either side, a fitting for the potentiometer used in this thesis is placed. Figure 97b provides a three-dimensional view of the Arduino mount plate.

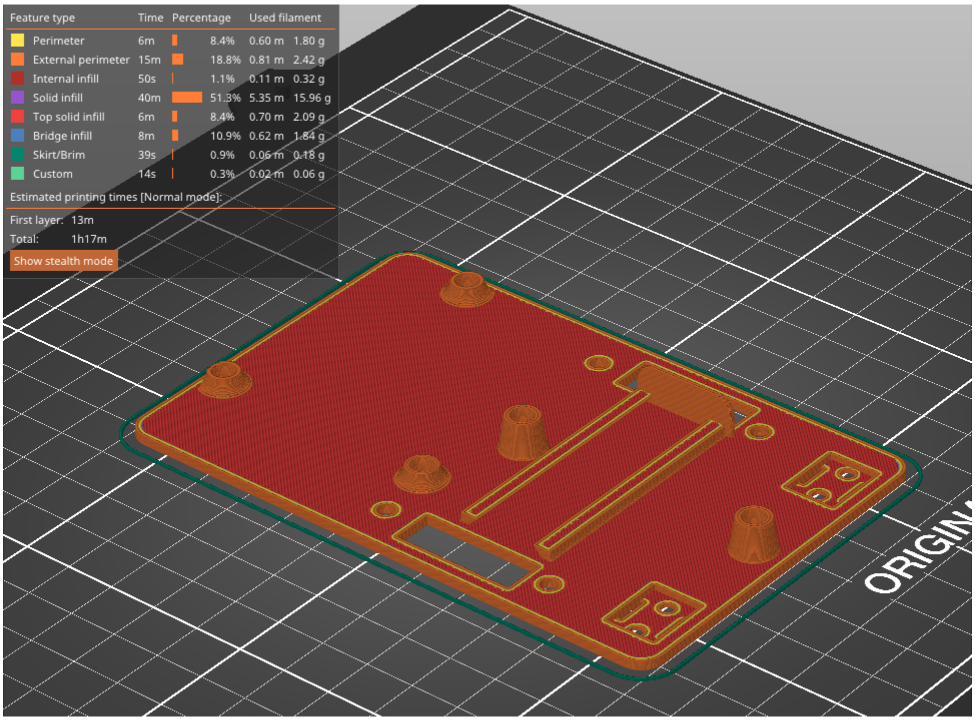


(a) Front view.

(b) Side view.

Figure 97: The Arduino mount plate.

The recommended 3D printing setup for the Arduino mount plate involves placing it flat with the rear side on the print bed, as illustrated in Figure 98.



Feature type	Time	Percentage	Used filament	
🟨 Perimeter	6m	8.4%	0.60 m	1.80 g
🟧 External perimeter	15m	18.8%	0.81 m	2.42 g
🟥 Internal infill	50s	1.1%	0.11 m	0.32 g
🟪 Solid infill	40m	51.3%	5.35 m	15.96 g
🟥 Top solid infill	6m	8.4%	0.70 m	2.09 g
🟦 Bridge infill	8m	10.9%	0.62 m	1.84 g
🟩 Skirt/Brim	39s	0.9%	0.06 m	0.18 g
🟩 Custom	14s	0.3%	0.02 m	0.06 g

Estimated printing times [Normal mode]:

First layer: 13m

Total: 1h17m

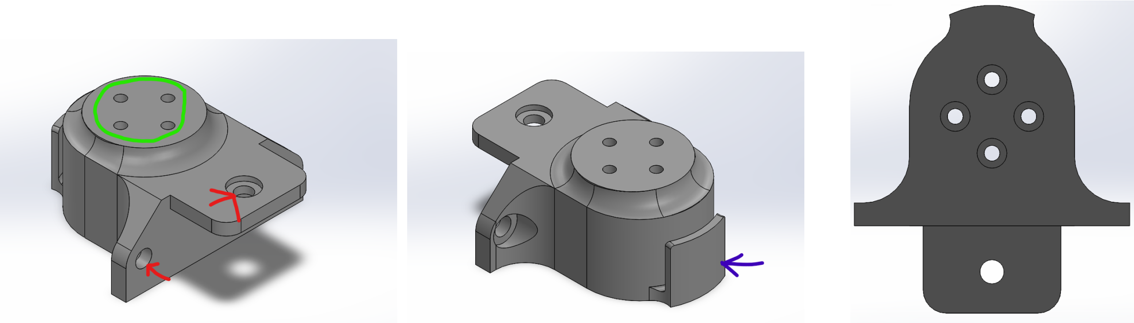
Figure 98: Example of the Arduino mount plate sliced in PrusaSlicer.

10.4 BLDC Mount

The BLDC mount is the component that secures the BLDC that drives the wheels. It is attached to a vertical aluminium profile using screws. Geometrically, it measures 67mm in height, and 60mm in width, and has a total thickness of 29mm.

Figure 99a presents a top front left view of the mount, where red arrows indicate the locations for the M5 screws to be affixed to the aluminium profile. The green circle shows where the BLDC flange is placed with its hole circle. The centre of the flange is 43.8mm from the mid-point of the aluminium profile, as illustrated in Figure 74. The mount features a 19mm protrusion, designed to overhang the aluminium profile, to prevent any interference during installation.

Moving on to Figure 99b, a top rear right view, a cable channel is visible. This has been designed to guide the three cables running from the BLDC to the ESC. Lastly, Figure 99c, a bottom view, highlights the intended placement of the bolts for the BLDC.



(a) Top front left view.

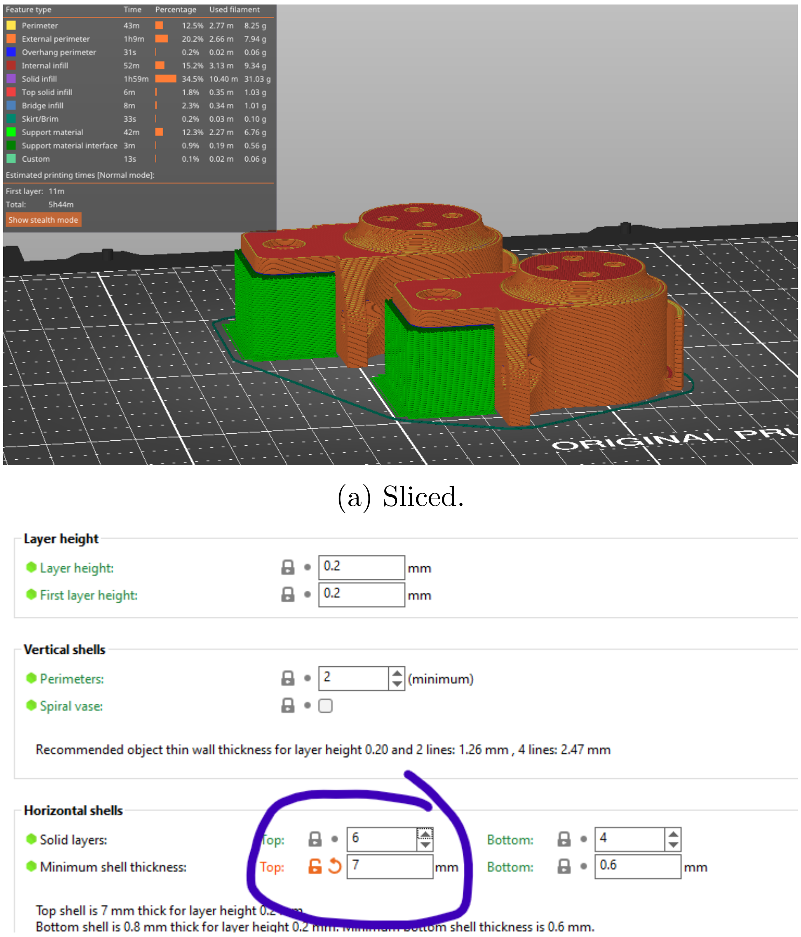
(b) Top rear right view.

(c) Bottom view.

Figure 99: The BLDC mount.

No simulations of the BLDC mount have been performed; however, careful considerations have been made regarding the thickness of the part where the BLDC is secured. Figure 100a illustrates how the BLDC Mount is positioned on the printer bed, with support required on the build surface.

Moreover, Figure 100b shows the modification of a setting to ensure a minimum top shell thickness of 7mm. This alteration results in a solid top layer. This measure has been introduced to prevent any potential vibration-induced dislodging of the infill patterns, thereby mitigating the risk of structural weakening and enhancing overall safety.



Feature type		Time	Percentage	Used filament	
🟨	Perimeter	43m	12.5%	2.77 m	8.25 g
🟧	External perimeter	1h9m	20.2%	2.66 m	7.94 g
🟦	Overhang perimeter	31s	0.2%	0.02 m	0.06 g
🟥	Internal infill	52m	15.2%	3.13 m	9.34 g
🟪	Solid infill	1h59m	34.5%	10.40 m	31.03 g
🟥	Top solid infill	6m	1.8%	0.35 m	1.03 g
🟦	Bridge infill	8m	2.3%	0.34 m	1.01 g
🟩	Skirt/Brim	33s	0.2%	0.03 m	0.10 g
🟩	Support material	42m	12.3%	2.27 m	6.76 g
🟩	Support material interface	3m	0.9%	0.19 m	0.56 g
🟩	Custom	13s	0.1%	0.02 m	0.06 g

Estimated printing times [Normal mode]:

First layer: 11m

Total: 5h44m

(a) Sliced.

Layer height

- Layer height: 0.2 mm
- First layer height: 0.2 mm

Vertical shells

- Perimeters: 2 (minimum)
- Spiral vase: □

Recommended object thin wall thickness for layer height 0.20 and 2 lines: 1.26 mm , 4 lines: 2.47 mm

Horizontal shells

- Solid layers: Top: 6 Bottom: 4
- Minimum shell thickness: Top: 7 mm Bottom: 0.6 mm

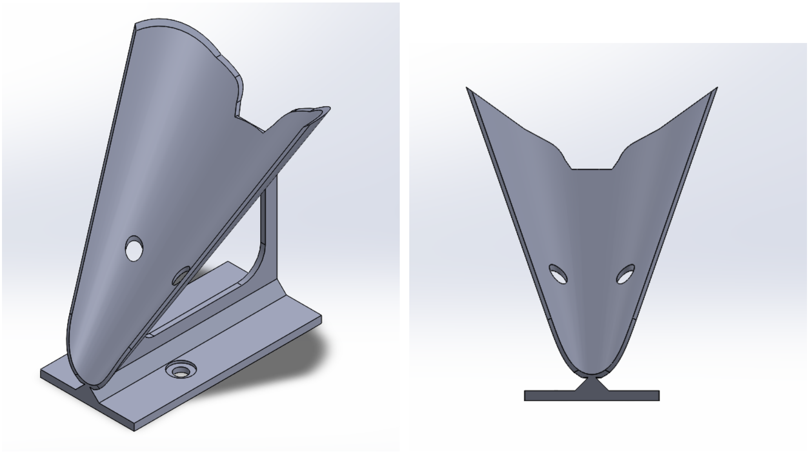
Top shell is 7 mm thick for layer height 0.2 mm.
Bottom shell is 0.8 mm thick for layer height 0.2 mm. Minimum bottom shell thickness is 0.6 mm.

(b) Change of setting.

Figure 100: Example of the BLDC mount sliced in PrusaSlicer.

Guider

The guider is placed between the BLDC mount and the fixed attachment. Geometrically, it has a length of 80mm, height of a total of 93.4mm and width of a total of 74.7mm. The guider's task is to receive the SCs from the tube and the feed mechanism and correctly guide them into the wheels. This is done by making a half-shaped spherical slightly larger dimension like the SC. The guide is attached with two M5 bolts and has a cut-out at the top so that the arm with the shuttle retainer can pass through so that the drop for the SC is as short as possible, as shown in Figure 101b. Furthermore, as shown in 101a, it stands on a hollow structure made to save weight and materials.

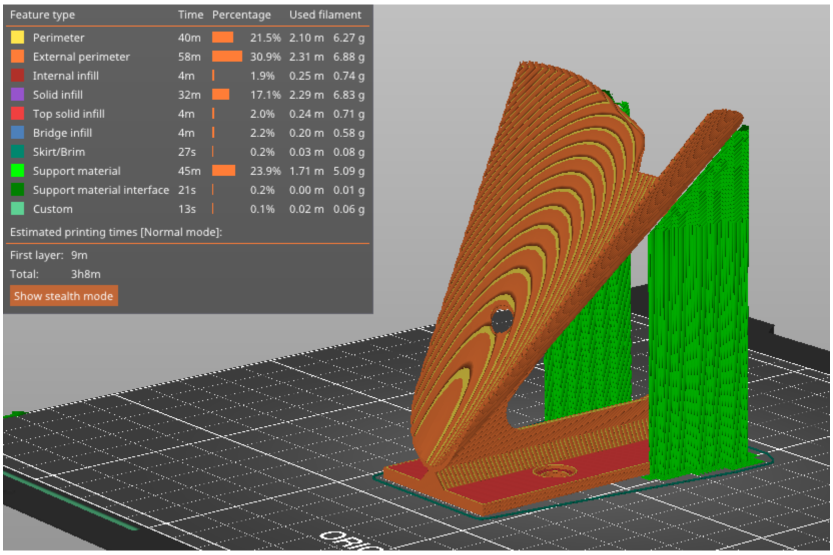


(a) Top front right view.

(b) Front view.

Figure 101: The guider.

A topology study has been carried out of the guider with a load case with the weight of 15 SC distributed over the top surface. The result gave an example of the cutout done on the rear wall. Figure 102 shows how it can be 3d printed and requires support from the build surface.



	Feature type	Time	Percentage	Used filament	
🟨	Perimeter	40m	21.5%	2.10 m	6.27 g
🟧	External perimeter	58m	30.9%	2.31 m	6.88 g
🟥	Internal infill	4m	1.9%	0.25 m	0.74 g
🟪	Solid infill	32m	17.1%	2.29 m	6.83 g
🟥	Top solid infill	4m	2.0%	0.24 m	0.71 g
🟦	Bridge infill	4m	2.2%	0.20 m	0.58 g
🟩	Skirt/Brim	27s	0.2%	0.03 m	0.08 g
🟩	Support material	45m	23.9%	1.71 m	5.09 g
🟩	Support material interface	21s	0.2%	0.00 m	0.01 g
🟩	Custom	13s	0.1%	0.02 m	0.06 g

Estimated printing times [Normal mode]:

First layer: 9m

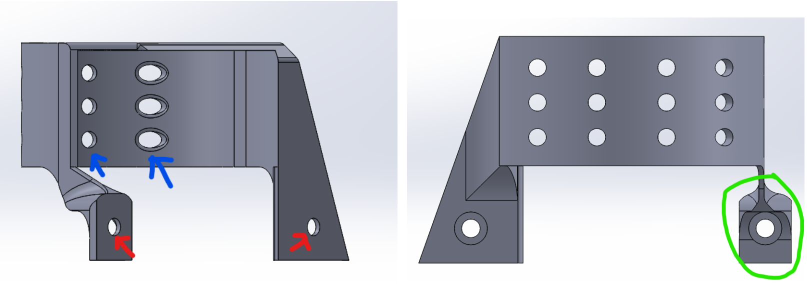
Total: 3h8m

Figure 102: Example of the Guider sliced in PrusaSlicer.

HSE Wheel Cover

The HSE wheel cover is placed on the BLDC mount and is screwed onto the vertical aluminium profile. Geometrically, it is 107mm long, 65.2mm high and 62mm wide, with a uniform thickness of approximately 2 mm.

The HSE wheel cover can prevent injuries that can occur by touching the wheels when they are in operation. Furthermore, they also have the function of reducing turbulence created by the wheels with the help of the holes shown in Figure 103a, and it is cut off at the rear edge to let air out behind the guider shown between the left blue and arrow in Figure 103a.

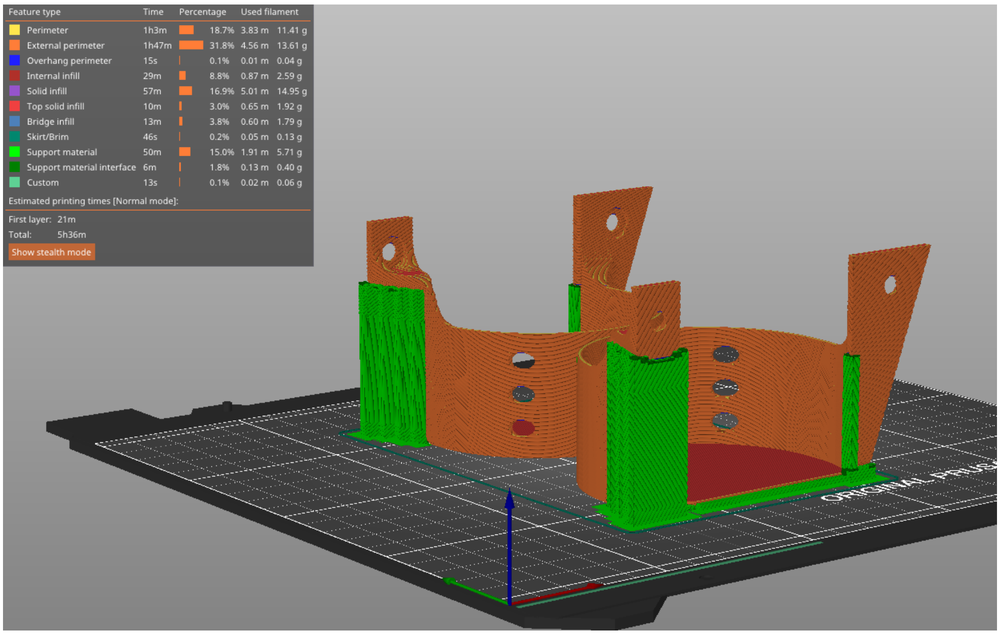


(a) Top front right view.

(b) Front view.

Figure 103: The HSE wheel cover.

No simulations were performed on the HSE wheel cover as the physical tests were considered suitable. Figure 104 shows the direction of 3D printing where two are placed and are mirrored. Support from the build surface is required, and a mirror function in PrusaSlicer can be used to mirror the HSE wheel cover.



	Feature type	Time	Percentage	Used filament	
🟨	Perimeter	1h3m	18.7%	3.83 m	11.41 g
🟧	External perimeter	1h47m	31.8%	4.56 m	13.61 g
🟦	Overhang perimeter	15s	0.1%	0.01 m	0.04 g
🟥	Internal infill	29m	8.8%	0.87 m	2.59 g
🟪	Solid infill	57m	16.9%	5.01 m	14.95 g
🟥	Top solid infill	10m	3.0%	0.65 m	1.92 g
🟦	Bridge infill	13m	3.8%	0.60 m	1.79 g
🟩	Skirt/Brim	46s	0.2%	0.05 m	0.13 g
🟩	Support material	50m	15.0%	1.91 m	5.71 g
🟩	Support material interface	6m	1.8%	0.13 m	0.40 g
🟩	Custom	13s	0.1%	0.02 m	0.06 g

Estimated printing times [Normal mode]:

First layer: 21m

Total: 5h36m

Figure 104: Example of the HSE wheel cover sliced in PrusaSlicer.

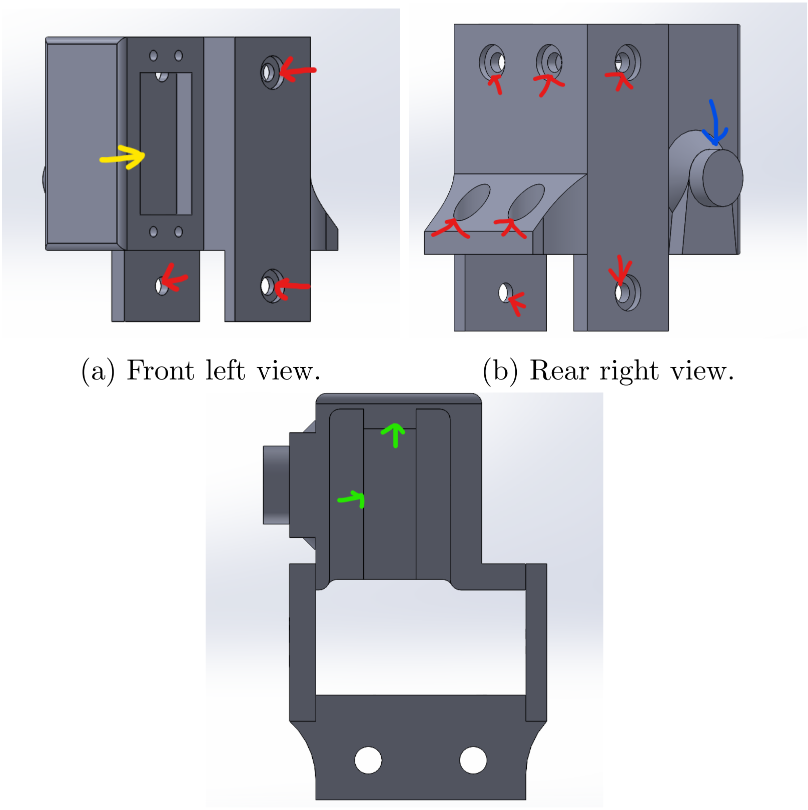
10.5 Pan tilt

Fixed Attachment

The fixed attachment is a central part of the SCL as it accommodates the aluminium profile and the 20kg digital servo motor for tilting. Its dimensions are 80mm high, 49mm wide, and 77.7mm thick.

The fixed attachment performs three essential functions:

1. It holds the vertical aluminium profile in place. This is done using four M5 screws, two marked with red arrows at the bottom of Figure 105a and two located centrally to the middle left in Figure 105b.
2. It supports the vertical profile, attached through a hole viewed from below in Figure 105c and secured using the holes marked with red arrows at the top of Figure 105b.
3. It houses the 20kg servo motor, which tilts the SCL. The motor is placed where the yellow arrow points in Figure 105a. The motor rests on the fixed attachment against the internal surfaces indicated by the green arrow in Figure 105c. On the opposite side, the fixed attachment rests on the tilt attachment using the protrusion marked by the blue arrow in Figure 105b. This arrangement ensures that the fixed attachment is centred against the axis of the servo motor.



(a) Front left view.

(b) Rear right view.

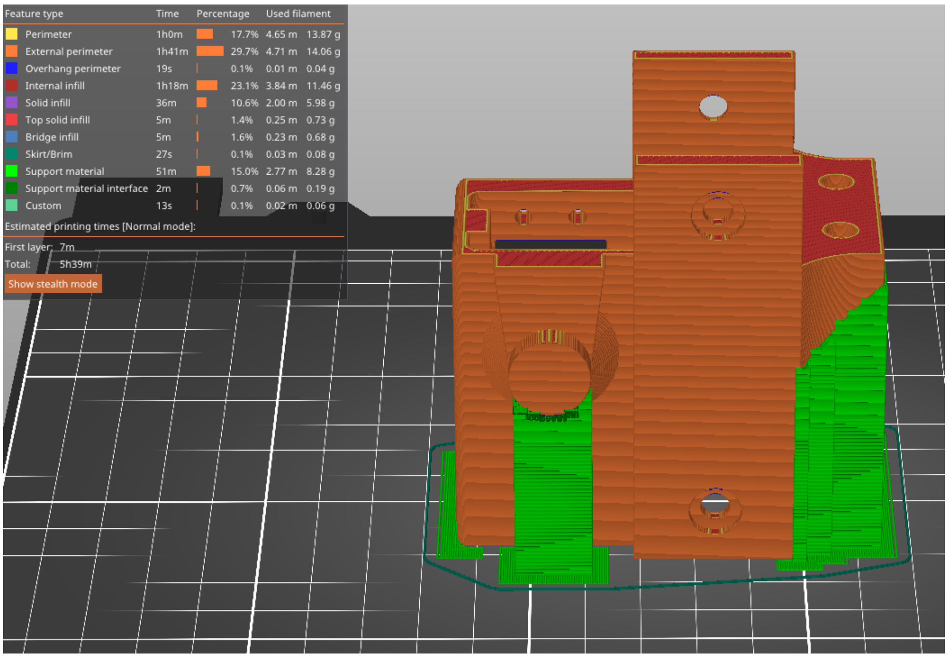
(c) View from below.

Figure 105: The fixed attachment.

Simulations have been conducted to evaluate the structural integrity of the fixed attachment. A series of static simulations were carried out and are presented in Chapter 6. Based on these simulations, no areas of concentrated force had a significant impact, indicating that the part was sufficiently robust to support the SCL. The maximum stress detected was around 6 MPa, given a load factor of 5 kg, while the machine weighs approximately 2 kg.

Considering PLA and PetG materials, there is a broad variety of types and suppliers, each with different material specifications. A safety factor of approximately five is determined without selecting a specific material, using a rough tensile strength reference of 30 MPa. This number is based on experience and the material's tensile strength in SW. However, PLA and PetG can possess higher or lower tensile strengths. When purchasing material, the suppliers provide the specifications and must be evaluated.

The print directions are also set as shown in Figure 106.



Feature type	Time	Percentage		Used filament	
🟨 Perimeter	1h0m	▬	17.7%	4.65 m	13.87 g
🟧 External perimeter	1h41m	▬	29.7%	4.71 m	14.06 g
🟦 Overhang perimeter	19s	|	0.1%	0.01 m	0.04 g
🟥 Internal infill	1h18m	▬	23.1%	3.84 m	11.46 g
🟪 Solid infill	36m	▬	10.6%	2.00 m	5.98 g
🟥 Top solid infill	5m	|	1.4%	0.25 m	0.73 g
🟦 Bridge infill	5m	|	1.6%	0.23 m	0.68 g
🟩 Skirt/Brim	27s	|	0.1%	0.03 m	0.08 g
🟩 Support material	51m	▬	15.0%	2.77 m	8.28 g
🟩 Support material interface	2m	|	0.7%	0.06 m	0.19 g
🟩 Custom	13s	|	0.1%	0.02 m	0.06 g

Estimated printing times [Normal mode]:

First layer: 7m

Total: 5h39m

Figure 106: Example of the Fixed Attachment sliced in PrusaSlicer.

This direction is not optimal, as the laminate direction will be perpendicular to assumed stresses, creating stretch stress in the lamina. However, screws have been added to distribute the stresses, minimising the impact of this sub-optimal orientation.

Tilt Attachment

The tilt attachment is an essential component of the SCL, assigned with carrying the entire SCL. It performs this role by holding the fixed attachment at the top, as illustrated in Figure 107a, where the blue arrows indicate the connection points.

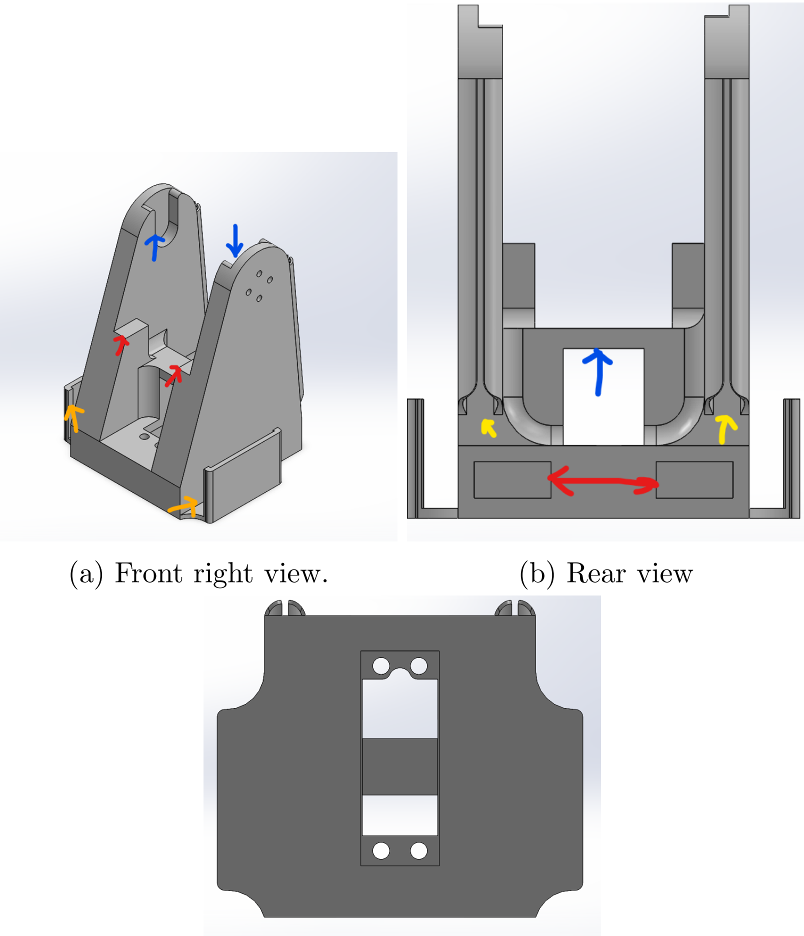
Geometrically, the tilt attachment measures 127mm in height, 97mm in width, and 80mm in thickness. This considerable size makes it one of the largest parts within the SCL. The design of the tilt attachment encompasses four functional surfaces:

1. The main job for the tilt attachment is to bear the weight of the fixed attachment and the entire SCL, at the position marked by the blue arrows in Figure 107a.
2. The cavity shown in Figure 107c is designed to house the 20kg servo motor responsible for rotation. This motor also sustains weight at this point. A broader resting beam has been designed for the servo motor to ensure efficient weight distribution, as highlighted by the blue arrow in Figure 107b. This

beam serves a dual purpose: firstly, to stop the servo motor and, secondly, to reinforce the structure.

3. The tilt attachment includes two resting surfaces for the vertical aluminium profile, which prevent the aluminium profile from falling too far and work as a physical stopper. These surfaces are pointed out by the red arrow in Figure 107a.
4. The tilt attachment has two custom-made slots to fit each ESC as illustrated by the orange arrow in Figure 107a. The WAGO connectors for power and grounding can be neatly tucked away within the holes portrayed in Figure 107b. Moreover, channels have been constructed to route power, ground and signal cables to the PCB, as demonstrated by the yellow arrows in Figure 107b.

Additionally, the flange shown in Figure 107a is designed so that its centre aligns with the axis of the servo motor beneath it, which is responsible for rotation. This alignment has been intentionally done to ensure a direct stress flow. As a result, bending stresses are avoided in one plane in theory. Furthermore, shown in Figure 107c, the bottom side of the attachment has been made flat to create a surface for a thrust bearing. This bearing is designed to absorb all the pressure, thus relieving any stress on the servo motor.



(a) Front right view.

(b) Rear view

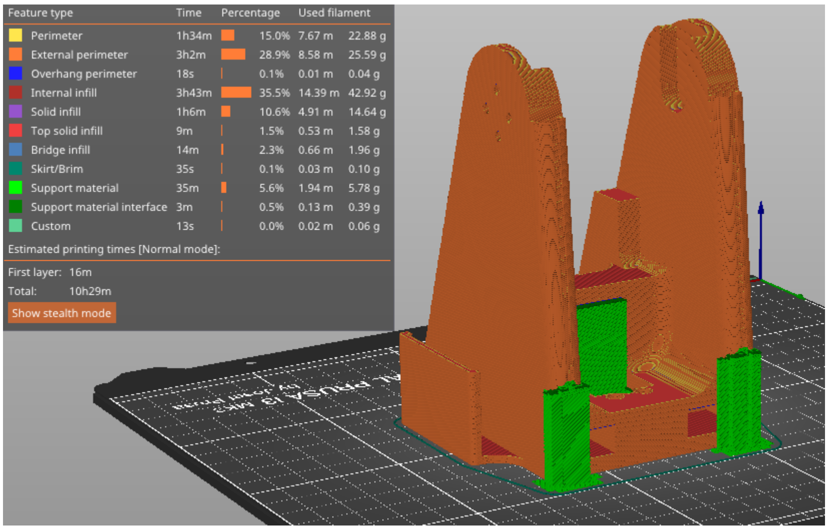
(c) View from below.

Figure 107: The tilt attachment.

Simulations have been carried out for tilt attachment, and the results are shown in chapter 6. Given the physical demands of the tilt attachment, careful considerations of material and print direction were made. However, considering the minimum tension results, a topology study was performed, which involved removing materials to optimise the part's robustness and durability. Minimal material was removed, as Figure 29 showed, and there is potential to make the construction smaller and lighter for further development.

However, material selection is vital for the tilt attachment, like the fixed attachment. Material properties must be carefully evaluated to ensure suitability for the task.

For 3D printing, the tilt attachment, a specific orientation is required. Figure 108 shows how it should be positioned on the print surface. In this configuration, the laminae align perpendicularly to the applied weight, reinforcing the bond and strength between the layers.



Feature type	Time	Percentage	Used filament	
🟨 Perimeter	1h34m	15.0%	7.67 m	22.88 g
🟧 External perimeter	3h2m	28.9%	8.58 m	25.59 g
🟦 Overhang perimeter	18s	0.1%	0.01 m	0.04 g
🟥 Internal infill	3h43m	35.5%	14.39 m	42.92 g
🟪 Solid infill	1h6m	10.6%	4.91 m	14.64 g
🟥 Top solid infill	9m	1.5%	0.53 m	1.58 g
🟦 Bridge infill	14m	2.3%	0.66 m	1.96 g
🟩 Skirt/Brim	35s	0.1%	0.03 m	0.10 g
🟩 Support material	35m	5.6%	1.94 m	5.78 g
🟩 Support material interface	3m	0.5%	0.13 m	0.39 g
🟩 Custom	13s	0.0%	0.02 m	0.06 g

Estimated printing times [Normal mode]:

First layer: 16m

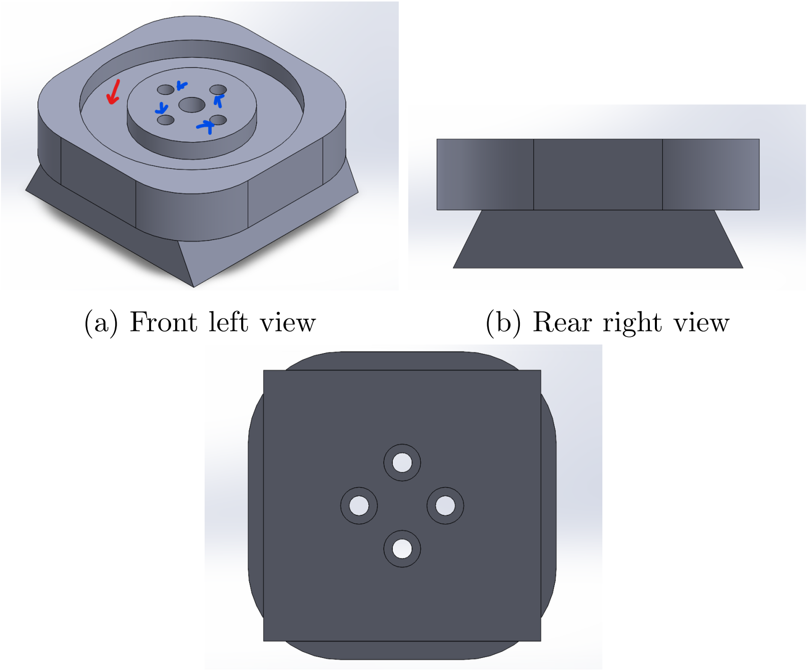
Total: 10h29m

Figure 108: Example of the Tilt Attachment sliced in PrusaSlicer.

Base Attachment

The base attachment comes in two versions, a 5-spoke which is screwed into onto a table and was introduced in chapter 6 where Figure 31 illustrates the attachment, or tripod as shown in Figure 109. Both have the task of standing stationary with a flange to connect the rotor servo. Figure 109a shows tripod base attachment, where the blue arrows point to the bolt circle for the flange of the 20kg servo, while the red arrow points to the surface and slot where the trust bearing is to be placed.

The geometry of the triangle at the bottom, as illustrated in Figure 109b, measures 45mm along the base and 36mm at the top. Its length is designed to accommodate a camera stand. Although this is not a standard, many models use dimensions of this size. The slot for the trust bearing is from a standard 51105 [58].



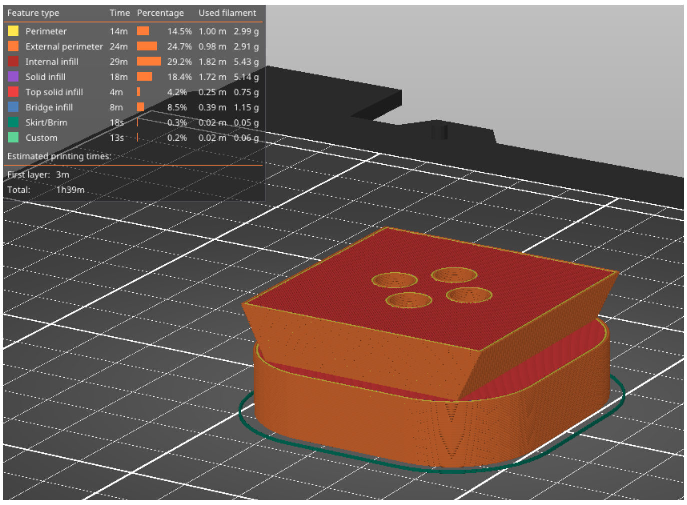
(a) Front left view

(b) Rear right view

(c) View from below

Figure 109: The Tripod Base Attachment.

Simulations performed on the tripod version of the base attachment revealed minimal stresses. For 3D printing is recommended to orient the part as shown in Figure 110, eliminating the need for support, and will align the laminae perpendicularly to the pressure stresses, reinforcing the bond and strength between the layers.



Feature type		Time	Percentage		Used filament	
🟨	Perimeter	14m	▬	14.5%	1.00 m	2.99 g
🟧	External perimeter	24m	▬	24.7%	0.98 m	2.91 g
🟥	Internal infill	29m	▬	29.2%	1.82 m	5.43 g
🟪	Solid infill	18m	▬	18.4%	1.72 m	5.14 g
🟥	Top solid infill	4m	▬	4.2%	0.25 m	0.75 g
🟦	Bridge infill	8m	▬	8.5%	0.39 m	1.15 g
🟩	Skirt/Brim	18s	▬	0.3%	0.02 m	0.05 g
🟩	Custom	13s	▬	0.2%	0.02 m	0.06 g

Estimated printing times:

First layer: 3m

Total: 1h39m

Figure 110: Example of the Tripod Base Attachment sliced in PrusaSlicer.

10.6 Additional Parts

Two additional components, the joystick and the extension tube, have yet to be presented in detail. While these parts are printable, no specific calculations, 3D print times, or directions are associated.

The joystick, as seen in Figure 58, was obtained from Thingiverse. It is designed to house a joystick which can be fastened inside. This assembly allows wires to be led outwards, or a small Bluetooth transmitter and receiver can be used.

The extension tube, seen at the top of Figure 68, is an extension of the magazine, allowing it to accommodate a larger quantity of SCs.

10.7 Choice of Material

When selecting a material for a product that most 3D printers can use, PetG plastic is the optimal choice due to its strength and material properties. It's essential to examine the specific PetG intends to use closely; always refer to the supplier's product data-sheet to ensure the material's properties meet the requirements. Verify that the tensile strength is at least 30MPa, the reference used in this thesis's simulations, and check that the selected PetG is compatible with the 3D printer. This is done by checking the recommended printed temperature settings on the material sheet.

While PLA has proven effective in general testing, it is biobased and therefore degrades over time [68], making PetG a more durable alternative. ABS could be another option, but it requires the 3D printer to be enclosed in a consistent temperature environment for successful printing. This requirement deviates from our objective of ensuring simplicity and ease in manufacturing.

PLA is most suitable for development and prototyping purposes, and will thus be employed for cost estimation [69].

10.8 Cost Estimation

An estimation has been performed to determine the cost of the SCL. As demonstrated in Table 11 and Table 12, material usage and approximate printing time for parts have been detailed. Worst-case scenarios involving time with 0.4 nozzles and material from 0.6 nozzles have been considered in estimating power consumption and material usage.

Table 13 presents a rough cost calculation of the parts. Not all parts are the same since the mechatronics lab has free access to various resources. However, the estimate indicated in the table represents the cost of production in Norway, considering Norwegian prices under circumstances where immediate procurement of parts is required.

As mentioned in 8.2, the battery used in this thesis was unbranded. However, the machine pulls roughly 0.2A, which means a 200mAh battery should last 1 hour with the 12.7volt power supply.

Generally, these batteries range from 600-3000 mAh. If an 18-volt battery is chosen and the ESC can operate on this voltage, then the values for the potentiometers would need to be reduced to prevent the BLDC from spinning too quickly. If an 11-volt battery is chosen, the potentiometer values must be increased to ensure the BLDC can achieve sufficient speed. The maximum voltage an ESC can handle is typically listed on the product page or within its specifications.

Table 11: 3D Print Time and Material Usage for Each Part Using 0.4mm Nozzle and PLA.

Part No.	Part Name	Print Time [min.]	Material Usage [Grams / meters]		Support Type
1	Tube	302	48.4	16.22	Everywhere
2	Top Tube Holder	284	47.2	15.8	None
3	Bottom Tube Holder	203	28.9	9.7	None
4	Arm.V2	23	3.7	1.2	None
5	Pusher	13	1.12	0.37	None
6	Shuttle Retainer	43	3.66	1.23	None
7	Cover Arduino Holder	250	48.98	16.42	On build plate only
8	PCB Holder	71	5.53	1.85	None
9	Arduino Mount Plate	111	24.39	8.18	None
10	Guider	188	27.26	9.14	On build plate only
11	HSE Wheel Cover	336	52.61	17.64	On build plate only
12	BLDC Mount	344	66.16	22.18	On build plate only
13	Fixed Attachment	339	55.43	18.59	On build plate only
14	Tilt Attachment	629	115.94	38.87	On build plate only
15	Base Attachment	99	18.49	6.2	None
	SUM:	53h55m	547.77	183.59	

Table 12: 3D Print Time and Material Usage for Each Part Using 0.6mm Nozzle and PLA.

Part No.	Part Name	Print Time [min.]	Material Usage [Grams / meters]		Support Type
1	Tube	186	53.1	17.8	Everywhere
2	Top Tube Holder	204	54.9	18.4	None
3	Bottom Tube Holder	149	32.6	11	None
4	Arm.V2	16	4.1	1.4	None
5	Pusher	11	1.12	0.38	None
6	Shuttle Retainer	31	3.59	1.2	None
7	Cover Arduino Holder	178	51.99	17.43	On build plate only
8	PCB Holder	50	6	2.01	None
9	Arduino Mount Plate	77	34.67	8.27	None
10	Guider	147	30.64	10.27	On build plate only
11	HSE Wheel Cover	216	57.84	19.39	On build plate only
12	BLDC Mount	252	73.25	24.56	On build plate only
13	Fixed Attachment	256	70.72	23.72	On build plate only
14	Tilt Attachment	449	136.08	45.63	On build plate only
15	Base Attachment	74	21.01	7.04	None
	SUM:	38h16m	631.61	208.5	

Table 13: Estimate of equipment for the SCL.

Part No.	Part Name	Dim	pcs	Price	Total [NOK]
1	Aluminium Profile	20x20	2	39 [70]	78
2	Bolts	M5x8	24	39 [71]	39
3	T-Nuts	M5	24	89 [72]	89
4	Bolts	M4x10	4	39 [73]	39
5.1	Bolts	M3x6	9		
5.2	Bolts	M3x10	8		
5	M3 bolt kit		1	89 [74]	89
6.1	Thread Inserts	M4x4	8		
6.2	Thread Inserts	M3x4	9		
6	Thread Insert Kit		1	219 [75]	219
7	Trust Bearing	51105	1	279 [55]	279
8	Hook-up cable	22AWG	1	299 [76]	299
9	Soldering breadboard	-	1	65 [77]	65
10	Header	-	1	40 [78]	40
11	Female headers	-	2	30 [79]	60
	SUM:				1 296

Part No.	Part Name		pcs	price	Total [NOK]
11	Arduino UNO R3		1	349 [60]	349
12	Potentiometer		2	13.18 [80]	26.36
13	WAGO		1	59 [81]	59
14	Dualsky Lite 45A ESC 2-3S		2	289 [8]	578
15	iFlight XING 2 2604 1650KV Race Motor		2	329 [8]	658
16	Single 20kg digital servo		2	895 [51]	1790
17	SG90 servo		1	69 [82]	69
18	VOLANTE wheels		1	299 [83]	299
19	PLA Plastic by 3D net	750g	1	299 [84]	299
20	Power	2Kr/kWh [85]	53.9	108	108
21	Battery		1	175 [86]	175
	SUM:				4 411
	+				1 296
	+ Assumed shipping cost				1 000
	SUM:				6 707

Considering that there are servos, microcontrollers, BLDC and ESC, cheaper ones will bring the price down. Therefore, an alternative cost estimate of the mentioned parts has been compiled.

Table 14: Alternative equipment to lower the cost.

Part No.	Part Name	Dim	pcs	Price	Total [NOK]
7	Trust Bearing	51105	1	12 [57]	12
14 and 15	BLDC and ESC kit of 4		1	852 [87]	852
16	2x20kg digital servo		1	388 [54]	388
11	Microcontroller		1	60 [88]	60
	SUM:				1 312
	+				1 017
	+				736
	SUM:				3 065
	Assumed shipping and tax:				1 000
	SUM:				4 065

Assuming two SCL are to be made, the BLDC and ESC kit could save approximately 400 as a strategy. Further, in terms of the tools needed, it is generally short-listed as both the M5 screws and the M4 screws use a 3mm Allen key, while all the m3 screws use a 2.5mm Allen key and are all that is needed for the screws. Then there is soldering equipment which is relatively cheap depending on what is desirable, and multi-meter and 3D printers, which can cost from NOK 3,000 to 14,000.

Both given estimates deliver a thorough insight and will be discussed after completing the experimental testing.

11 Experimental Testing of the Shuttlecock Launcher

This chapter presents the experimental testing of the developed SCL. The objective is to investigate the SCLs performance attributes by examining the range, accuracy, and consistency under different tilt angles. An additional focus is placed on evaluating the efficiency and reliability of the feeding mechanism.

This experiment aims to confirm that the SCL meets the performance goals from chapter 1.4 and performs reliably under different operating conditions.

11.1 Equipment

The evaluation of the SCL required the following tools and equipment:

- The SCL developed for this thesis.
- Battery charger
- Tachometer
- Marker
- Multimeter
- A set of 11 used SCs, with additional ones for potential misfires or damages.
- A large, clear space marked with distance indicators to perform the tests.
- Measuring tape.

11.2 Preparation Procedure

The SCL is stationed at the service line with the tilting mechanism initially set to 0 degrees. It operates from a constant starting point and is approximately 74cm above the ground (see Figure 111). Throughout all tests, the launch rotation direction remains constant.

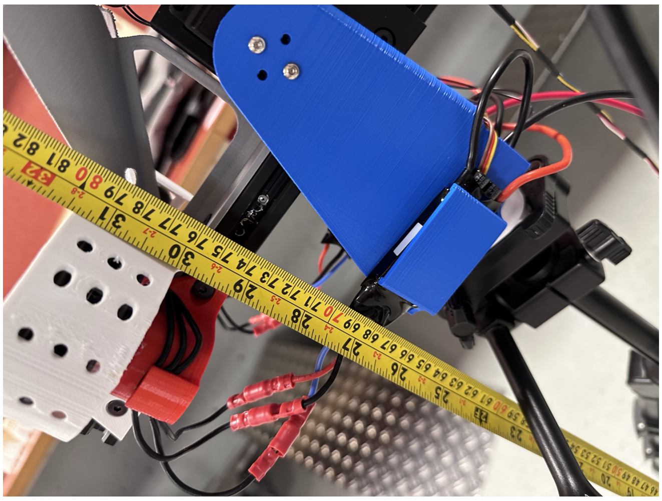


Figure 111: SCL positioned above ground, measured from the base to the vertical aluminium profile.

The test field is marked, where 0 meters is set at the SCL, and additional markers are placed up to 8 meters as shown along the table with note pads in Figure 112.

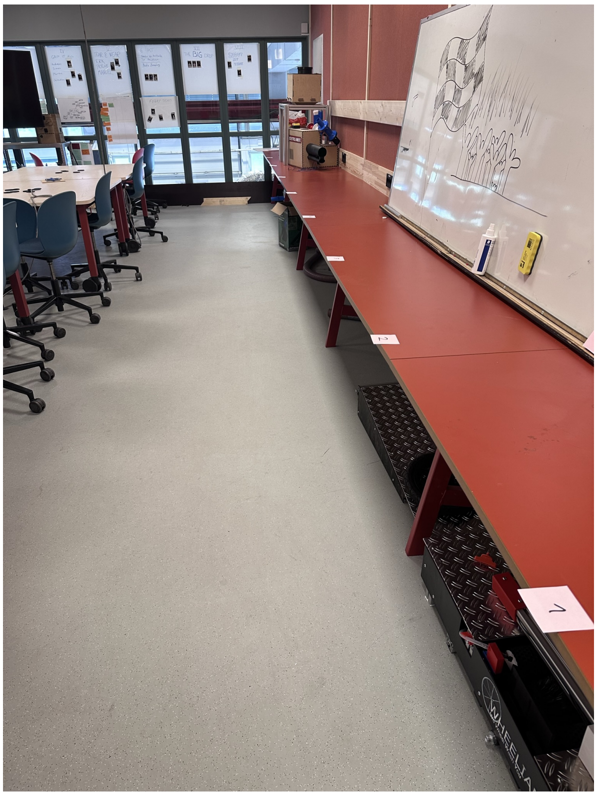


Figure 112: The test field.

11.3 Testing Procedure

1. Mark each of the 11 SCs to be used in each test round as shown in Figure 113. Document the sequence in which they are loaded into the extension tube of the SCL.

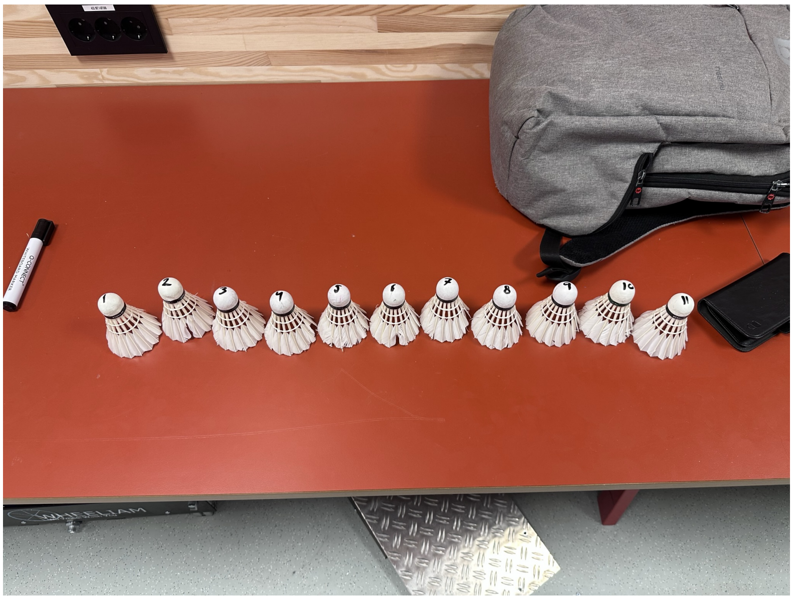


Figure 113: SC's marked before testing.

2. Start up MSA and MSB, measure the speeds and check the battery's voltage. (In this experiment, the speed was 9,100 RPM, and the voltage was 12.7v.)

3. For two rounds, load and note the SC in the sequence 1 to 11 to identify patterns of faults, if any potentially. Load and note the SC in a random sequence for the remaining eight rounds.

4. Launch the SC one by one and note any observed errors such as jams, double shots, or no-shot occurrences.

5. Set the RPM to 0, then record the distance where each SC lands by measuring the landing spot compared to the distance marks.

6. Repeat the steps above ten times before adjusting the launch angle.

7. After adjusting to the new angle, check the SC for damage and replace them if they cannot be used for further testing. Figure 114 shows the 11 SC used after 355 shots.

Figure 114: SC's marked after testing round.

Table 15 layout is specifically made to document the distance each SC travels, any encountered errors, and additional notes per shot.

Table 15: Example table layout for documenting observations.

Test No.	Tilt Angle	Shuttlecock No.	Distance [m]	Double shot (0/1)	No shot (0/1)	Jamming (0/1)	Notes
1	0	1	5.2		1		
1	0	2		1			
2	0	9	5.1				
2	0	5	5.5			1	

(This link shows a test round.)

Note: Blank on distance means 0m

11.4 Result

Table 16 show the summarised testing result. The test sheets result for each angle can be found in Appendix C.

Table 16: Result summarised.

Tilt Angle	Total Shots	Total Errors	Double shot	No shot	Jams	Avg. Distance [m]	Max Distance [m]	Min Distance [m]
0	111	14	8	5	1	4.77	6	3.1
10	111	15	10	5	0	5.14	6.2	2.1
20	111	17	4	11	2	5.25	6.1	3
30	22	10	0	10	0			
SUM	355	56	22	31	3			

Figure 115 presents the summarised results in a graphical format. Figure 116 shows the distance each SC was shot, categorised by SC number and SCL angle and Figure 117 shows the error type sorted by SC No.

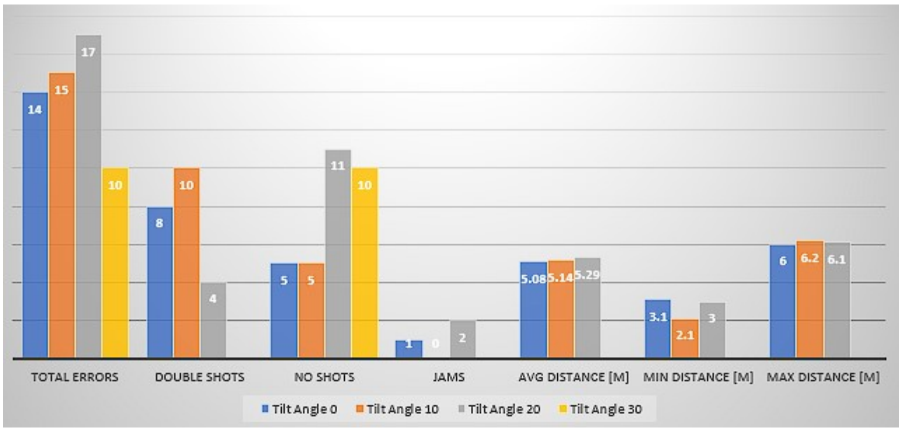


Figure 115: Graphical representation of results.

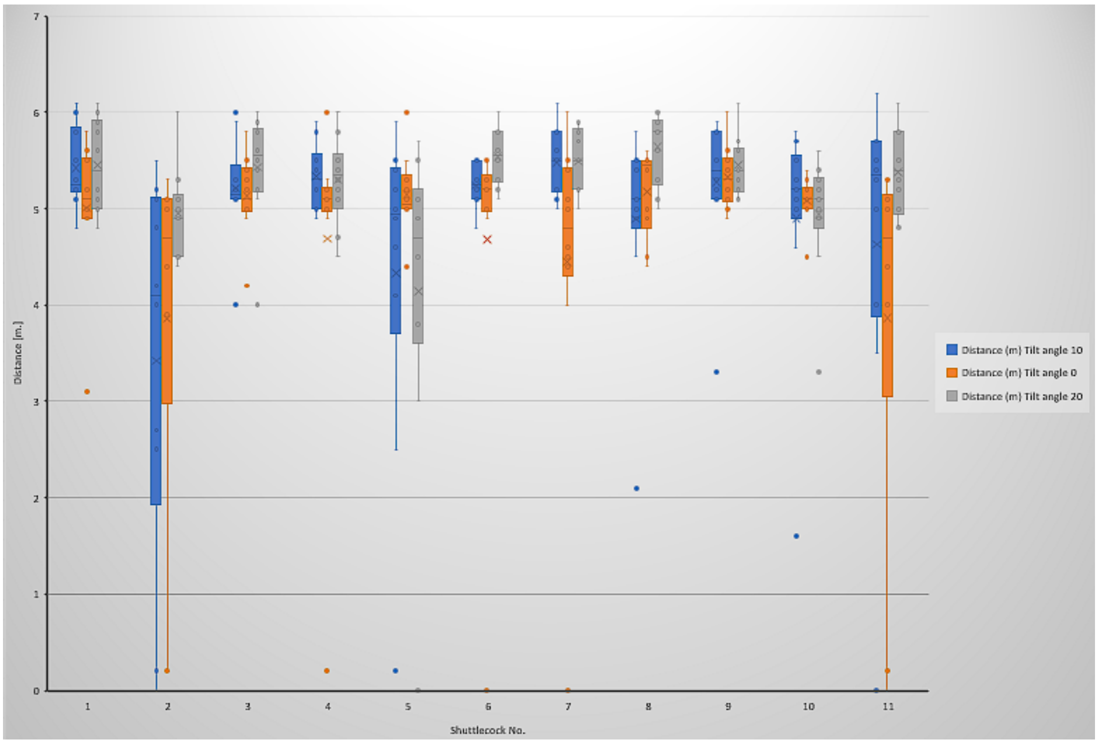


Figure 116: Shot length of each SC.

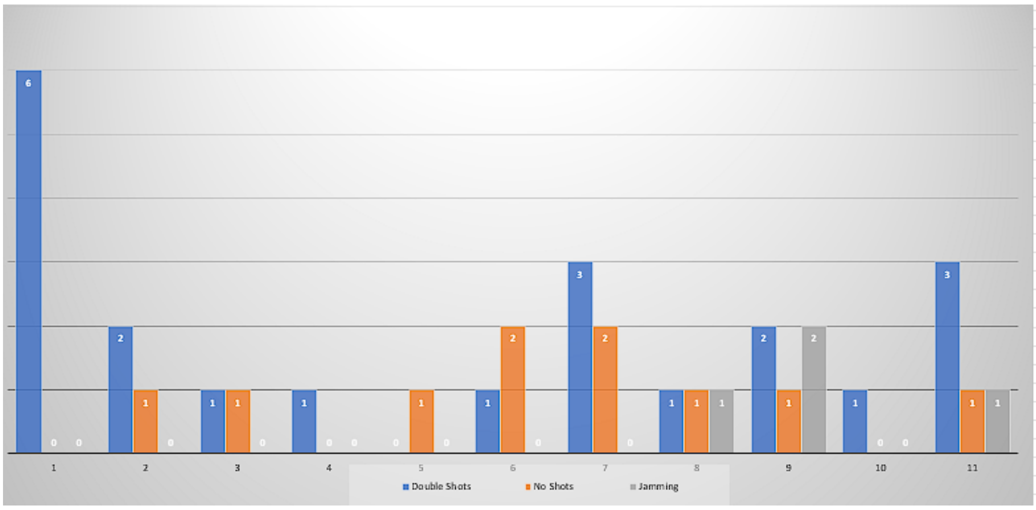


Figure 117: Errors sorted by SC No. Excluded test with a 30-degree angle.

12 Discussion

The results, as summarised in Table 16 show that the prototype SCL can replace hand feeding, which was the primary objective. However, at 30 degrees tilt, the design was not declining enough for a SC to efficiently drop from the tube to the wheels, frequently resulting in SCs nearly falling out of the tube and getting stuck onto the guider. While no SCs remained fully stuck, this problem often led to many failed attempts at shooting. In such instances, SCs would typically dislodge on the second or third attempt or gradually slide down the guider. In some cases, the slow movement from the tube to the guider caused the shuttle retainer to push on the SC feathers, leading to an incorrect trajectory, often resulting in the SC landing on the floor front of the SCL.

Given these issues, testing was halted after two rounds due to excessive errors. Consequently, it was concluded that the current SCL configuration is not suitable for launching at a 30-degree tilt. These specific errors will thus not be included in discussing the experimental results but addressed later in discussion and reflection in the context of product development.

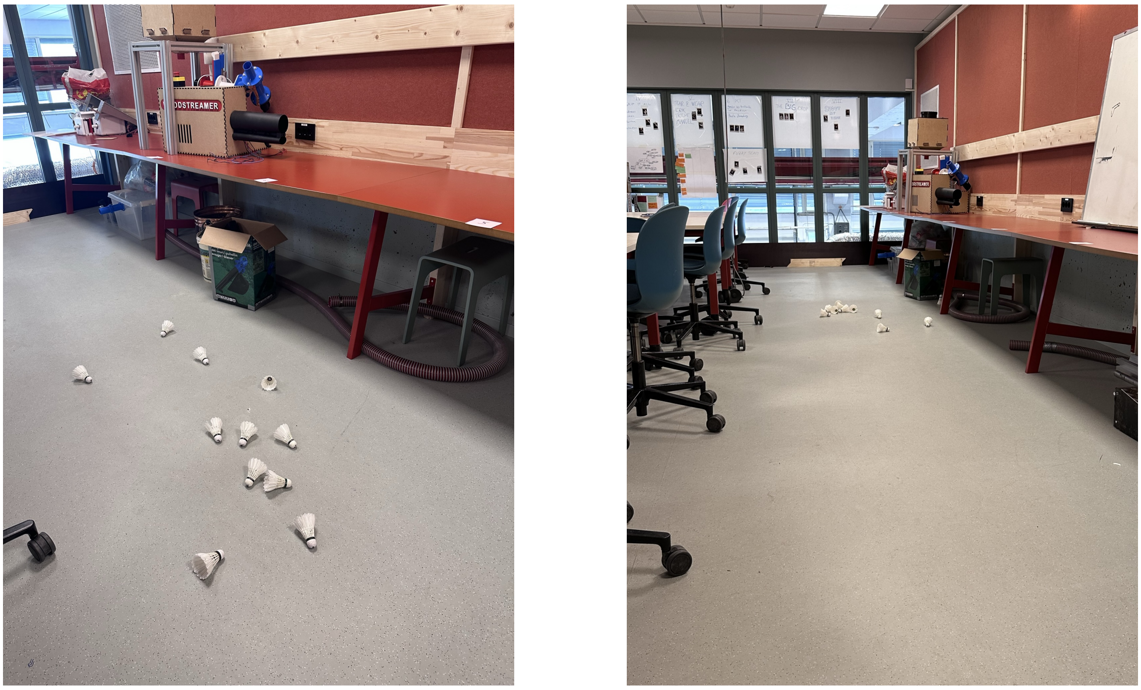
12.1 Discussion of the Experimental Results

After completing the experimental testing, a total of 46 errors were reported within the 0 to 20 degrees tilt range; 22 were double shots. As suggested by Figure 117, the weight of a fully loaded extension tube might cause the first two SCs to drop rapidly and then trigger a double shot. Figure 117 also implies a potential connection either with the weight of the SCs or the wear of the SC marked as 2. Double shots occurred in at least six rounds involving the SC marked as 1 and 2. The connection here is that for two tests, the SCs were arranged sequentially from 1 to 11 at each tilt angle, and on average, this occurrence was repeated in every round where the SCs were arranged in this sequence. However, despite these complications, the operation of the SCL was largely unhindered, and these errors were considered non-critical for the functionality of the SCL prototype.

A total of 21 No shots were recorded. However, as shown in Figure 117, these instances don't have a pattern. This issue typically occurred when the SC became stuck as it made contact with the needle. However, it was usually self-resolved by the second or third attempt, and therefore, it was considered an insignificant concern. No-shots and double-shots were considered acceptable deviations, primarily because they were self-corrected without manual intervention. On the other hand, jam errors were considered undesirable.

Out of 333 shots, three jams were observed, correlating to a jam error rate of a mere 1%. However, even this 1% could be seen as exaggerated. This is because a jam, by definition, must require a repair that involves stopping the machine. Notably, one of these jams and two no-shots directly resulted from the extension tube coming loose during the final test round at a 20-degree tilt. This malfunction disrupted the testing process and contributed significantly to the total error count. Therefore, it is essential to note that the jam and failed shots were not mechanical design errors but consequences of design issues. The reflection section 12.2 will discuss possible solutions to prevent jam failures.

As for accuracy, scattering was minimal, with SCs typically landing in roughly the same location, as depicted in Figures 118a and 118b. Such minimal scattering is beneficial when training beginners who prefer to concentrate on hitting the SC rather than on footwork. Moreover, it could ease future development in calculating trajectories and predicting landing spots.



(a) Distribution of SCs on the test field.

(b) Landing of SCs on the test field.

Figure 118: Test field distributions and landing locations.

The shoot distances did not reach the (performance goals 1.4) of 6-8 metres, but the average of 0, 10 and 20 degrees of tilt was about 4.8, 5.1 and 5.3m, respectively, which are decent lengths. The machine can be easily moved due to its lightweight construction (which was one of the objectives 1.3), and since it is easy to move, it can be placed by the net or another position that suits the training session.

Considering the measurement accuracy, the SCs will bounce when it hits the ground, affecting the accuracy of the reading. However, the SCs were placed approximately 0.1m further back to compensate for this. Consequently, the reported results are likely underestimated, with an assumed margin of error of +- 0.2m. Regardless, few people can consistently and accurately throw an SC between 4-6 meters, which the machine can do with minimal error. The machine shows promise, with only a few more modifications needed for fail-free usability, which will be discussed in the next chapter.

12.2 Discussion And Reflection

The SCL prototype has largely fulfilled its intended objectives and goals. The files for all the parts are open-source and available for further modification and uploaded on Google Drive [89]. These files are uploaded in STL format, ready for direct printing, and SLDPRT format, compatible with programs such as SW, a subscription-based software, and AutoCAD 360, which offers a free version CAD software.

In terms of cost-effectiveness, the SCL prototype provides a compelling alternative to more costly options priced at NOK 21,000 or NOK 140,000. As detailed in Table 13, the prototype's estimated cost is roughly NOK 6,700. This significant cost reduction and the potential for further development could render the SCL as functional as a more advanced machine.

The prices for each component were obtained directly from sellers and served only as estimates and guidance. For instance, it is not necessary to purchase Ratrig's 2040 aluminium profile, bolts and nuts - these were selected due to one store nearby extensive product range. Other suppliers might be cheaper on platforms such as eBay or Amazon. Furthermore, the 2040 profile size is a standard dimension, aligning with the thesis's objective to utilise standard parts for the design. This flexible approach to sourcing allows for cost optimisation and adaptability depending on the specific circumstances and availability of the builder.

The design of the SCL is designed to simplify assembly. Primarily, all parts are designed to fit up to the fixed attachment, requiring only two types of Allen keys (3mm and 2.5mm) to assemble the entire SCL, provided that the same size of screws are used. Additionally, threaded inserts have eased the assembly and disassembly of components such as microcontrollers, the tube and the Arduino mounting plates.

There is no requirement to screw directly into the plastic, thus increasing the ease of assembly and disassembly without concerns about the strength and grip of the screws. Also, it is possible to put the whole SCL in a training bag, which makes it easy to store.

However, while reflecting on the thesis's outcomes, it's relevant to acknowledge areas where improvements can be made. Specifically, the launch distance and reliability goals didn't fully align with the performance goals (1.4). The length of the shoots is assumed to come from the wheels not getting a good enough grip on the SC. It is also assumed that using sandpaper or something to increase friction may be a solution, and higher speed if it can be done. If not, then new wheels may need to be examined and tested.

12.2.1 Jams

During the testing phase, three instances of jamming were observed, as detailed in Table 16 and Figure 117. Addressing and correcting these jams are the highest priority, given their undesirable occurrence. The first two types of jams are considered easily fixable with minor adjustments since they occurred due to the feeding mechanism trapping the cork of the SC between the arm and the tube, specifically between the shuttle retainer and the pusher. Two areas were identified where the SC cork could get stuck, circled in red displayed Figure 119.

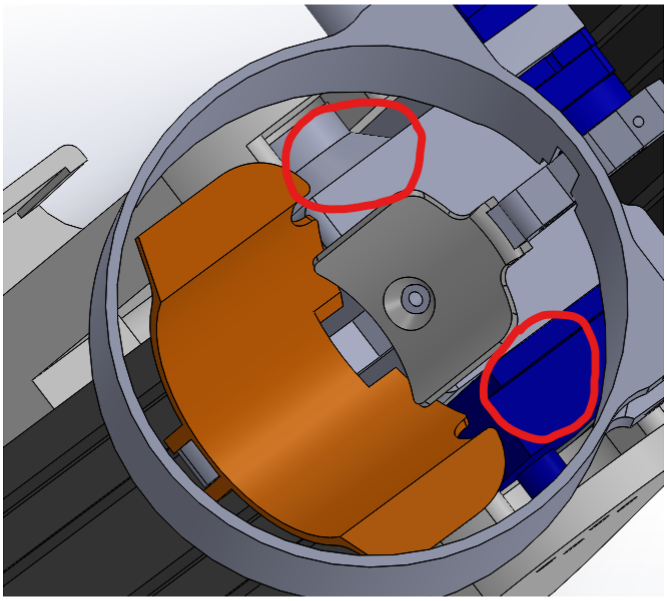


Figure 119: Inside the feeding mechanism showing where the cork can get stuck.

This problem may seem straightforward for those familiar with CAD and SW. However, not all users possess these skills, and the time constraints on updates after testing did not allow for a solution to be implemented. However, a possible solution can be to apply tape from the top of the shuttle retainer to the top of the pusher, as illustrated in Figure 120, and then mark where the tape hits the tube when the arm moves.

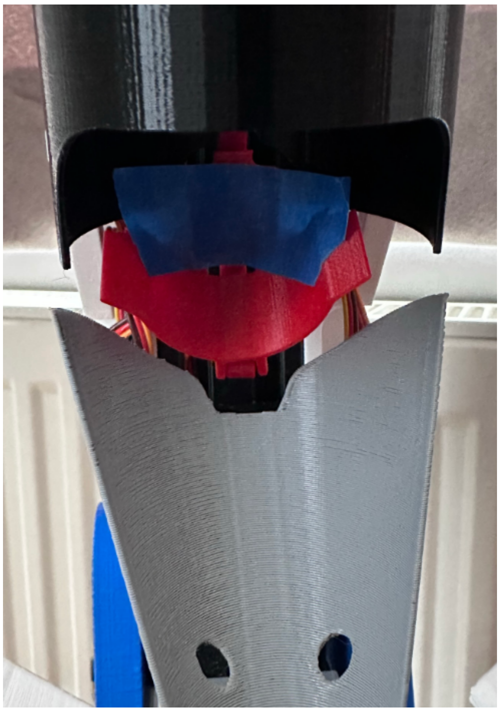


Figure 120: Example solution where the tape should be placed.

After applying the tape to resemble a cup and marking the tube, a knife can trim and remove material around the marked area. This measure is necessary by design, allowing the shuttle retainer and pusher to pass the cut-out in the tube, as shown in Figure 121.

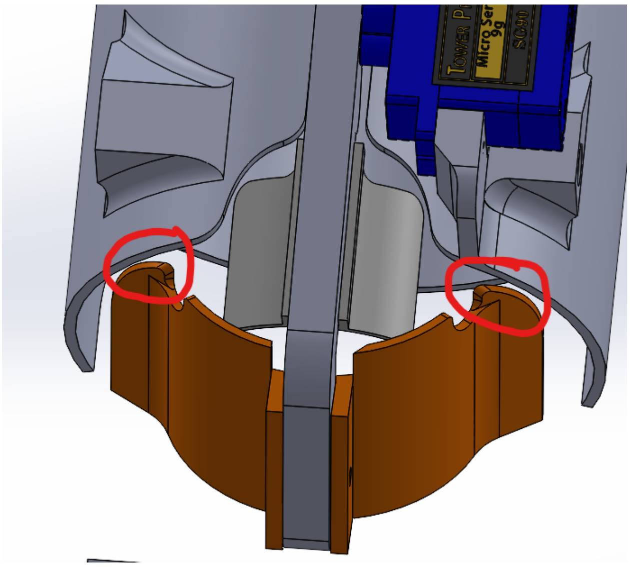


Figure 121: The tube with feeding mechanism at its tight spot.

This solution should be achievable without significantly compromising the strength of the construction, even though the motor mount shown on the top right in Figure 121 might be in the way. After removing the material, the tape may still be attached and prevent any SCs from jamming between the tube and the arm, effectively solving the problem.

Alternatively, the shuttle retainer and pusher could be merged into a single part to eliminate the gaps. This approach wasn't pursued initially due to smaller parts', faster printing and testing times, but it could be considered for future modifications.

The third type of jam occurred when the extension tube came loose. A possible solution involves expanding the tube to create a seamless transition with a ledge. This would require a modification of the tube's diameter at the top to create the ledge, providing a seamless transition as illustrated in Figure 122.

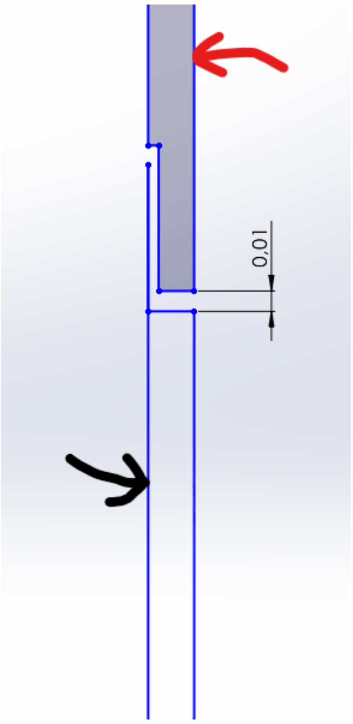


Figure 122: Example: cross-section of the tube and the extender tube.

In this cross-section, the black arrow points to the tube, which could be made thicker at the top, with a ledge added for the extender tube (marked with a blue arrow) to press against, ensuring a secure fit.

After making these modifications, a new experimental test can be carried out. If it manages hundreds of shots without jams, it can be considered to have solved the problem, and the SCL can thus be used for training the condition it is excluded from use in 30-degree tilt.

12.2.2 30-Degree Tilt Errors

During the experimental testing at a 30-degree tilt, the completion of the test was hindered due to many errors caused by the inclined of the guider. This limitation impacts the machine's versatility, a feature highlighted in the objectives outlined in section 1.3. The machine is capable of rotating and tilting, and the joystick can be used for that, but from the design now, it cannot shoot at 30 degrees and more without error. According to these objectives, the SCL should be capable of launching SCs at adjustable speeds and in different directions to mimic real-game scenarios, thus enhancing its usefulness for badminton practice.

To address this issue, several potential solutions have been considered. These include modifying the guider to be lower at one end and taller at the tube end or incorporating a mechanism that adjusts by the tilt's angle of the guider.

Another solution could involve changing the BLDC mount's surface to position the wheels at a 15-degree angle and reducing the tilt attachment's physical stops. This modification would mean that the vertical frame poses at -10 degrees resting and shoots 5 degrees out. Thus, a tilt of 20 degrees should theoretically yield a launch angle of 35 degrees. However, It should be noted that the HSE wheel cover may also need to be redesigned to prevent the wheel from hitting them, but this would need further testing for confirmation. This modification could thus achieve the objective of versatility.

12.2.3 Feeding Mechanism

After a long development process featuring extensive testing and slow-motion capture of its operation, the feeding mechanism has demonstrated its effectiveness. The 'Arm' mechanism, which has replaced the iris shutter, represents a considerable improvement that aligns with the thesis's ease of assembly objective (1.3). The arm mechanism requires three screws and three components to assemble, countering the iris shutter, which needs the perfect alignment of twelve dual-sided blades for smooth functionality. The iris shutter must also be printed with a 0.4 nozzle and could require extra work with a file to make the parts fit [8], while the arm only has one specific direction it needs to be printed in.

However, a limitation of the arm mechanism has been noted. The arm is only compatible with feather SCs, causing some plastic and synthetic SCs to be incompatible. A potential solution to this issue might involve using arm V.1.2 with full spikes to accommodate non-feather SCs, but this proposal demands further testing. Moreover, it implies that someone would have to switch the arms, which might not always be ideal if the SCL is loaded with different types of SCs. An alternate approach could be a hybrid of the arm and iris mechanisms, potentially allowing for a broader compatibility with various SCs.

Maintaining only the base of the arm mechanism and substituting the needle with the iris shutter could allow the system to accommodate all types of SCs. However, in pre-project testing, the friction caused by the blades of the iris shutter resulted in the failure and short-circuiting of an SG90 motor. This indicates the need for a more robust motor to handle the increased load. Consequently, the motor mount would need to be modified, and the tube centred under the iris shutter would need to be adjusted. One solution is lubricating the surfaces, but getting grease on the

SCs may be undesirable. Alternatively, is it possible to use existing grip mechanisms compatible with SG90 motors as inspiration, and then redesign the mechanism to perform the same role as an iris shutter. These solutions should be able to use all types of SCs and would be one modification prioritised for further development.

12.3 Considerations for Future Developments

The current SCL prototype represents significant strides towards fulfilling the thesis objectives and goals, but there are several areas for further investigation and modifications to realise these aims fully.

One area to note is mechanical reliability. Encountered issues, such as jams within the SCL, underline the need to find reliable solutions to these operational conditions. Thus, the primary focus of future SCL developments should be on modifications or options that address these issues.

The feeding mechanism is the main source of jams. A complete overhaul could be an effective solution, and a comprehensive redesign could offer a durable solution to these jams. A potential approach could be the design of an increased capacity SC magazine, capable of holding 20-40 SCs, followed by the redesign of the feeding mechanism. This approach differs from the existing SCL design methodology, starting from the wheels and building upwards to the feeding mechanism and then the magazine.

Further, integrating automation into the SCL could lead to considerable advancements, including components, such as wheel speed sensors that can control the speed. This may facilitate accurate prediction and control of the trajectory and landing point of SCs and could be done with statistics where speeds, tilt and rotation are recorded regarding where the SC lands. This integration of automation would be beneficial as it takes more focus away from controlling the machine and more focus on the training itself

Safety enhancements are another significant area for future consideration. By upgrading the SCLs safety mechanisms, users could safely be in closer proximity during operation, thus reducing the potential for accidents. These improvements could comprise a more robust protective casing, the integration of motion sensors, or the inclusion of an emergency stop function.

The development of an application also holds considerable potential. This could provide a user-friendly interface for controlling various aspects of the SCL, from modifying launch parameters to activating safety measures. Further, the application could offer data collection and analysis capabilities, adding another layer of usefulness to the SCL.

These samples of possible future work demonstrate the vast potential as an open-source the SCL have. As more individuals and organisations engage with the SCL, its design and features are anticipated to evolve continuously, driven by feedback, innovative solutions, and the community's unwavering commitment to improving the SCL.

13 Conclusion

The primary objective of this thesis was to develop a SCL that could effectively replace hand-feeding in badminton training. This objective has been successfully met, providing an automated, affordable, and versatile solution.

The SCL prototype, estimated to a cost of NOK 6,700, provides a significantly more economical solution than controversial models priced between NOK 21,000 to 140,000. Furthermore, the design around two lightweight 2040 aluminium profiles has enhanced portability, construction strength and storage ease, potentially fitting the whole SCL into a standard training bag.

The design and development of the SCL are based on an open-source model, marking an approach to fostering community-driven development by utilising widely available parts, and cost-effective materials, including parts fabricated through 3D printing technology. A balanced approach has been successfully demonstrated to create an affordable and functional SCL. This contribution can be seen as a significant foundation for the future evolution of sports equipment or similar types.

The SCL exhibits comparable functionality to higher-end models in performance and versatility. Although it did not reach the projected maximum launch distance, the SCL demonstrated consistent performance within a range beneficial for beginners and technique-focused intermediate badminton training. Its innovative pan/tilt mechanism provides varying launch directions and adjustable speeds with the ESCs for the BLDCs controlled by a microcontroller, highlighting its versatility and enhancing its flexibility for future developments.

The rapid prototyping approach used in this thesis, alongside CAD software and 3D printing, proved to be effective and efficient, enabling quick design iterations and testing. Using an iPhone 14 Pro for slow-motion video analysis demonstrated the potential of widely available tools for detailed real-time study and refinement of the SCL's operation.

In conclusion, this thesis successfully demonstrates the feasibility of creating accessible, affordable, and versatile badminton training SCLs using the methods employed, particularly 3D printing technology.

Bibliography

[1] F. Whittemore. *Badminton Equipment Regulations*. Oct. 2022. URL: https://www.sportsrec.com/5590266/badminton-equipment-regulations. (accessed: 06.03.2023).

[2] R. Bryhn and C. Francke. *badminton*. URL: https://snl.no/badminton. (accessed: 06.03.2023).

[3] *Automatic Badminton Service Machine Badminton Ball Pitching Tool for Coachesr*. https://www.ebay.com/itm/155456698974. 2023. Accessed: 2023-05-28.

[4] Boldmaskine. *S3025 Boldmaskine til Badminton*. 2023. URL: https://boldmaskine.dk/produkt/s3025-boldmaskine-til-badminton/. Accessed: 2023-05-28.

[5] Boldmaskine. *i4.0 Boldmaskine Badminton*. 2023. URL: https://boldmaskine.dk/produkt/i4-0-boldmaskine-badminton/. 2023-05-28.

[6] Kickstarter. *BADDy: Open Source Badminton Robot*. 2017. URL: https://www.kickstarter.com/projects/401702470/baddy-open-source-badminton-robot?ref=card. Accessed: 12.05.2023.

[7] H. Rishovd and A. S. S. Kristiansen. 'Improvements of the shuttlecock launcher robot BADDY'. MA thesis. Norwegian University of Science and Technology (NTNU), 2021.

[8] K. Halvorsen. 'Designing and development of a new shuttlecock launcher'. 2022.

[9] *Raspberry Pi 4 Model B*. URL: https://thepihut.com/products/raspberry-pi-4-model-b?src=raspberrypi&variant=41005997392067. (accessed: 15.04.2023).

[10] T. H. N. (i Østfold). *terminal (IT)*. URL: https://snl.no/terminal_-_IT. (accessed: 03.05.2023).

[11] E. Upton and G. Halfacree. *Raspberry Pi User Guide*. 4th. Hoboken, NJ: John Wiley & Sons, 2020.

[12] P. B. Andersen. *mekatronikk*. 2020. URL: https://snl.no/mekatronikk. (accessed: 15.04.2023).

[13] E. Upton and G. Halfacree. *Raspberry Pi User Guide*. 4th ed. Wiley, 2016, pp. 199–202. URL: https://dn.odroid.com/IoT/other_doc.pdf.

[14] P. B. Andersen and I. M. Liseter. *Kretskort*. 2023. URL: https://snl.no/kretskort. Accessed: 09.05.2023.

[15] D. Ibrahim. 'Chapter 8 - Advanced PIC32 Projects'. In: *Designing Embedded Systems with 32-Bit PIC Microcontrollers and MikroC*. Ed. by D. Ibrahim. Oxford: Newnes, 2014, pp. 371–382. ISBN: 978-0-08-097786-7. DOI: https://doi.org/10.1016/B978-0-08-097786-7.00008-7. URL: https://www.sciencedirect.com/science/article/pii/B9780080977867000087.

[16] P. B. Andersen. *automatisering*. URL: https://snl.no/automatisering. (accessed: 09.03.2023).

[17] B. B. Larsen. *mikrokontroller*. 2021. URL: https://snl.no/mikrokontroller. (accessed: 15.04.2023).

[18] E. Upton and G. Halfacree. *Raspberry Pi User Guide*. 4th. Hoboken, NJ: John Wiley & Sons, 2020. ISBN: 978-1-119-68160-5.

[19] E. Upton and G. Halfacree. *Raspberry Pi User Guide*. 4th. Hoboken, NJ: John Wiley & Sons, 2020. Chap. 14.

[20] M. Banzi. *Getting Started with Arduino*. 3rd. Sebastopol, CA: Maker Media, Inc., 2015. ISBN: 978-1-4571-8696-4.

[21] Arduino. *Servo library reference*. 2021. URL: https://www.arduino.cc/en/reference/servo. (accessed: 05.03.2023).

[22] A. B. (NTNU). *lodding*. 2018. URL: https://snl.no/lodding. (accessed: 11.05.2023).

[23] A. Almar-Næss. *loddebolt*. 2018. URL: https://snl.no/loddebolt. (accessed: 11.05.2023).

[24] L. Mæhlum. *3D-printingi*. 2021. URL: https://snl.no/3D-printingi. (accessed: 16.04.2023).

[25] Y. L. Yap et al. 'Fused deposition modeling of novel scaffold architectures for tissue engineering applications'. In: *Rapid Prototyping Journal* 8.4 (2002), pp. 248–257. DOI: 10.1108/13552540210441166. (accessed: 03.05.2023).

[26] K. V. Wong and A. Hernandez. 'A Review of Additive Manufacturing'. In: *International Scholarly Research Notices* 2012 (2012), pp. 1–10. DOI: 10.5402/2012/208760. URL: https://www.hindawi.com/journals/isrn/2012/208760/. (accessed: 03.05.2023).

[27] Prusa. *Original Prusa i3 MK3S+ 3D printer*. URL: https://www.prusa3d.com/product/original-prusa-i3-mk3s-3d-printer-3. (accessed: 28.03.2023).

[28] Creality. *Ender-3 Pro 3D Printer*. URL: https://www.creality.com/products/ender-3-pro-3d-printer. (accessed: 28.03.2023).

[29] Creality. *Ender-3 Pro 3D Printer*. URL: https://store.creality.com/products/ender-3-pro-3d-printer?official-website-product-ender-top=&spm=..product_baabd2f7-66bd-478f-b989-a3def0337231.nav_link_store_1.1. (accessed: 28.03.2023).

[30] P. Research. *PrusaSlicer*. https://www.prusa3d.com/prusaslicer/. 2023. Accessed: 2023-05-28.

[31] H. Kondo. *3D Printing First Layer Problems: How to Make It Perfect*. 2022. URL: https://all3dp.com/2/3d-printing-first-layer-problems-how-to-make-it-perfect/. (accessed: 08.04.2023).

[32] E. G. Che Simons. *Extruder Calibration: How to Calibrate E-Steps*. 2023. URL: https://all3dp.com/2/extruder-calibration-calibrate-e-steps/. (accessed: 08.04.2023).

[33] J. O'Connell. *Z Banding / Z Wobble: How to Prevent It*. 2022. URL: https://all3dp.com/2/3d-printer-z-banding/. (accessed: 08.04.2023).

[34] All3DP. *All3DP*. 2023. URL: https://about.all3dp.com/. 2023-05-28.

[35] E. Rossen. *DAK*. URL: https://snl.no/DAK_-_IT. (accessed: 13.04.2023).

[36] R. W. M. Kolbein Bell. *elementmetoden*. URL: https://snl.no/elementmetoden. (accessed: 13.04.2023).

[37] P. M. Kurowski. *Engineering Analysis with SolidWorks Simulation 2017*. Mission: SDC Publications, 2017. ISBN: 9781630570767.

[38] A. Dørum. *Maskindeler 1*. Akademika, 2001.

[39] T. Aambø. *topologi*. 2023. URL: https://snl.no/topologi. (accessed: 15.04.2023).

[40] SolidWorks. *Everything You Need to Know about Topology Optimization*. 2019. URL: https://www.solidworks.com/media/everything-you-need-know-about-topology-optimization. (accessed: 15.04.2023).

[41] SolidWorks. *SIMP Method for Topology Optimization*. 2023. URL: https://help.solidworks.com/2023/English/SolidWorks/cworks/c_simp_method_topology.htm. (accessed: 15.04.2023).

[42] S. R. Rajpurohit, H. K. Dave and M. Bodaghi. 'Classical laminate theory for flexural strength prediction of FDM 3D printed PLAs'. In: *Materials Today: Proceedings* (2023). ISSN: 2214-7853. DOI: https://doi.org/10.1016/j.matpr.2023.03.310. URL: https://www.sciencedirect.com/science/article/pii/S2214785323013561.

[43] *Prototyp*. 2021. URL: https://snl.no/prototyp. Accessed: 14.05.2023.

[44] S. Kalpakjian and S. R. Schmid. *Manufacturing Engineering and Technology*. Singapore: Prentice Hall, 2010. Chap. 1. ISBN: 9810681445.

[45] L.-M. Chen, Y.-H. Pan and Y.-J. Chen. *A study of Shuttlecock's trajectory in badminton*. Dec. 2009. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3761540/.

[46] B. Zhang et al. *Suppressing resonant vibrations using nonlinear springs and dampers*. June 2007. URL: https://eprints.whiterose.ac.uk/74614/.

[47] A. Varshney and B. Dwivedi. 'Performance analysis of a BLDC drive under varying load'. In: July 2016, pp. 1–4. DOI: 10.1109/ICPEICES.2016.7853626.

[48] D. Macri. *Designing an ESC Module to Control Drone Motors*. 2021. URL: https://www.powerelectronicsnews.com/designing-an-esc-module-to-control-drone-motors/. Accessed: 2023-05-28.

[49] Daniel. *Control the Basic ESC with a Potentiometer and Arduino*. https://bluerobotics.com/learn/guide-for-controlling-the-basic-esc-with-a-potentiometer/. Accessed: 12.05.2023.

[50] japersik. *Servo Pan Tilt - Raspberry Pi camera*. 2023. URL: https://www.thingiverse.com/thing:4710301. (accessed: 20.03.2023).

[51] *Power HD WH-20KG HV WP 20kg/0.08s*. URL: https://www.elefun.no/p/prod.aspx?v=50882. (accessed: 20.04.2023).

[52] *Savöx Servo SC-1256TG+ - 0.15 speed/20kg*. URL: https://www.elefun.no/p/prod.aspx?v=59274. (accessed: 20.04.2023).

[53] *ANNIMOS 20KG Digital Servo High Torque Full Metal Gear Waterproof for RC Model DIY, DS3218MG,Control Angle 270° red*. URL: https://www.amazon.com/ANNIMOS-Digital-Waterproof-DS3218MG-Control/dp/B076CNKQX4. (accessed: 20.04.2023).

[54] *DS3218 20KG Servo and DS3218PRO 20KG Servo*. URL: https://www.amazon.com/DS3218-20KG-Servo-DS3218PRO/dp/B08L6VHB9F. (accessed: 20.04.2023).

[55] *SKF KULELAGER AKSIAL 51105*. URL: https://www.xn--verkty-fya.no/51105. (accessed: 22.04.2023).

[56] *uxcell 51105 Thrust Ball Bearings 25mm x 42mm x 11mm Chrome Steel ABEC3 Single Row Roller*. URL: https://www.amazon.com/uxcell-Single-Direction-Thrust-Bearings/dp/B07G847BB5/ref=sr_1_4?content-id=amzn1.sym.9575273b-ecd8-4648-9bf0-15f20c657e0a&keywords=51105%2Bthrust%2Bbearing&pd_rd_r=0e6d8914-4d1e-499b-be30-3943d1c7e38f&pd_rd_w=oolkh&pd_rd_wg=rU77f&pf_rd_p=9575273b-ecd8-4648-9bf0-15f20c657e0a&pf_rd_r=CQ3YEW0QB13STZ3KQ6X6&qid=1682251615&sr=8-4&th=1. (accessed: 23.04.2023).

[57] *1 pcs high quality Planar plane thrust bearing 51100 51101 51102 51103 51104 51105 51106 51107*. URL: %5Curl%7Bhttps://www.aliexpress.com/item/1005004838039678.html?pdp_npi=2%40dis%21EUR%21%E2%82%AC%201.06%21%E2%82%AC%200.85%21%21%21%21%21%402103011216822513120807127ef0db%21120000030686737260%21btf&_t=pvid:d1fe95c8-182d-4a9d-a6ef-9d8fc673cf66&afTraceInfo=1005004838039678__pc__pcBridgePPC__xxxxxx__1682251312&spm=a2g0o.ppclist.product.mainProduct%7D. (accessed: 23.04.2023).

[58] *51105*. URL: https://www.skf.com/in/products/rolling-bearings/ball-bearings/thrust-ball-bearings/productid-51105. (accessed: 22.04.2023).

[59] Digital Impuls. *Raspberry Pi4 4GB Kit Avansert PSU Alu Fan Case NOOBS 32GB 2x HDMI*. https://www.digitalimpuls.no/okdo/145999/raspberry-pi4-4gb-kit-avansert-psu-alu-fan-case-noobs-32gb-2x-hdmi. 2021. Accessed: 09.05.2023.

[60] Digital Impuls. *Arduino Uno - ATmega328 MCU Board Rev. 3, 8-bit, 14 digitale I/O-pins, 6 analoge*. https://www.digitalimpuls.no/hovedkort/127062/arduino-uno-atmega328-mcu-board-rev-3-8-bit-14-digitale-i-o-pins-6-analoge. 2021. Accessed: 09.05.2023.

[61] Digital Impuls. *Raspberry Pi Pico - Mikrokontroller bygd rundt RP2040*. https://www.digitalimpuls.no/raspberry-pi/148242/raspberry-pi-pico-mikrokontroller-bygd-rundt-rp2040. 2021. Accessed: 09.05.2023.

[62] Digital Impuls. *Nano Utviklingskort m/USB-kabel - Arduino Nano 3.0 kompatibelt, ATmega328*. https://www.digitalimpuls.no/hovedkort/148871/nano-utviklingskort-m-usb-kabel-arduino-nano-30-kompatibelt-atmega328. 2021. Accessed: 09.05.2023.

[63] I. G. M. A. Rosvold. *batteri*. 2023. URL: https://snl.no/batteri. (accessed: 18.05.2023).

[64] *eBay: 362672759125*. https://www.ebay.co.uk/itm/362672759125. Accessed: 12.05.2023.

[65] R. Nelson. *Mini Joystick Housing : With Space For Arduino Micro Pro (Leonardo)*. https://www.thingiverse.com/thing:3250017. Accessed: 12.05.2023.

[66] N. E. KOMITE. *Elektriske lavspenningsinstallasjoner NEK 400:2018 utgave 6*. 2018.

[67] S. Taherkhani et al. 'Aerodynamic Drag Reduction of Emergency Response Vehicles'. In: *Advances in Automobile Engineering* 4 (Sept. 2015). DOI: 10.4172/2167-7670.1000122. URL: https://www.researchgate.net/publication/283123078_Aerodynamic_Drag_Reduction_of_Emergency_Response_Vehicles. Accessed: 14.05.2023.

[68] I. G. K. (OsloMet). *PLA*. 2023. URL: https://snl.no/PLA. Accessed: 2023-05-28.

[69] G. Slump. *PLA vs ABS vs PETG: Differences Compared*. 2022. URL: https://all3dp.com/2/pla-vs-abs-vs-petg-differences-compared/ (visited on 30/05/2023). Accessed: 2023-05-28.

[70] Elefun. *SLOT 2020 - Black Anodized 250mm*. 2023. URL: %7Bhttps://www.elefun.no/p/prod.aspx?v=50268%7D. Accessed: 2023-05-28.

[71] Elefun. *Low Profile Screws - M5x8mm - 25 Pack*. 2023. URL: %7Bhttps://www.elefun.no/p/prod.aspx?v=50294%7D. Accessed: 2023-05-28.

[72] Elefun. *T-Nuts - Drop In M5 for 2020- (25 Pack)*. 2023. URL: %7Bhttps://www.elefun.no/p/prod.aspx?v=52490%7D. Accessed: 2023-05-28.

[73] Clas Ohlson. *Maskinskrue*. 2023. URL: %7Bhttps://www.clasohlson.com/no/Maskinskrue/p/Pr111133640%7D. Accessed: 2023-05-28.

[74] Polyalkemi. *M3 Skrukit Hex Skrue Kit Med Runde Hoder 80deler*. 2023. URL: %7Bhttps://polyalkemi.no/produkt/m3-skrukit-hex-skrue-kit-med-runde-hoder-80deler/%7D. Accessed: 2023-05-28.

[75] Fruugo Norge. *330 stk gjengeinnsats pressemutter M2 M3 M4 M5 innvendig gjenge knurled nuts messing embedding nuts for plastdeler i 3D printede deler*. 2023. URL: %7Bhttps://www.fruugonorge.com/330-stk-gjengeinnsats-pressemutter-m2-m3-m4-m5-innvendig-gjenge-knurled-nuts-messing-embedding-nuts-for-plastdeler-i-3d-printede-deler/p-167265773-357044384%7D. Accessed: 2023-05-28.

[76] Digital Impuls. *Hook Up Ledning 6 stk x 7.62m 22AWG Solidkjerne*. 2023. URL: %7Bhttps://www.digitalimpuls.no/the-pi-hut/152385/hook-up-ledning-6-stk-x-7-62m-22awg-solidkjerne%7D. Accessed: 2023-05-28.

[77] Digital Impuls. *Kemo Electronic Stripboard Epoxy 1.5mm 100mm x 100mm*. 2023. URL: %7Bhttps://www.digitalimpuls.no/diverse/150700/kemo-electronic-stripboard-epoxy-15mm-100mm-x-100mm%7D. Accessed: 2023-05-28.

[78] Artigere Liv. *Male Pin Header 40pin - 10 stk*. 2023. URL: %7Bhttps://artigereliv.no/produkt/elektronikk/mikrokontrollere/sensorer-og-tilbehor/male-pin-header-40pin-10-stk%7D. Accessed: 2023-05-28.

[79] Artigere Liv. *Female Pin Header 40pin - 10 stk*. 2023. URL: %7Bhttps://artigereliv.no/produkt/elektronikk/mikrokontrollere/sensorer-og-tilbehor/female-pin-header-40pin-10-stk%7D. Accessed: 2023-05-28.

[80] Elfa Distrelec. *Rotary Potentiometer, 10kOhm, 50mW, ±20, Bourns, PTV09A-4020U-B103*. 2023. URL: %7Bhttps://www.elfadistrelec.no/en/rotary-potentiometer-10kohm-50mw-20-bourns-ptv09a-4020u-b103/p/30115467%7D. Accessed: 2023-05-28.

[81] Elektroimportøren. *Wago 221-413 3-leder Transparent - 5stk*. 2023. URL: %7Bhttps://www.elektroimportoren.no/wago-221-413-3-leder-transparent-5stk/59054/Product.html?Event=brandlist%7D. Accessed: 2023-05-28.

[82] Modellflybutikken. *SG90 Micro Servo*. 2023. URL: %7Bhttps://modellflybutikken.no/produkt/servoer/sg90-micro-servo/%7D. Accessed: 2023-05-28.

[83] Gla'pris. *Volante 36 Hjul - 4pk*. 2023. URL: %7Bhttps://www.glapris.no/produkt/volante-36-hjul-4pk/%7D. Accessed: 2023-05-28.

[84] 3DNet. *3DNet PLA 1.75 - NY*. 2023. URL: %7Bhttps://3dnet.no/collections/1-75-filament/products/3dnet-pla-1-75-ny?variant=34430795284644%7D. Accessed: 2023-05-28.

[85] Global Petrol Prices. *Electricity Prices around the world*. 2023. URL: %7Bhttps://www.globalpetrolprices.com/electricity_prices/%7D. Accessed: 14.05.2023.

[86] Elefun. *4s 450mAh - 70C - CNHL Ministar XT30*. 2023. URL: %7Bhttps://www.elefun.no/p/prod.aspx?v=38813%7D. Accessed: 2023-05-28.

[87] Amazon UK. *ERYUE DX2205 2300KV Brushless Motor, 2CW 2CCW DX2205 2300KV Brushless Motor For QAV250 QAV300 FPV Racing Drone Multicopter*. 2023. URL: %7B%5Curl%7Bhttps://www.amazon.co.uk/ERYUE-Brushless-Reverse-Rotating-Multicopter/dp/B0BW89SX3B/ref=sr_1_33?crid=20L5ND4EFS6KJ&keywords=DX2205+2300KV+Brushless+Motor&qid=1685375344&sprefix=dx2205+2300kv+brushless+motor%2Caps%2C96&sr=8-33%7D%7D. Accessed: 2023-05-28.

[88] AliExpress. *AliExpress Product*. 2023. URL: %7B%5Curl%7Baliexpress.com/item/1005004412855782.html?pdp%5C_npi=2%40dis%21EUR%21%E2%82%AC7.06%21%E2%82%AC4.24%21%21%21%21%21%402101c5ac16853759356561512e343d%21120000029093659774%21btf%5C&_t=pvid:aa9dee91-df9a-4448-9562-a051749bdde0&afTraceInfo=1005004412855782%5C_%5C_pc%5C_%5C_pcBridgePPC%5C_%5C_xxxxxx%5C_%5C_1685375936&spm=a2g0o.ppclist.product.mainProduct%7D%7D. Accessed: 2023-05-28.

[89] K. Halvorsen. *$SCL_2$023*. 2023. URL: %5Curl%7Bhttps://drive.google.com/drive/folders/1WsZqK37Nylb0FO1MIBV6fRCzvC8VeU-4?usp=sharing%7D. Accessed: 2023-05-28.

Appendix

A Full ESC Python code from pre-project

Listing 8: Python code from pre-project [8].

```python

import os
import time
os.system ("sudo pigpiod")
time.sleep(1)
import pigpio

ESC_1 = 2
ESC_2 = 17

Count_ESC = [ESC_1,ESC_2,]

pi = pigpio.pi();

for E in Count_ESC:
    pi.set_servo_pulsewidth(E, 0)

max_value = 2500
min_value = 1000

print("Disconnect battery.")

def calibrate():
    for E in Count_ESC:
        pi.set_servo_pulsewidth(E, 0)
        time.sleep(1)
        pi.set_servo_pulsewidth(E, max_value)
    print("Connect battery and then press Enter")
    inp = input()
    if inp == '':
        for E in Count_ESC:
            pi.set_servo_pulsewidth(E, min_value)
        time.sleep(7)
        time.sleep(5)
        for E in Count_ESC:
            pi.set_servo_pulsewidth(E, 0)
        time.sleep(2)
        print("Arming..")
        for E in Count_ESC:
            pi.set_servo_pulsewidth(E, min_value)
        time.sleep(1)
        control()

def control():

    time.sleep(1)
    speed = min_value
    print(" a : decrease speed & d : increase speed")
    print("␣q␣:␣decrease␣a␣lot␣&␣e␣:␣increase␣a␣lot")
    while True:
        for E in Count_ESC:
            pi.set_servo_pulsewidth(ESC_1, speed - 90)
            pi.set_servo_pulsewidth(ESC_2, speed)
        inp = input()

        if inp == "q":
            speed -= 50
            print("speed␣=␣%d" % speed)
        elif inp == "e":
            speed += 50
            print("speed␣=␣%d" % speed)
        elif inp == "d":
            speed += 5
            print("speed␣=␣%d" % speed)
        elif inp == "a":
            speed -= 5
            print("speed␣=␣%d" % speed)
        elif inp == "stop" or inp == "s":
            stop()
            break
        else:
            print("Press␣a,q,d␣or␣e")

def stop():
    for E in Count_ESC:
        pi.set_servo_pulsewidth(E, 0)
    pi.stop()

def main():
    print("chose␣an␣input,␣Cal␣for␣calibrate,␣c␣for␣control␣or␣s␣for␣stop")
    inp = input("")
    if inp == "cal":
        calibrate()
    elif inp == "c":
        control()
    elif inp == "s":
        stop()
main()
```

B Final Adjusted Code for the SCL

Listing 9: Code to control the SCL with a wired joystick and button

```
#include <Servo.h>

// Button and feeding servo configuration
const int buttonPin = 3;
const int feedingServoPin = 11;

const unsigned long debounceDelay = 40;
int startPosition = 0;
int targetPosition = 75;

// Joystick and tilt/pan servos configuration

// Constants for tilt
const int tiltServoPin = 10;
const int tiltJoystickPin = A3;

const int minTiltAngle = 0;
const int maxTiltAngle = 44;
const int startTiltAngle = 5;

// Constants for pan
const int panServoPin = 13;
const int panJoystickPin = A2;

const int minPanAngle = 0;
const int maxPanAngle = 180;
const int startPanAngle = 90;

// ESC configuration
const int ESC_A_Pin = 12;
const int ESC_B_Pin = 9;

const int potAPin = A1;
const int potBPin = A0;

const int ESC_A_Min = 1080; // Min puls width
const int ESC_A_Max = 1170; // Max puls width

const int ESC_B_Min = 1130; // Min puls width
const int ESC_B_Max = 1253; // Max puls width
const int ESC_Arm_Delay = 2000;

// Servo speed
const unsigned long updateInterval = 20; // 20 milliseconds
const int angleChange = 2;

// Create servo objects
Servo feedingServo;
Servo tiltServo;
Servo panServo;
Servo ESC_A;
Servo ESC_B;
```

```

// Variables for button press handling
volatile bool buttonPressed = false;
unsigned long lastDebounceTime = 0;

// Variables to store the last servo angles
int lastTiltAngle = startTiltAngle;
int lastPanAngle = startPanAngle;

// Variable to store the last update time
unsigned long lastUpdateTime = 0;

void setup() {
  pinMode(buttonPin, INPUT_PULLUP);
  attachInterrupt(digitalPinToInterrupt(buttonPin), onButtonPress, FALLING);
  feedingServo.attach(feedingServoPin);
  feedingServo.write(startPosition);

  tiltServo.attach(tiltServoPin);
  panServo.attach(panServoPin);

  tiltServo.write(startTiltAngle);
  panServo.write(startPanAngle);

  ESC_A.attach(ESC_A_Pin);
  ESC_B.attach(ESC_B_Pin);

  // Arm the ESCs
  ESC_A.writeMicroseconds(ESC_A_Min);
  ESC_B.writeMicroseconds(ESC_B_Min);
  delay(ESC_Arm_Delay);

  Serial.begin(9600);
}

void loop() {
  if (buttonPressed) {
    if ((millis() - lastDebounceTime) > debounceDelay) {
      buttonPressed = false;
      feedingServo.write(startPosition+5);
      delay(800);
      feedingServo.write(targetPosition);
      delay(200);
      feedingServo.write(targetPosition-12);
      delay(200);
      feedingServo.write(targetPosition);
      delay(1200);
      feedingServo.write(startPosition);
    }
  }

  int tiltJoystickValue = analogRead(tiltJoystickPin);
  int panJoystickValue = analogRead(panJoystickPin);

  unsigned long currentTime = millis();

  if (currentTime - lastUpdateTime >= updateInterval) {
    if (tiltJoystickValue > 800) {
      lastTiltAngle = min(lastTiltAngle + angleChange, maxTiltAngle);
    } else if (tiltJoystickValue < 200) {
      lastTiltAngle = max(lastTiltAngle - angleChange, minTiltAngle);
    }
    tiltServo.write(lastTiltAngle);

        if (panJoystickValue > 800) {
      lastPanAngle = min(lastPanAngle + angleChange, maxPanAngle);
    } else if (panJoystickValue < 200) {
      lastPanAngle = max(lastPanAngle - angleChange, minPanAngle);
    }
    panServo.write(lastPanAngle);

    lastUpdateTime = currentTime;

    Serial.print("Tilt␣angle:␣");
    Serial.print(lastTiltAngle);
    Serial.print(",␣Pan␣angle:␣");
    Serial.println(lastPanAngle);
  }

  // Read potentiometer values
  int potAValue = analogRead(potAPin);
  int potBValue = analogRead(potBPin);

  // Map potentiometer values to ESC range
  int ESC_A_Value = map(potAValue, 0, 1023, ESC_A_Min, ESC_A_Max);
  int ESC_B_Value = map(potBValue, 0, 1023, ESC_B_Min, ESC_B_Max);

  // Write the mapped values to the ESCs
  ESC_A.writeMicroseconds(ESC_A_Value);
  ESC_B.writeMicroseconds(ESC_B_Value);

  // Print the mapped values for debugging purposes
  Serial.print("ESC␣A␣Value:␣");
  Serial.print(ESC_A_Value);
  Serial.print(",␣ESC␣B␣Value:␣");
  Serial.println(ESC_B_Value);

  delay(15);
}

void onButtonPress() {
  if ((millis() - lastDebounceTime) > debounceDelay) {
    lastDebounceTime = millis();
    buttonPressed = true;
  }
}
```

C Experimental Data

C.1 Results of the SCL Testing with 0-degree Tilt

Table 17: Results of the SCL testing with 0-degree tilt.

Test No.	Tilt-Angle	Shuttlecock No.	Distance (m)	Double Shots (0/1)	No Shots (0/1)	Jamming (0/1)	Notes
1	0	1	3.10	1			
1	0	1	3.10	1			
1	0	2	0.20				
1	0	3	5.50				
1	0	4	6.00				
1	0	5	5.00				
1	0	6	5.50				
1	0	7	5.10				
1	0	8	4.50				
1	0	9	5.60				
1	0	10	5.00				
1	0	11	5.10				
2	0	2	5.10				
2	0	1	5.20	1			
2	0	6	5.30				
2	0	10	5.40				
2	0	5	5.30				
2	0	11	4.00				
2	0	7	5.40				
2	0	4	4.90				
2	0	3	5.4				
2	0	9	5.4				
2	0	8	5.5				
3	0	11	0.2	1			
3	0	4	0.2				
3	0	6	5				
3	0	5	5				
3	0	9	5.2				
3	0	7	4.6				

Continued on next page

Table 17 – continued from previous page

Test No.	**Tilt-Angle**	**Shuttlecock No.**	**Distance (m)**	**Double Shots (0/1)**	**No Shots (0/1)**	**Jamming (0/1)**	**Notes**
3	0	2	3.9				
3	0	3	5.2				
3	0	8	5.6				
3	0	10	5				
3	0	1	5.6				
4	0	4	5.2				
4	0	7	4.4				
4	0	2	4.4				
4	0	10	5.1				
4	0	6	5.2				
4	0	5	5.1				
4	0	3	5.8				
4	0	9	5.4				
4	0	11	4.4				
4	0	1	5.8				
4	0	8	5.4				
5	0	8	4.4				
5	0	9	4.9				
5	0	3	5				
5	0	1	4.9				
5	0	2	5				
5	0	11	5.3				
5	0	7	0	1	1	0	did not shoot on first attempt
5	0	5	4.4				
5	0	6	5.2		1		
5	0	4	5.1				
5	0	10	5.1				
6	0	1	4.9	1			
6	0	2	0.2				
6	0	3	4.9				
6	0	4	5.1				

Continued on next page

Table 17 – continued from previous page

Test No.	Tilt-Angle	Shuttlecock No.	Distance (m)	Double Shots (0/1)	No Shots (0/1)	Jamming (0/1)	Notes
6	0	5	5				
6	0	6	4.9				
6	0	7	4				
6	0	8	5.5				
6	0	9	5.5				
6	0	10	4.5				
6	0	11	5				
7	0	7	5				
7	0	2	4.4				
7	0	11	4.4				
7	0	6	5				
7	0	4	5				
7	0	9	5.1				
7	0	10	5.1				
7	0	5	5.5				
7	0	3	4.2				
7	0	8	5.5				
7	0	1	5.5				
8	0	2	5.3				
8	0	11	5				
8	0	4	5				
8	0	6	5.5				
8	0	7	5.5				
8	0	8	5.5				
8	0	5	5.2				
8	0	1	5				
8	0	10	5.3				
8	0	9	6				
8	0	3	5.3		1		
9	0	9	5.2				
9	0	11	5.3	1		1	
9	0	5	6				
9	0	8	5				

Continued on next page

Table 17 – continued from previous page

Test No.	**Tilt-Angle**	**Shuttlecock No.**	**Distance (m)**	**Double Shots (0/1)**	**No Shots (0/1)**	**Jamming (0/1)**	**Notes**
9	0	4	5.3				
9	0	7	6				
9	0	6	0		1		
9	0	10	5.1				
9	0	2	5.1				
9	0	3	5				
9	0	1	5.2				
10	0	5	5				
10	0	7	4.5				
10	0	6	5.2				
10	0	8	4.9				
10	0	3	5				
10	0	4	5.1				
10	0	1	4.9				
10	0	10	5.2				
10	0	11	0	1	1		
10	0	2	5				
10	0	9	5	1			
							End of table

C.2 Results of the SCL Testing with 10-degree Tilt

Table 18: Results of the SCL testing with 10-degree tilt.

Test No.	Tilt-Angle	Shuttlecock No.	Distance (m)	Double Shots (0/1)	No Shots (0/1)	Jamming (0/1)	Notes
1	10	1	5.2	1			
1	10	2	0				
1	10	3	5.1				
1	10	4	5.5				
1	10	5	4.1				
1	10	6	5.5				
1	10	7	6.1				
1	10	8	5.5				
1	10	9	5.1				
1	10	10	5.2				
1	10	11	6.2				
2	10	8	4.5				
2	10	4	5.2				
2	10	6	4.8	1			
2	10	11	0				
2	10	7	5.2				
2	10	2	2.7				
2	10	3	5.9				
2	10	9	3.3	1	1		
2	10	10	1.6				
2	10	1	6.1				
2	10	5	5.2		1		
3	10	3	6	1			
3	10	1	6				
3	10	6	5.1				
3	10	8	4.9				
3	10	11	5				
3	10	10	5				
3	10	7	5.8				
3	10	4	5.8				
3	10	5	4.9				

Continued on next page

Table 18 – continued from previous page

Test No.	Tilt-Angle	Shuttlecock No.	Distance (m)	Double Shots (0/1)	No Shots (0/1)	Jamming (0/1)	Notes
3	10	2	5.1				
3	10	9	5.8				
4	10	2	5.5	1			
4	10	5	0.2				
4	10	8	5				
4	10	6	5.5				
4	10	4	5.5				
4	10	3	4				
4	10	1	5.5				
4	10	10	5.5				
4	10	11	5.5				
4	10	7	5.5	1			
4	10	9	5.5				
5	10	1	5.1	1			
5	10	2	0.2				
5	10	3	5.3				
5	10	4	5.4				
5	10	5	4.6				
5	10	6	5.1				
5	10	7	5.5				
5	10	8	5.4				
5	10	9	5.3				
5	10	10	4.6				
5	10	11	5.4				
6	10	6	5.2				
6	10	1	5.3				
6	10	9	5.2				
6	10	3	5.3				
6	10	5	2.5				
6	10	10	5.8				
6	10	8	2.1	1			
6	10	11	5.3				
6	10	4	5				

Continued on next page

Table 18 – continued from previous page

Test No.	Tilt-Angle	Shuttlecock No.	Distance (m)	Double Shots (0/1)	No Shots (0/1)	Jamming (0/1)	Notes
6	10	7	5.8				
6	10	2	4.2		1		
7	10	11	4				
7	10	2	2.5				
7	10	7	5.2	1			
7	10	1	4.8				
7	10	8	5.1				
7	10	9	5.1				
7	10	5	5				
7	10	3	5.1				
7	10	10	5.2				
7	10	4	5				
7	10	6	5.1				
8	10	11	5.7	1			
8	10	2	5.2				
8	10	5	5.4				
8	10	9	5.8				
8	10	1	5.2				
8	10	3	5.2				
8	10	10	5.3				
8	10	6	5.3				
8	10	7	5.1				
8	10	8	5.1				
8	10	4	5.2				
9	10	9	5.8				
9	10	5	5.9				
9	10	11	3.5				
9	10	10	5.7				
9	10	6	5.3				
9	10	1	5.2				
9	10	2	4				
9	10	3	5.1				
9	10	4	5.9				

Continued on next page

Table 18 – continued from previous page

Test No.	**Tilt-Angle**	**Shuttlecock No.**	**Distance (m)**	**Double Shots (0/1)**	**No Shots (0/1)**	**Jamming (0/1)**	**Notes**
9	10	8	5.8				
9	10	7	5.6		1		
10	10	7	5				
10	10	10	5.1				
10	10	5	5.5				
10	10	1	5.8				
10	10	6	5.5				
10	10	4	4.9				
10	10	3	5.1				
10	10	2	4.8				
10	10	11	5.7				
10	10	9	5.9				
10	10	8	5.5		1		
							End of table

C.3 Results of the SCL Testing with 20-degree Tilt

Table 19: Results of the SCL testing with a 20-degree tilt.

Test No.	Tilt-Angle	Shuttlecock No.	Distance (m)	Double Shots (0/1)	No Shots (0/1)	Jamming (0/1)	Notes
1	0	1	5.1				
1	0	2	4.9				
1	0	3	5.5				
1	0	4	5.5				
1	0	5	5.1				
1	0	6	5.8				
1	0	7	5.5				
1	0	8	5.9				
1	0	9	5.7				
1	0	10	4.9				
1	0	11	5.5				
2	0	1	5				
2	0	2	4.4	1			
2	0	3	5.8				
2	0	4	5.1				
2	0	5	3.8				
2	0	6	5.6				
2	0	7	5.8				
2	0	8	5.1				
2	0	9	5.1				
2	0	10	5.6				
2	0	11	5				
3	0	10	5.2				
3	0	6	5.3				
3	0	2	5.3				
3	0	5	5.7				
3	0	7	5.7				
3	0	3	5.1				
3	0	4	5.2				
3	0	9	5.1				
3	0	1	4.8				

Continued on next page

Table 19 – continued from previous page

Test No.	Tilt-Angle	Shuttlecock No.	Distance (m)	Double Shots (0/1)	No Shots (0/1)	Jamming (0/1)	Notes
3	0	8	6				
3	0	11	4.8				
4	0	2	4.5				hit the roof
4	0	6	5.1				
4	0	4	5.3				
4	0	10	5.1				
4	0	9	5.6				
4	0	3	5.6				
4	0	1	6				
4	0	7	5.2				
4	0	11	6.1				
4	0	5	3		1		
4	0	8	5.6				
5	0	3	5.2				
5	0	9	5.2				
5	0	8	5				
5	0	7	5.5				
5	0	4	5.8				
5	0	6	5.5				
5	0	11	5.5				
5	0	1	5.8				
5	0	10	4.5				
5	0	5	4.9		1		
5	0	2	4.9				
6	0	10	5.4				
6	0	8	5.8				
6	0	6	6				
6	0	4	5.4				
6	0	1	6.1				
6	0	7	5				
6	0	11	5.8				
6	0	5	4.5				
6	0	2	5				

Continued on next page

Table 19 – continued from previous page

Test No.	Tilt-Angle	Shuttlecock No.	Distance (m)	Double Shots (0/1)	No Shots (0/1)	Jamming (0/1)	Notes
6	0	3	5.4				
6	0	9	5.4		1		
7	0	8	5.3	1			
7	0	1	5.9				
7	0	6	5.6				
7	0	5	5.5				
7	0	2	4.5				
7	0	11	4.8		1		
7	0	3	6				
7	0	7	5.9				
7	0	10	5.3				
7	0	4	5.5				
7	0	9	5.4				
8	0	7	5.9	1	1	1	
8	0	3	5.9				
8	0	1	5.2				
8	0	6	5.2				
8	0	4	6				
8	0	9	6.1				
8	0	5	0	1	1		
8	0	8	6				
8	0	10	5				
8	0	11	5.8				
8	0	2	6				
9	0	6	5.5		1		
9	0	1	5.6				
9	0	5	5.1				
9	0	10	5.1				
9	0	2	4.9				
9	0	8	5.8				
9	0	11	5.3				
9	0	9	5.6				
9	0	3	4				

Continued on next page

Table 19 – continued from previous page

Test No.	**Tilt-Angle**	**Shuttlecock No.**	**Distance (m)**	**Double Shots (0/1)**	**No Shots (0/1)**	**Jamming (0/1)**	**Notes**
9	0	7	5.2		1		
9	0	4	4.5			1	
10	0	7	5.2		1		Extension tube got loose
10	0	11	5.2				
10	0	6	5.8		1		
10	0	9	5.3				1
10	0	1	5				
10	0	4	4.7				
10	0	2	5.1				
10	0	3	5.8		1		
10	0	10	3.3				
10	0	5	3.8				
10	0	8	5.9				

End of table

C.4 Results of the SCL Testing with 30-degree Tilt

Table 20: Results of the SCL testing with a 30-degree tilt.

Test No.	**Tilt-Angle**	**Shuttlecock No.**	**Distance (m)**	**Double Shots (0/1)**	**No Shots (0/1)**	**Jamming (0/1)**	**Notes**
1	0	1			1		All hit the roof
1	0	2					
1	0	3			1		
1	0	4					
1	0	5					
1	0	6					
1	0	7					
1	0	8					
1	0	9					
1	0	10					
1	0	11					
2	0	6			1		
2	0	10			1		
2	0	11			1		
2	0	2					
2	0	7			1		
2	0	3			1		
2	0	5			1		
2	0	9					
2	0	4			1		
2	0	1			1		
2	0	8					
							End of table

NTNU

Norwegian University of
Science and Technology