Aalto University
ELEC-D0301 Protopaja
Year 2024

Final report

Portable Target Acquisition System

Saab

Date: 01.09.2024

Topias Malinen
Risto-Pekka Siponen
Ilia Lebedev
Mikko Vanhanen
Elmeri Huikko

Information page

Students
Topias Malinen
Risto-Pekka Siponen
Ilia Lebedev
Mikko Vanhanen
Elmeri Huikko

Project manager
Topias Malinen

Sponsoring Company
Saab Finland Oy

Starting date
3.6.2024

Submitted date
01.09.2024

Abstract

The goal of this project was to research and develop a prototype of a portable target acquisition system. The requirements for the prototype given by Saab are as follows: the user should be able to use the device to find and designate an arbitrary target in clear weather, and accurately measure the distance between the system and said target (minimum 50 meters). After this, the device should calculate the GPS coordinates of the designated target and display the calculated coordinates for the user. The device should be drone-mountable and therefore weigh no more than one kilogram.

The prototype developed by the end of the project consists of two parts: a drone-mountable air unit and a portable ground control station. The air unit, which features a stabilized 2-axis gimbal, continuously transmits real-time video footage that is then received and displayed by the ground unit. The ground station is used for monitoring and controlling the air unit, that is capable of accurately measuring the distance between the device and the desired target upwards of 600 meters, with a theoretical limit at 1000 meters using the current laser rangefinder module. The system can then use the measured distance, the system's own position and the compass bearing of the target relative to the device to calculate the GPS coordinates of the designated target. The results are then transmitted and displayed on the user interface of the ground unit.

While we had managed to develop a prototype that fulfilled the predefined requirements given by Saab, there are a few areas where the prototype does not consistently meet the specified goals. Specifically, the target coordinates calculated by the system are not consistent as the calculated compass bearing of the target is inaccurate. This is likely due to the location and quality of the air unit's magnetometer. The majority of the issues encountered during the project can be attributed to constraints in the development timeline.

Overall, the final prototype demonstrates the feasibility of the proposed design and serves as a solid foundation for potential future iterations.

Tiivistelmä

Projektin tavoitteena oli tutkia ja kehittää kannettavan maalinhakujärjestelmän prototyyppiä. Saab antoi prototyypille seuraavat vaatimukset: Käyttäjän tulee pystyä käyttämään laitetta löytämään mielivaltainen maali hyvällä säällä sekä mittaamaan tarkka etäisyys järjestelmän ja kyseisen maalin välillä (vähintään 50 metriä). Maalin etäisyyden mittaamisen jälkeen laitteen tulee laskea mitatun maalin tarkat GPS-koordinaatit ja välittää ne käyttäjälle. Prototyypin täytyy olla droneen kiinnitettävissä, joten laitteen tulee painaa enintään yhden kilogramman.

Lopullinen prototyyppi koostuu kahdesta osasta: droneen kiinnitettävästä ilmayksiköstä ja kannettavasta maayksiköstä. Ilmayksikkö, jossa on vakautettu 2-akselinen gimbaali, lähettää jatkuvasti reaaliaikaista videokuvaa, jonka maayksikkö vastaanottaa ja näyttää käyttöliittymässään. Maayksikköä käytetään ilmayksikön valvontaan ja ohjaamiseen. Ilmayksikkö kykenee mittaamaan tarkan etäisyyden haluttuun maaliin jopa yli 600 metrin etäisyyksillä. Teoreettinen raja etäisyyden mittaukselle on 1000 metriä nykyisellä laseretäisyysmittarilla. Ilmayksikkö käyttää mitattua etäisyyttä, järjestelmän omaa sijaintia sekä maalin suhteellista suuntimaa laitteeseen nähden laskeakseen valitun maalin GPS-koordinaatit. Lasketut koordinaatit lähetetään maayksikölle ja näytetään maayksikön käyttöliittymässä.

Vaikka onnistuimme kehittämään prototyypin, joka täyttää Saabin ennalta määritellyt vaatimukset, on kuitenkin joitakin alueita, joissa lopullinen prototyyppi ei johdonmukaisesti täytä määritettyjä tavoitteita. Erityisesti järjestelmän laskemissa maalin koordinaateissa voi olla paljon vaihtelua, sillä järjestelmän laskema maalin suuntima on epätarkka. Tämä johtuu todennäköisesti ilmayksikön magnetometrin sijainnista sekä laadusta. Valtaosa kohdatuista ongelmista voidaan katsoa johtuvan projektin rajallisesta aikataulusta.

Lopullinen prototyyppi osoittaa ehdotetun järjestelmän toimivuuden ja tarjoaa vankan perustan mahdollisille tuleville iteroinneille.

Table of Contents

1. Introduction

The increased usage of drone technology in modern military conflicts has highlighted the need and potential for advanced drone-mounted technology. Modern scenarios, from conflict zones to disaster response, require precise data and real-time intelligence to ensure effective coordination in various operations.

This project aims to address some of these needs by creating a portable drone-mounted device that can accurately measure the distance to a desired target and subsequently provide the user with the precise GPS coordinates of said target.

We expect that the primary application of this device would be in military contexts, such as target acquisition and designation, reconnaissance and surveillance operations. Additional use cases beyond military purposes include search and rescue operations, land surveying, environmental and geographical studies.

2. Objective

The project's objective is to be a working prototype of a drone-mountable camera device that can find the target's location seen through the camera. The target is located using the device's location, the distance between the device and the target and direction of the camera. The device will be wirelessly communicating with the ground station, in this case a commercial laptop, through which a human user can control the camera.

A GPS module is used to find the location of the device. The distance between the device and the target is found with a laser distance finder attached to the camera. The direction of the camera and the laser finder is found with the combination of a compass, which gives the horizontal angle, and the tilt sensor of the laser rangefinder, which gives the vertical angle.

The location of the target is shown to the user. The user does not need to have experience with drone flying or with the system itself to use it properly. The system should work in clear weather up to 600-meter distances.

3. Hardware Development

3.1. Component Selection

Due to our limited time and budget, the component selection for this project was largely determined by the price and availability of the components. The notable key components used in the final prototype are listed below:

- **2S LiPo Battery (+7.4V 2200 mAh)**: We chose this battery as the main power supply for our system due to its large capacity and ease of use. The +7.4v output allows us to connect this battery directly to the motor controller board. With this chosen capacity the battery life is guaranteed to outlast the drone which the system will be attached to.

- **SparkFun BabyBuck Regulator Breakout (+5V output)**: This buck converter is used to convert the voltage from the battery into +5v that the Compute Module 4 and the main board require. We chose this module as we needed the regular to be as small as possible and cheap while still being able to handle large currents (up to 3A).

- **Raspberry Pi Compute Module 4**: The “brains” of the device. Opting to use a Raspberry Pi for our device was an easy choice due to its extensive community support, ease of use and relatively low price. And it provides more than enough computing power for our application, while being smaller than a credit card. For our purposes we are using the version with 4 GB of RAM and 16 GB of eMMC storage.

- **Arducam 12MP IMX708 Camera (B0308)**: We chose this camera module as the IMX708 sensor with a fixed-focus lens is recommended for use by the OpenHD community. It also has a low price and is easy to use with the Raspberry Pi.

- **MAX-M10S GNSS Module**: For finding the location of the system. U-blox was chosen due to its reputability. The MAX-M10S was chosen due to its high sensitivity and low power consumption. It has an inbuilt LNA and SAW filter, which makes passive antennas usablee.

- **ICM-20948 9-axis Inertial Measurement Unit**: This IMU is used for calculating the compass heading of the air unit. We chose this chip because of its low price and extensive library support.

- **BMP388 Digital Pressure Sensor**: This pressure sensor is used for determining the altitude of the device. As this information was not crucial for the functionality of our prototype, we chose this sensor due to its very low price.

- **Laser Rangefinder Module (LRFX00M3LS)**: Finding a rangefinder module that was suitable for our budget and purposes was difficult. This module was chosen due to its relatively low price, good long-range capability and ease of integration. The module also comes with an integrated tilt sensor which allows the module to calculate the horizontal distance to the measured target, which makes the calculation of target coordinates easier. The module can measure up to 1000 meters.

- **ALFA AWUS036ACH v.2 Wi-Fi Adapter**: This Wi-Fi adapter was chosen as our radio transmitter and receiver after the decision to go with the Wi-Fi based OpenHD communication link. The adapter was in the list of recommendation by the developers and compared to other alternatives had high potential range due to two external antennas, potential high TX power and additionally was comparatively well available.

- **iPower GM3506 Gimbal Motor w/Encoder**: Two of these BLDC motors are used to drive the gimbal, specifically in pan and tilt directions. These motors have been chosen thanks to their high power required for the weight of our system.

- **Storm32 BGC 32Bit 3-Axis Brushless Gimbal Controller**: The controller board for the above-mentioned motors provides their accurate positioning based on the motor encoder values as well as on board IMU for gimbal’s stabilization. Specifically, this controller was chosen due to its high market availability and low price relative to similar alternatives.

3.2. Circuit Design

For this prototype two PCBs were designed and ordered: the main board which carries the Raspberry Pi Compute Module 4 and is responsible for communicating with all the peripherals and sending the video and telemetry data to the ground unit, as well as a custom board for our GNSS module. The schematics and PCBs for this project were designed using KiCad 8.0.

The Compute Module is attached to the main board using two 100-pin Hirose connectors. Due to the large number of pins and their purposes, the connectors have been placed on separate sheets, which you can see in the schematics below.

The first schematic (Figure 1) shows the camera connector and the two USB ports, connected using a USB multiplexer, as the Compute Module 4 only comes with one USB 2.0 port by default. The USB-C connector can be used for programming and powering the CM4, while the second USB port is reserved for the Wi-Fi adapter. The power, ground and data wires of the adapter are to be soldered directly to the board. This is done to create a more secure connection as the vibrations and movement while the system is in the air could result in a poor connection when using a standard connector.

The user can select between the two USB ports by manipulating the dip switch. By default, the dip switch is placed in the *off* position, meaning that the Compute Module 4 acts as a USB host, and the system accepts USB slave device connections using the primary USB port, i.e. the plated through holes reserved for the Wi-Fi adapter. On the contrary, if the dip switch is placed in the *on* position, and a power cable is plugged into the USB-C connector, the Raspberry Pi will in turn act as a USB slave. This allows for the user to flash the operating system of the CM4, similarly as is done on the CM4 IO board. The board also includes the option to pull the nRPIBOOT pin to ground, which is a requirement for flashing the onboard eMMC memory.

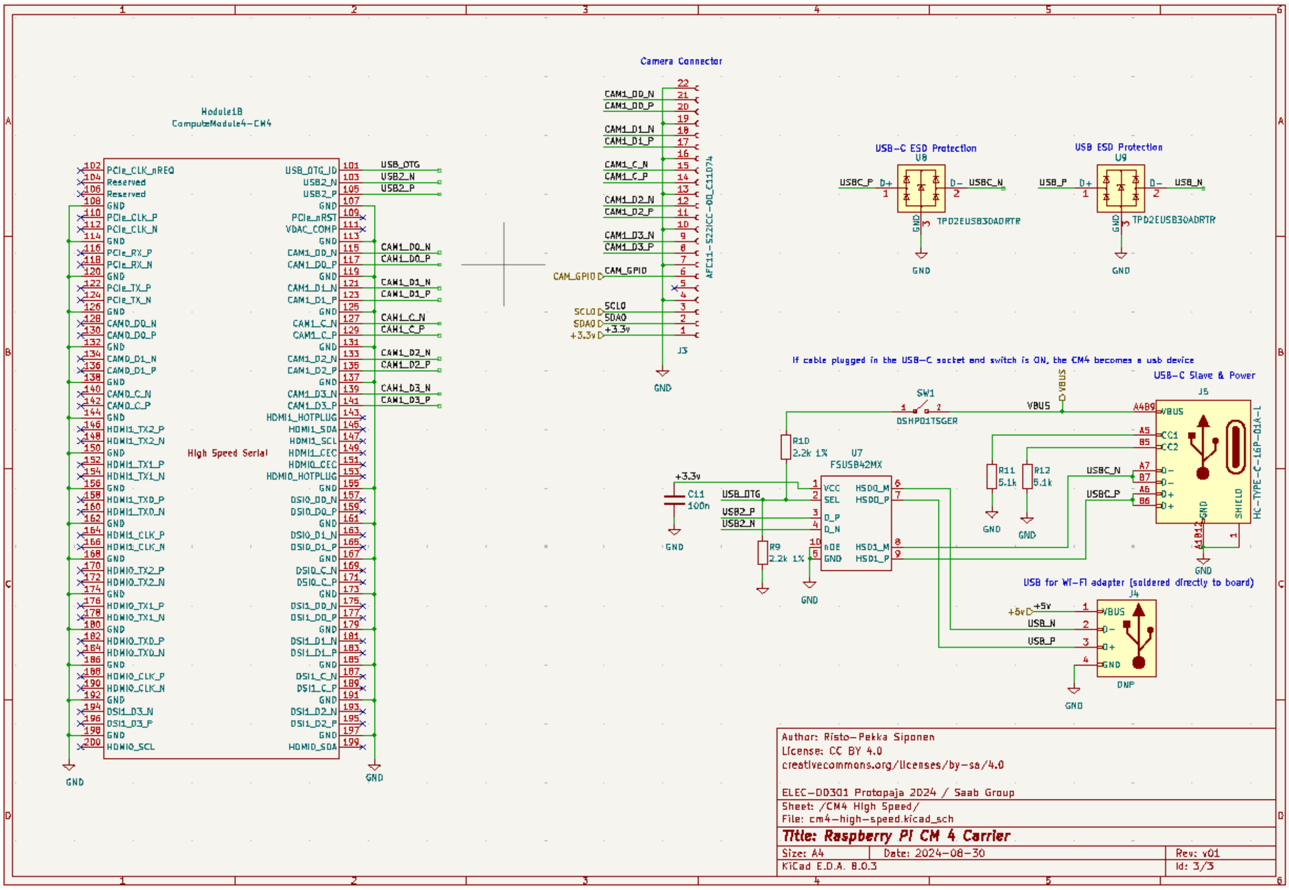


Figure 1. Main board schematic (USB, Camera)

The second schematic (Figure 2) shows the usage of the GPIO pins of the device, such as the I2C connections for the IMU and pressure sensor. As the IMU uses +1.8V logic voltage, and the CM4 is configured for +3.3V logic, level shifting was required for the use of this sensor. There are three JST-SH connectors used for communicating with the laser module, GPS module and the motor controller board. All the aforementioned modules communicate via UART protocol. Auxiliary GPIO pins are left exposed to facilitate potential future upgrades, such as an improved IMU or other sensors. These pins support SPI, UART and I2C.

As the board has two +5v inputs, an OR-switch is used to connect the two power rails. The primary power input wires, which come from the external switching regulator, are to be soldered directly to the board. The previously mentioned USB-C connector functions as a secondary power input, which is mainly intended for use during the prototype's development cycle.

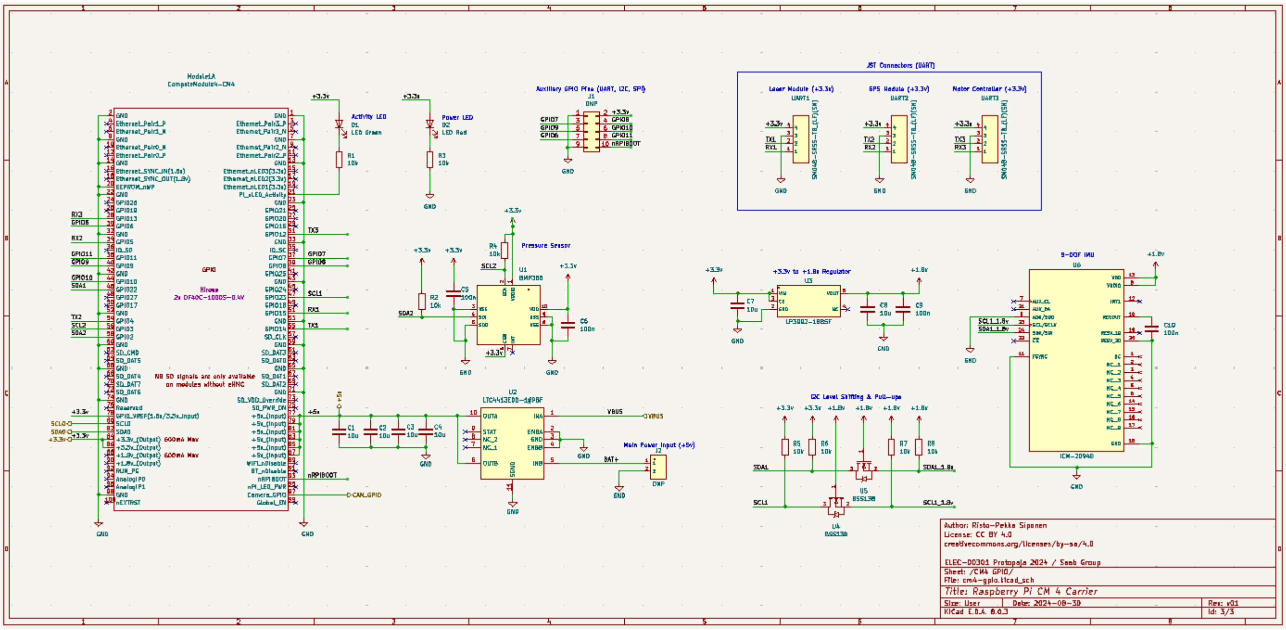


Figure 2. Main board schematic (GPIO, Sensors)

The third schematic (Figure 3) is of the GNSS Module which uses the u-blox MAX-M10S GNSS receiver as its main component. The MAX-M10S can be used with both active and passive antennas due to the inbuilt LNA filter.

The module can be improved with firmware configurations, as it is on the default configuration as of now, to utilize the 2-pin design, which is based off of the MAX-M10S integration manual. The design allows for the detection of short circuits in the antenna RF path and for powering off the antenna supply when using passive antennas.

The module has a backup battery holder suitable for CR927 and CR1025 lithium coin cells that when used the receiver maintains time information and navigation data to speed up the receiver start up after a loss of power.

The module has 3 main connectors; The SMA connector for the antenna, UART connection for communicating with the air unit’s CM4 and 4 open sockets for the pins timepulse, reset, safeboot and ground, in case they are needed for configuration or error searching. The I2C pins of the module are left open and unused.

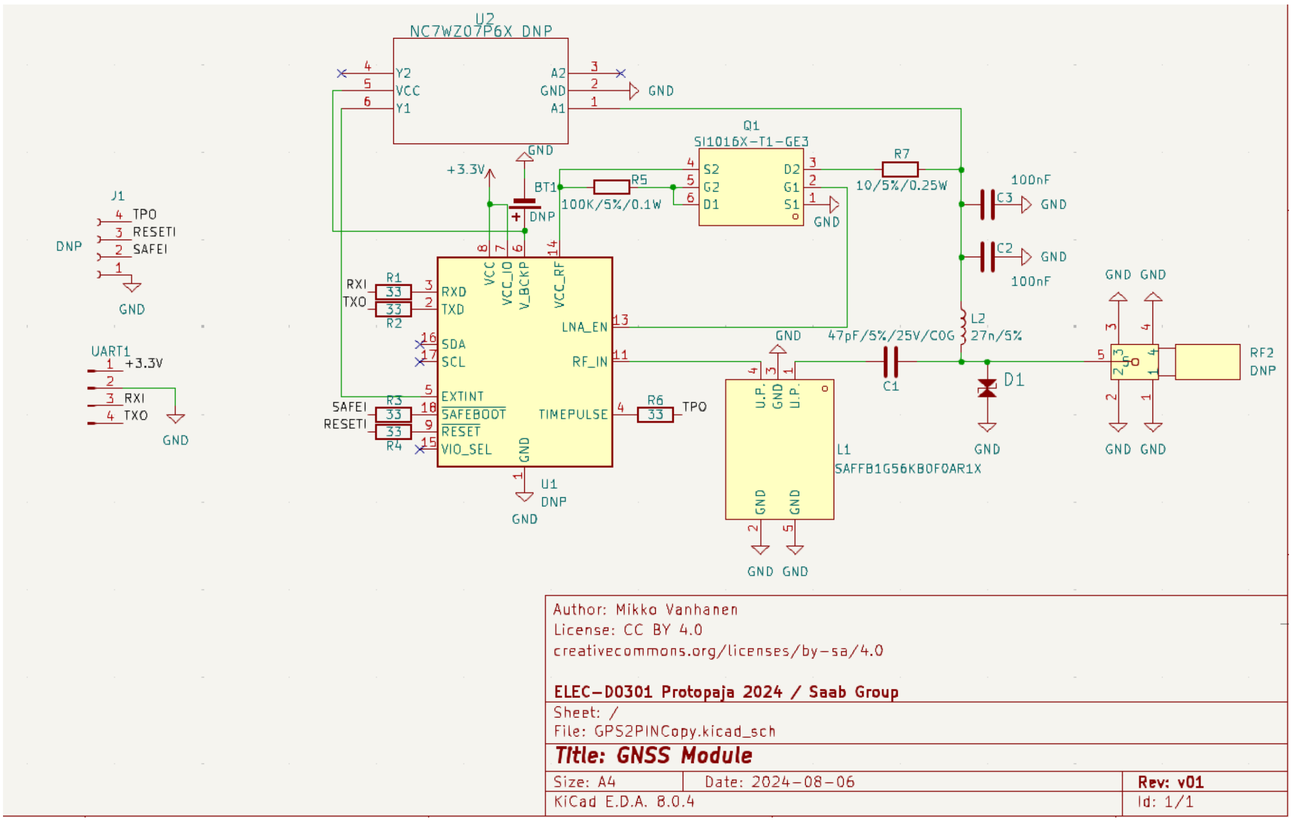


Figure 3. GNSS Schematic

3.3. PCB Layouts

As the main board must be inside the gimbal, we wanted it to be as small as possible. Initially, our plan was to order the main PCB with double-sided assembly, where the CM4 would rest on one side of the board, and all the connectors and other components would be on the other side. This would have meant that the main board could be as small as the Compute Module itself (55mm x 40mm). However, while estimating the PCB's assembly costs, we found that the double-sided assembly would have increased the costs of the order by nearly 100 euros (~200€ total).

After careful consideration, single-sided assembly was chosen to cut costs. This meant that the PCB had to be made larger to accommodate all the components on one side of the board. Fortunately, this didn’t turn out to be too big of a problem, as the size of the gimbal enclosure could simply be increased appropriately.

Furthermore, a more modular design was chosen for this prototype, where some components were deliberately placed externally off the board, e.g. the switching regulator and GPS module. This was done to reduce the likelihood of potential production issues and to mitigate their effects on our progress, there was very limited time and budget to work with. This meant that there were less components overall on the main board, which made placing everything on one side easier.

Initially, some of the smaller components had been placed under the Compute Module to save space, but unfortunately the taller Hirose connectors (1,5 mm clearance) for the CM4 were out of stock on JLCPCB, which forced us to use the shorter connectors (0 mm clearance). Thus, the board had to be further expanded to fit all the components. The final board dimensions are 60mm x 62,5 mm and it has 4 x M2.5 mounting holes, same size and positions as the mounting holes on the Compute Module.

The longest USB data lines (from the multiplexer to the compute module) have been routed as a 90 Ω (+-15%) differential pair, which means that with our 2-layer board the traces are noticeably wider. Although, given that USB 2.0 is notoriously forgiving at short distances, this was likely redundant.

The final iteration of the PCB layout can be seen in the figure below (Figure 4). The top of the PCB includes a wireless keepout zone surrounded by via shielding. This was added for the usage of the onboard Wi-Fi chip of the Compute Module, which can be used to communicate with the CM4 using SSH. This is used purely for debugging purposes during development, not intended for the final product.

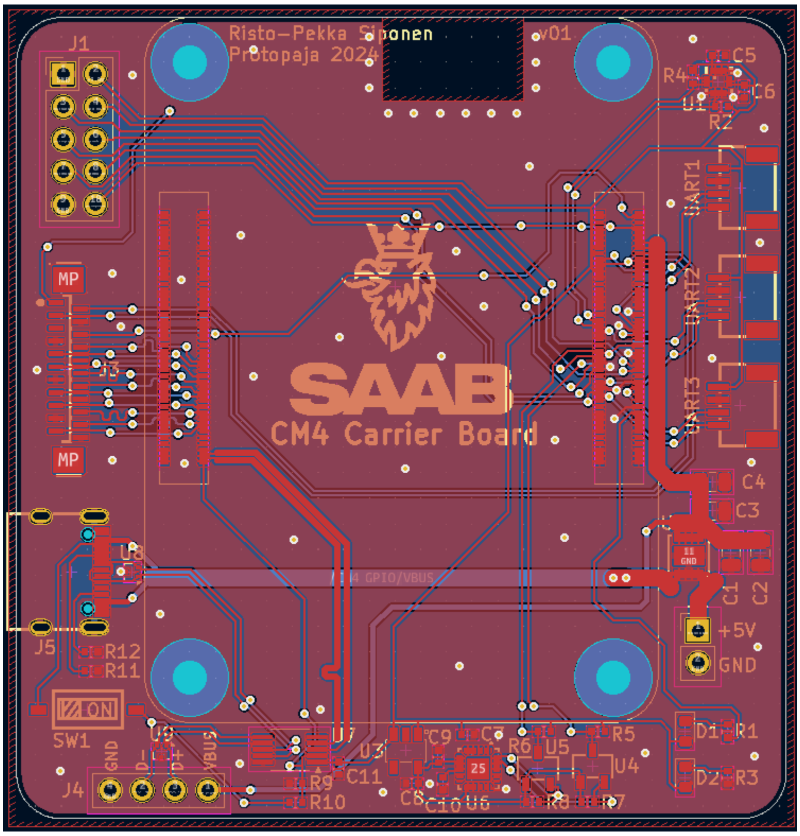


Figure 4. Main board PCB layout

The final board received from JLCPCB can be seen below (Figure 5). Notice the missing ESD protection for the USB data lines (U8 & U9). This was seemingly caused by human error during the ordering process. Fortunately, this component is not critical for the performance of the board.

Out of the two assembled boards received from JLCPCB, only one was usable straight out of the box. The other board was unusable as one of the 100-pin connectors for the CM4 was not placed perfectly straight on the board. This meant that the connectors of the Compute Module were not aligned with the connectors on the board. This could most likely be fixed with minimal effort in-house, but as the other board functioned without issues, this was no cause for concern. It is unclear why the connector was skewed, but a probable cause is the fact that we chose the cheaper economic assembly from JLCPCB, rather than the more expensive standard assembly, which would likely yield in higher quality assembly.

As an improvement for a potential future iteration of the main board, more/bigger capacitors should be considered for the main +5V rail (output from the OR-switch). Currently when plugging in a high-power USB device (such as a Wi-Fi transmitter) to the primary USB port, the large inrush current causes a sudden voltage to drop on the input rail, which results in the CM4 rebooting itself. However, this is not a critical issue, as connecting the USB device before booting up the CM4 results in normal behavior. Furthermore, the Wi-Fi transmitter cable will be soldered directly to the board and remain connected, meaning that this issue is only noticeable during the development process. Alternative solution for this issue could be to replace the multiplexer with a powered USB hub, although this could result in more complexity.

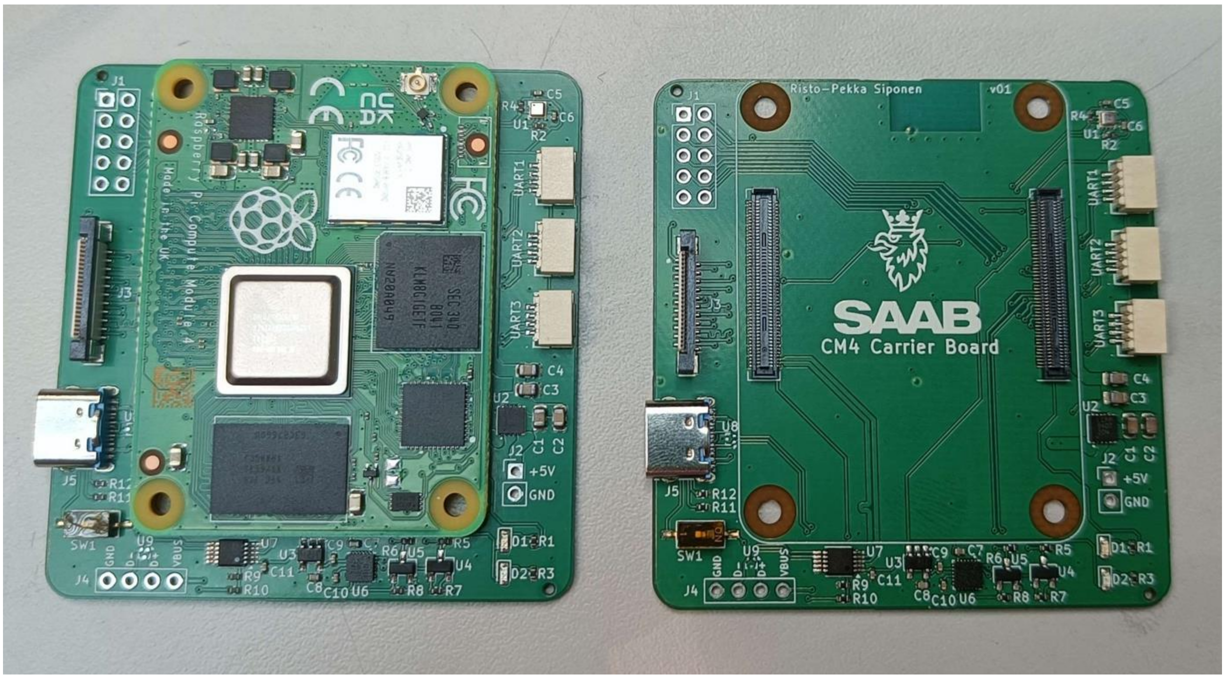


Figure 5. Finished main PCB (left – with CM4 attached, right – without CM4)

The GNSS PCB is mainly designed around the RF path, leading from the SMA connector to the RF-IN pin of the receiver. For proper impedance matching (50Ω) along the RF path it was crucial to get a specific trace width and stackup, which was chosen with JLCPCB’s controlled impedance calculator. Extensive use of ground vias were needed along the RF path to ensure no interference affecting the RF signal.

The final dimensions ended up being 40mm x 30mm x 0,8mm. For mounting, the same 2,5mm holes as in the main board were used. All components except the GNSS receiver, the battery holder and SMA connector, which were soldered by hand, were assembled by JLCPCB.

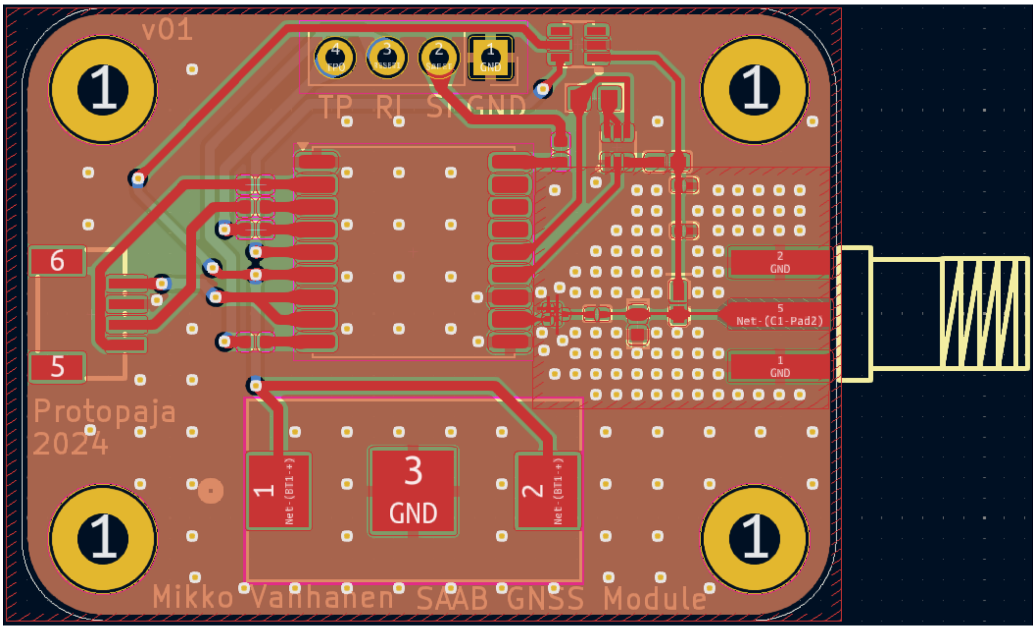


Figure 6. GNSS Module layout

The board could be improved by e.g. adding an LED to the timepulse path to detect if the receiver is getting data without needing to solder anything to the timepulse socket. A larger battery holder could be used due to the CR927 and CR1025, which the current holder can be used with, being hard to find. A rechargeable battery that wouldn’t need replacing could also be used, but those are also hard to find. Using lower quality materials in assembly and normal assembly speed would make the PCB noticeably cheaper.

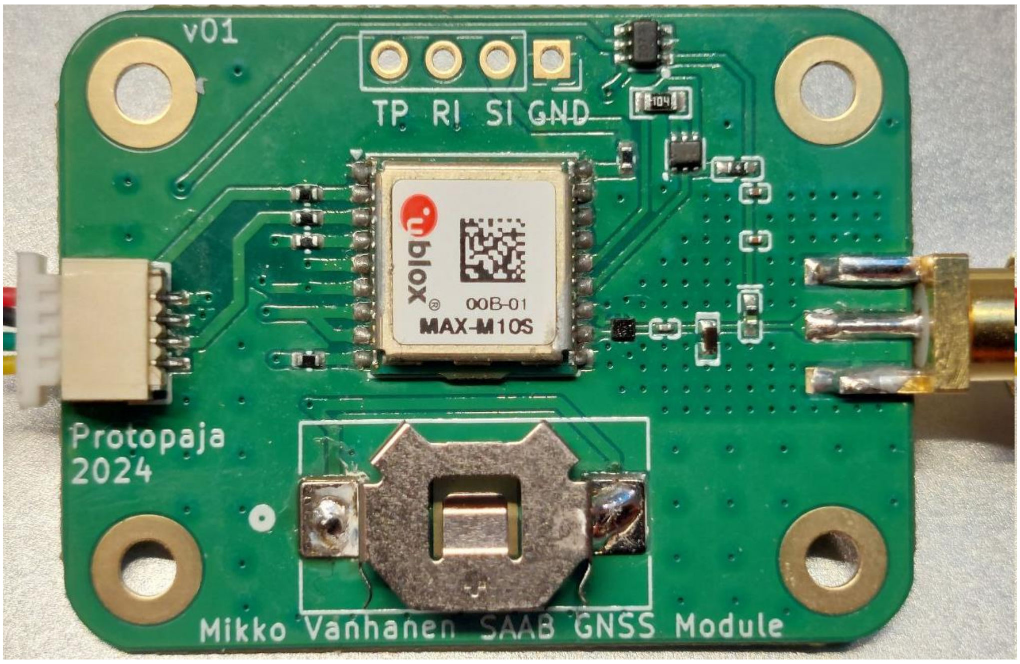


Figure 7. Finished GNSS module

4. Mechanical solutions and enclosure

4.1. Worm drive solution

One of the project’s solutions to control our optics (Camera & Rangefinder) was to use a mechanical solution called worm drive. Worm drive have two main parts which include worm screw and worm gear (Figure 8) and it differs from traditional cogwheel by not having straight teeth. The idea behind this solution is to attach motor’s axle directly to worm screw which then rotates worm gear which is attached to our ball part of the prototype.

We started doing research firstly by 3D modelling and printing our own worm drive mechanism from 3D printed PLA (Figure 8). The advantage of making your own worm drive mechanism is that you can customize the gear ratio very quickly and test them with different kinds of motors. Adjusting gear ratio and motor’s angular speed you can customize whole system’s torque and speed which in our case needs to be well balanced because of the unstable drone and need of high precision for long targeting ranges.

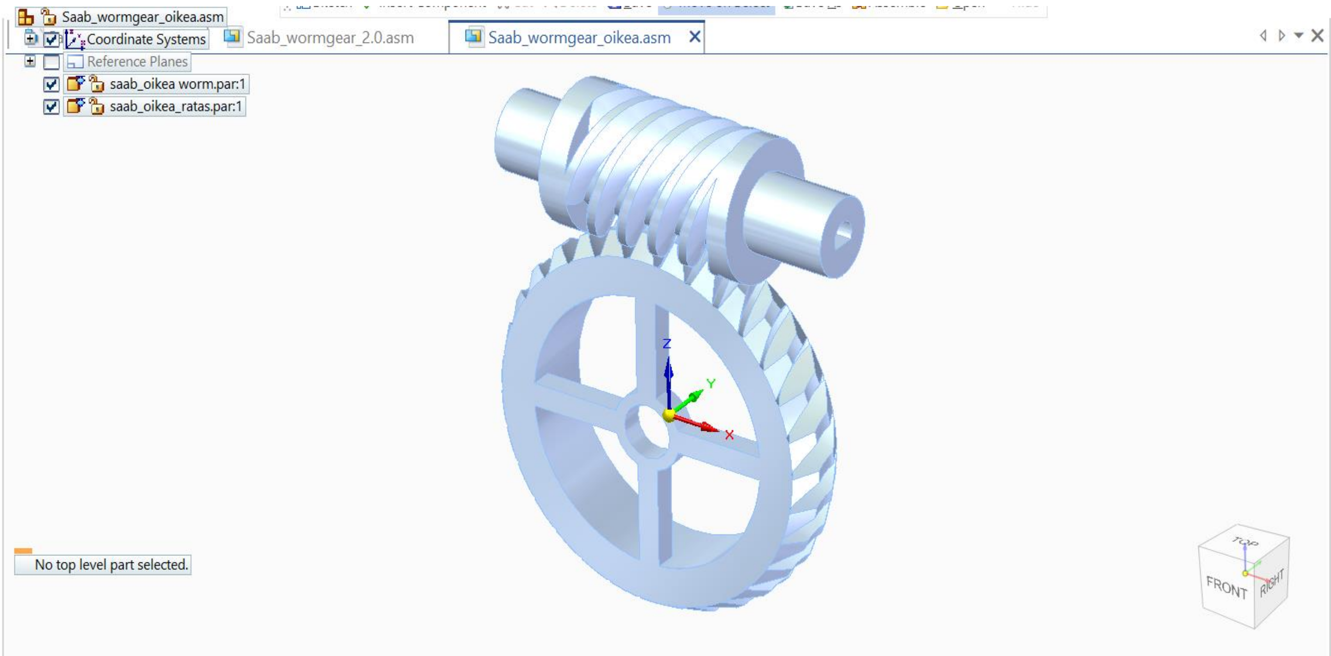


Figure 8. 3D design of the worm drive

Our first test included (Figure 9):

- 1x Pimoroni metal geared micro motor
- 1x Worm made of 3D printed PLA
- 1x Gear made of 3D printed PLA
- 1x Small dot laser module
- Bolts and nuts
- 1x 5M bolt screw
- 3x bearings
- 1x L293DNE motor driver
- Wires
- 1x Arduino Uno
- 3D printed holder for metal geared micro motor

With this test setup and 5 runs, we got precision of 3.4 mm while distance between target and this setup was 1 meter (Figure 9). **This precision at 600 meters would equal approximately 2 meters of error** (Between steps the motor took) **this equals a 0.19 degrees per so called step**. This was a short test so there is no absolute guarantee of how reliable this solution itself would be. These results can be considered accurate if we consider: the time we built this test setup, we didn't use any lubricant between the worm gear contact area, and the materials we used are far from the best. Layered PLA is not the solution to use as a material for worm drive and there are available much more suitable materials, such as:

- Nylon
- Aluminum alloys
- Delrin (POM-H)
- Different kinds of composites with high mechanical resistance
- Bronze with lightweight alloys as option

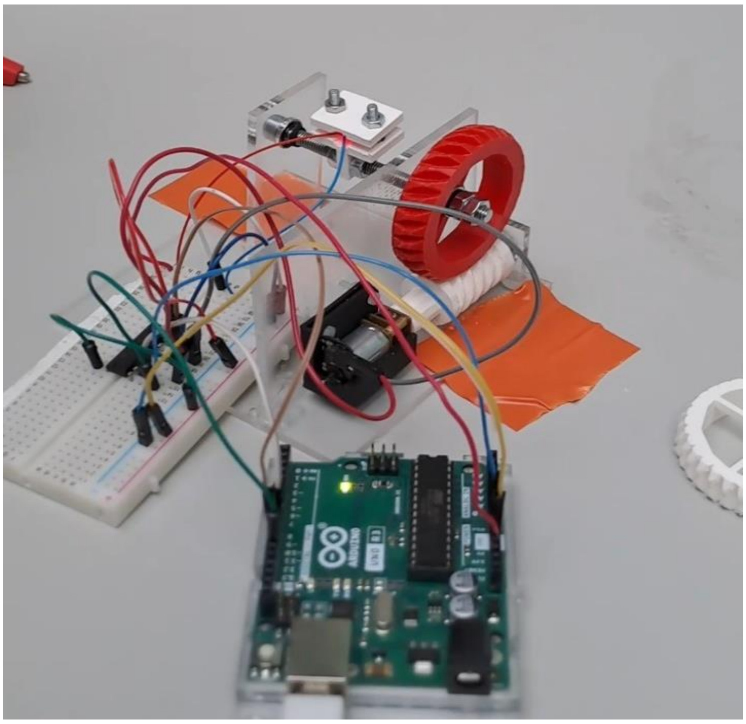


Figure 9. First test setup

After we tested our own worm gear mechanism, we decided to move on to buy readymade solution. This included few Pimoroni micro metal gear motors (Right Angle Micro Metal Gearmotor (pimoroni.com)) and motor driver breakout board (TMC7300-BOB Analog Devices Inc./Maxim Integrated | Kehitysalustat, sarjat, ohjelmointilaitteet | DigiKey). This worm drive solution would have been better because it required less space and weighed less than our own. With another test setup (Figure 10) we were able to pull off even more precise results where **in 600m error was approximately 1.6 meters (0.15 degrees)**. Unfortunately, time was becoming a bit of an issue and we needed to start integrating our prototype. Our enclosure was made only for BLDC motors, so we didn't have time to continue with both system's control solutions. Because of this the worm drive solution was never tested with the actual prototype.

Second test setup had:

- 2x Pimoroni Right Angle Micro Metal Gearmotors
- 3D printed casings
- Arduino Uno

- L293DNE motor driver
- Joystick

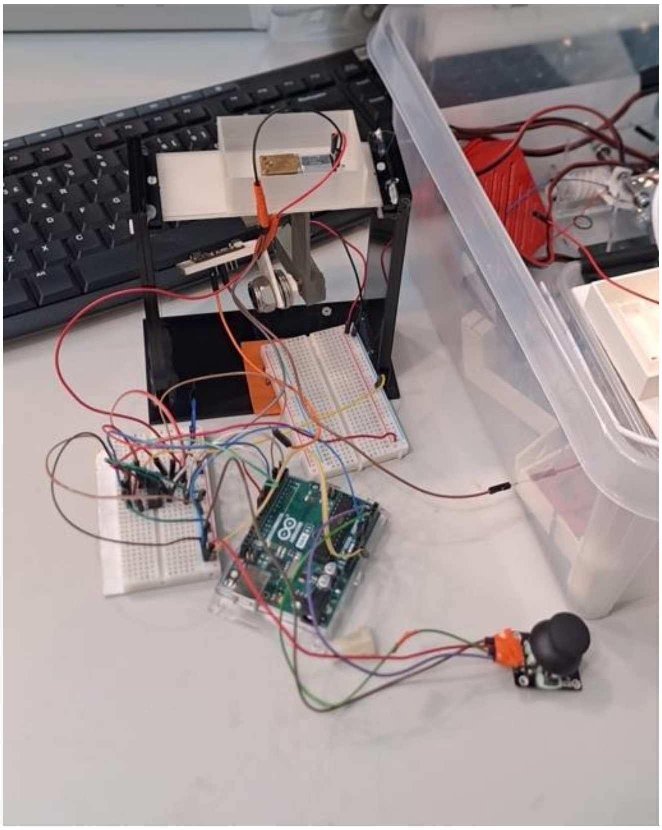


Figure 10. Second test setup

What we learnt about worm drive mechanism?

- **You should use different materials between gear and worm**. Prefer greater surface durability strength material for worm and material with less durability for gear. (Making sense out of gear materials | Gear Solutions Magazine Your Resource to the Gear Industry)
- **Balancing precision and reactiveness**. Consider the exact use case for the prototype (What is the exact thing you want?). For longer ranges with slower targeting you should favor very high precision rather than reactivity of the motors. On the other hand, if you want to do very fast targeting for shorter ranges then you should favor faster and more reactive motor/mechanical options.
- **Works better for long range**. Worm drive solutions are most suitable for slower long-range targeting because of how easy it is to get great precision with them.
- **Backlash**: Worm drives usually have some sort of backlash in them which reduces the actual precision of them. However, if worm gear parts are done by professionals from suitable materials the backlash doesn't significantly reduce the precision of the result. According to following scientific article the normal backlash value of high precision drives is usually under 15 micrometers. (Materials | Free Full-Text | Worm Gear Drives with Improved Kinematic Accuracy (mdpi.com))

Strengths and weaknesses for worm drive

Strengths

- **Weight**: Reduces prototype's weight by 118 grams (**Reduces total weight approximately 15%!**), if you are using the Pimoroni motors which were mentioned in the second test setup.
- **Precision**: We were able to get very precise results using worm drive in a very short period of time (approx. 2 weeks). Precision of the worm drive with minimal effort is also known very well in scientific research groups around the world (Worm Gear Drives with Improved Kinematic Accuracy (researchgate.net)). With more suitable worm gear materials, use of lubricant between contact points, and greater gear ratio the final precision could be same as BLDC motors or even better.
- **Power consumption**: When holding the optics' position, worm drive solution consumes very nearly 0 milliamps. Worm drive only consumes power when moving the optics.
- **High torque**: Worm drives are known for their ability to provide high torque, which is beneficial for systems that require precise movement for heavy loads.

Weaknesses

- **Reliability**: Concerns about reliability, how well this solution works with the actual prototype is unknown.
- **Stabilization**: Because of the low speed and reactivity, the stabilization with worm drive solution might not be very good. Not tested.
- **Speed**: Worm drive solution usually have very low speed. Although this can be customized with different motors and gear ratio, accuracy will then suffer.

Summary

With the current statements and research done towards worm drive solution we can very safely suggest that **if you are trying to build prototype that maximizes preciseness, battery life, weight reduction, and operates in rather stable environment which requires very long range targeting you should heavily consider using worm drive mechanism to control optics of the entire system**. On the other hand, if you are trying to get something **between very long-range targeting and reactiveness there might be possibility to integrate external motors for the system which would do stabilization, and you can still use the worm drive solution for the actual targeting**. This kind of arrangement would consume more power and weigh more because it has outer reactive stabilization motors and slower targeting motors (worm drive). But if this solution is done right, it would not end up being extremely heavy or consume much more power than the BLDC motor solution, which we used in the prototype itself. This kind of system which uses two different motors (Inner & Outer) could be mounted on even bigger military drones, which have higher flying altitude or bigger consumer drones like PX4.

4.2. BLDC motor solution

A second solution for driving the gimbal which was in the end implemented in the final prototype. The idea of such a solution is that gimbal is directly attached to the brushless DC motors without any gearbox between them. The type of the BLDC motors used was specifically designed for gimbal applications. To keep track of the rotor's position the motors are equipped with magnetic encoders which send the absolute angle information of the shaft through the SPI protocol (Figure 12). The mechanism of this measurement solution is based on a small permanent magnet attached to the shaft of the motor and the magnetic encoder sensor above it. As the shaft rotates so does the magnetic field of the permanent magnet. These movements are detected by the hall effect sensor integrated into the magnetic encoder chip. The configuration described above is one of the most popular ways of driving gimbals: there are many ready-to-use solutions on the market. One such solution has been purchased by the team: 2 BLDC motors equipped with magnetic encoder sensors (Figure 11). A certain amount of time has been spent on choosing the proper size of the motor: the main concern was that the motors wouldn't handle enough weight, so we had to consider how much torque each motor size could provide. In the end we chose GM3506 version of the motors which was a good balance between its mass and torque provided.

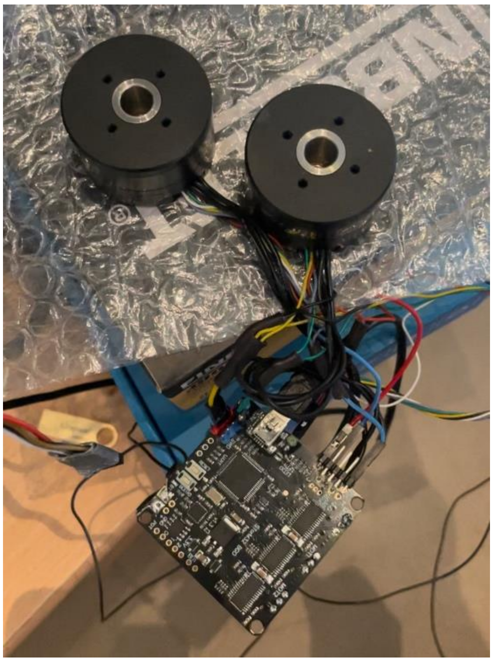


Figure 11. BLDC motors connected to the controller board

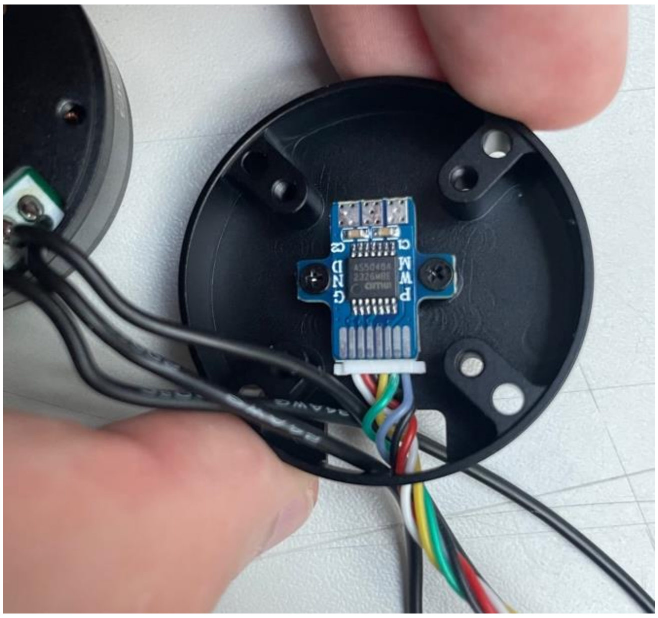


Figure 12. Encoder board mounted into the motor's enclosure.

Strengths and weaknesses of the BLDC solution

Strengths:

- **Accuracy:** high precision is achieved thanks to the accurate magnetic (hall effect) encoders attached to the motors. The maximum declared accuracy is 0.05 degrees.
- **Movements:** smooth movements which prevent image jitter.
- **Response time:** the motors are back drivable since there is no gearbox attached to them, which can be considered as an advantage since it provides a quicker response time. In general, BLDC motors provide a quick response time also due to their structure.
- **Community support:** since BLDC gimbal motors are the most common way of driving gimbals, there is a wide support of them in the community. For instance, our solution is implemented with the help of open-source code and hardware (Storm32) designed specifically for BLDC motors.
- **Robustness:** highly reliable because there are no components undergoing friction while the motor is moving.

Weaknesses:

- **Cost: i**n comparison with the worm gear motors, BLDC motors and especially their specialized gimbal version equipped with the magnetic encoder are more costly. Each motor with encoder module cost 40 EUR.
- **Power consumption:** while static gimbal motor power consumption can be minimized by balancing the system properly so that the forces acting on it cancel each other out, there is still some current flowing into the motors to keep the system stable since there are no gears involved.
- **Size:** electric motors power and size positively correlate and so in order to provide enough torque to stabilize the gimbal a big enough motor is needed. In cases when a geared motor is used this issue could be alleviated by increasing the reduction ratio and so choosing a smaller less powerful motor at the expense of longer response time.
- **Weight:** as with the weakness created by the size of the motors, higher weight presents a problem since generally BLDC gimbal motors feature large windings to provide sufficient torque.

What have we learnt about BLDC gimbal motors?

All in all, we have learnt that BLDC gimbal motors can provide extremely smooth movements which cannot be achieved by either geared or stepper motors. Also, we have learnt that this type of motor consumes power even when the system is static, however this passive power consumption was quite small in comparison to other sinks of the electric power in our system (around 150 mW). One more issue related to the BLDC motors is that they require more sophisticated control electronics in comparison to brushed DC motors.

Summary

Based on the provided information it can be said that BLDC gimbal motors combined with magnetic encoders are a perfect solution for gimbals where additional complexity is not considered as an obstacle: these complexities include a higher weight and size, a need for specific motor driver chips to run the motors, as well as higher cost. On the other hand, these motors are necessary in applications where a smooth jitter free picture from the video feed as well as small response time is an important requirement.

4.3. Other solutions

It is also worth mentioning other solutions considered in the beginning of the project and corresponding test prototypes produced for them: stepper motor (figure 13) and mini servo motor (figure 14) solutions. Both solutions were suspended due to the small torque of the stepper motor chosen, while the mini servo motors had too shaky movements and low angular position accuracy. Also, the team didn't possess enough time resources to research these solutions in more detail.

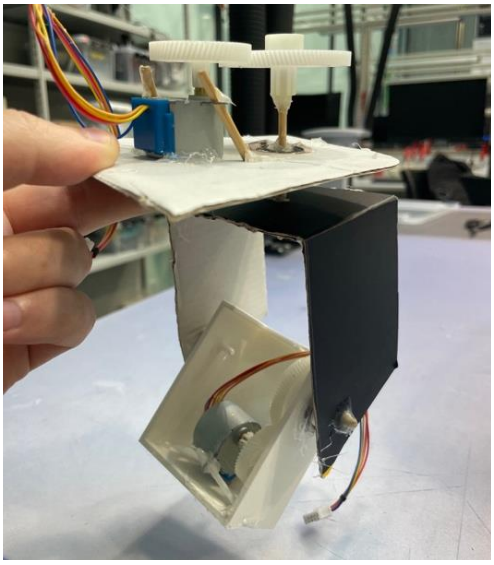


Figure 13. Stepper motor solution

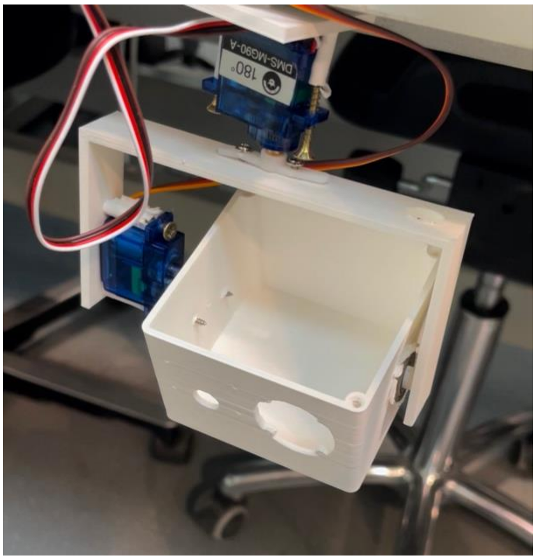


Figure 14. Servo motor solution

4.4. Enclosure

During the planning phase we considered different design options of the gimbal. One of them was the classical design of gimbals which presumes the use of rectangular shaped frame to which motors are attached as well as the object to be stabilized which is typically a camera. Another option we selected was a dome gimbal with a ball-shaped enclosure. The reason for choosing such a configuration was mainly motivated by the improved aerodynamics of a rounded shape compared to a more rectangular one as well the absence of any protruding parts.

Enclosure design we came up with (figure 15) can be divided into several structural parts: static part which goes on top and which gets attached to the drone (or a different mounting platform), gimbal ball where camera, range finder and the compute module are located as well as battery holder which can be slided in and out of the static part. The exploded view of the whole prototype including the enclosure can be seen in figure 16. The enclosure has been FDM 3D printed from PLA, the surface has been sanded and covered with matte grey primer.


Figure 15. The final look of the prototype.

The ball part of the gimbal is composed of two halves in which all the contents are located as well as its holder part which is attached to the motor on one side and is inserted into the bearing on the other side as shown in figure 16. The bearing serves not only as a structural support component but also as a hole for the cables coming from the static part to be traced through. In scenarios where rotating parts containing electronics are present in a product it is a common solution to use sliprings to prevent overtwisting of the wires. However, in our situation this solution wouldn't have worked since some of the cables going into the ball were carrying high speed signals like USB.

The static part is mainly hollow and contains separate volumes. In the front rounded part, there is a space to mount the motor responsible for the yaw movement and cables coming out of the ball and the pitch motor. In the top part, there is a space for installing the Wi-Fi transmitter, gimbal motor controller and the GPS module. The antennas for the transceiver and GPS module are connected in the back. The lid closes the top part and is attached to it with screws. In the static part's back, there is a space for the battery plugged into there. The battery itself is installed into an enclosure which repeats the shape of the hole in the back.

To attach the camera and the range finder without making the assembly process too complicated it was decided to fix them to a separate holder which is then attached to the ball's front part.

There were several issues while designing the ball part of the gimbal. As has been mentioned already there were some size constraints because of the PCB inside it, which made us increase the diameter of the ball. Fitting a relatively large gimbal motor inside was also an issue: in order to avoid any protruding parts, the motor had to be mostly deepened into the ball part which on the other hand created complications in the form more challenging assembly as well as very small clearance to the compute module carrier board. Another problem was the strength of the ball: one of the first prototypes of the ball part was divided into top and bottom halves, both connected with 4 screws. This configuration resulted in a bad overall strength of the ball due to small contact area and pressing on the two halves. In the end, the ball part of the gimbal has been divided into front and back halves fixed with metal screws (5 at the top and 5 at the bottom). This provided a high level of stability and strength of this part. One problem remained however, and it still needs improvement: the holes for the tilt motor as well as the bearing are in both halves. This leads to a very low pressing of these 2 components which results in significantly lower reliability of the prototype. In order to solve this problem, the holes need to be shifted to only one half: either front or back one.

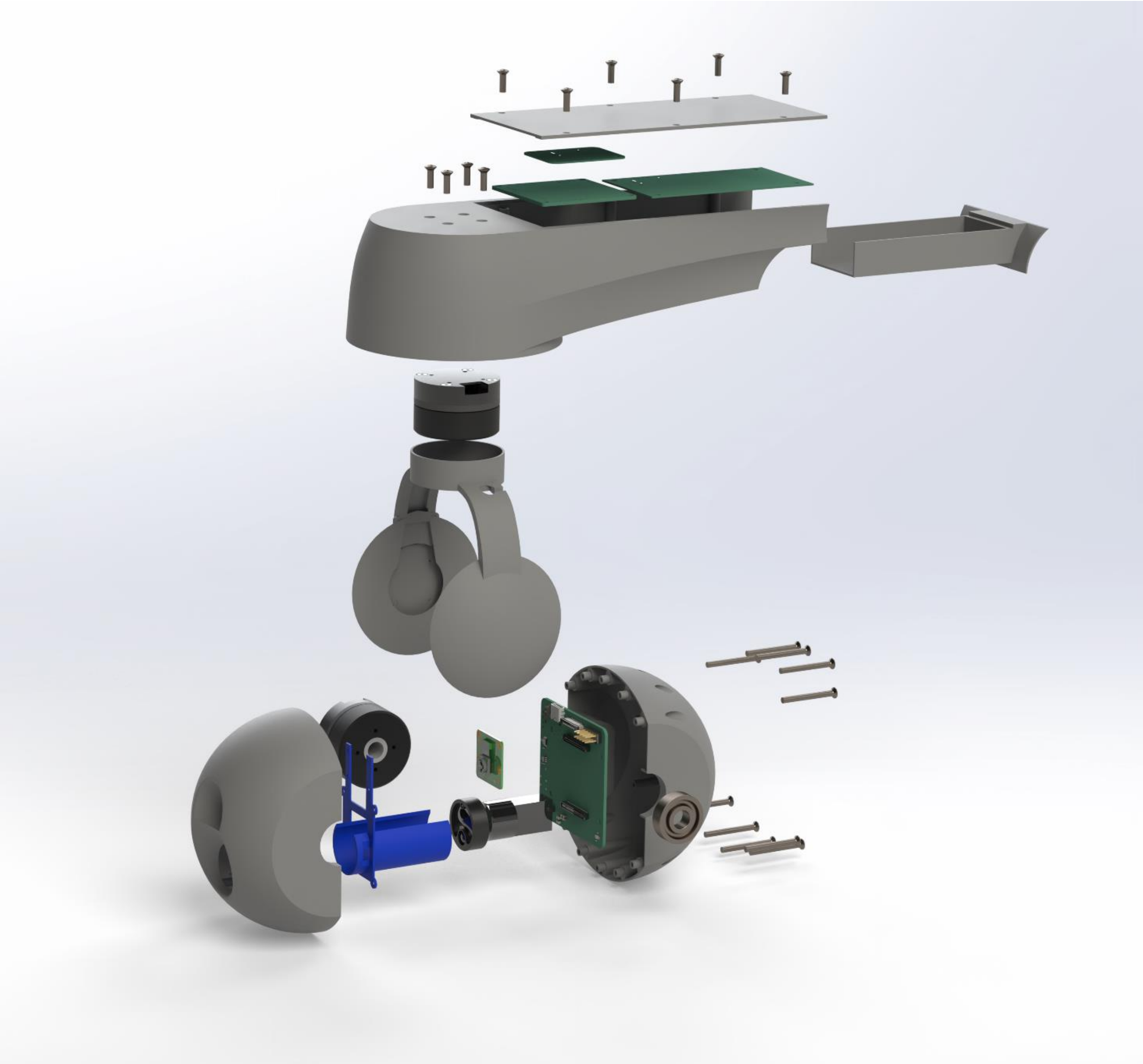
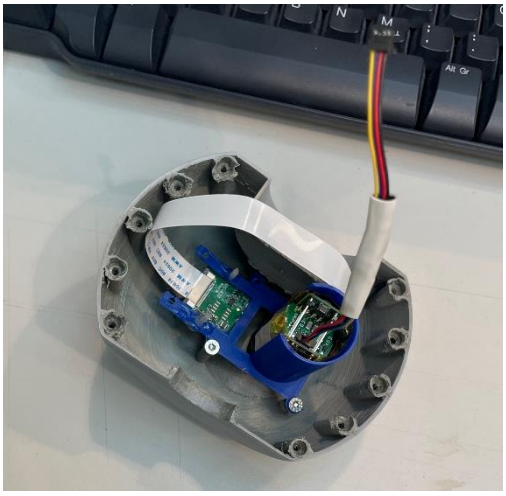
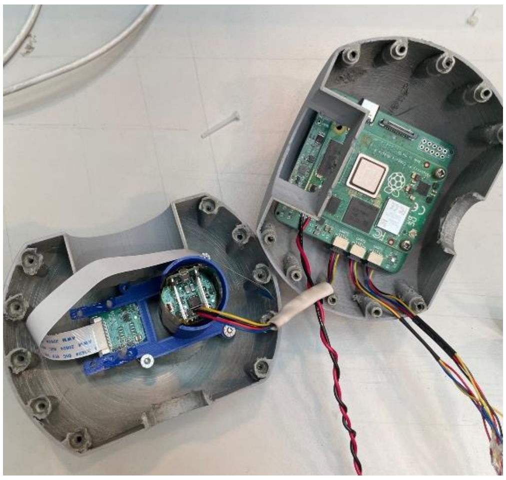
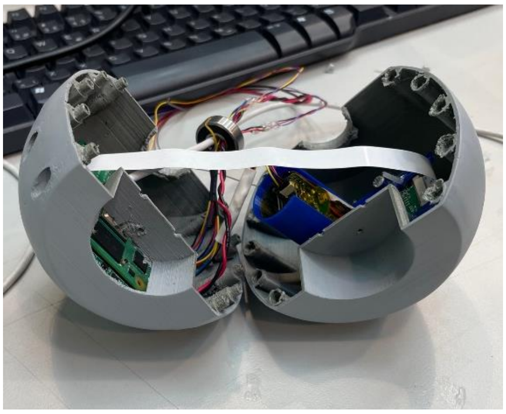
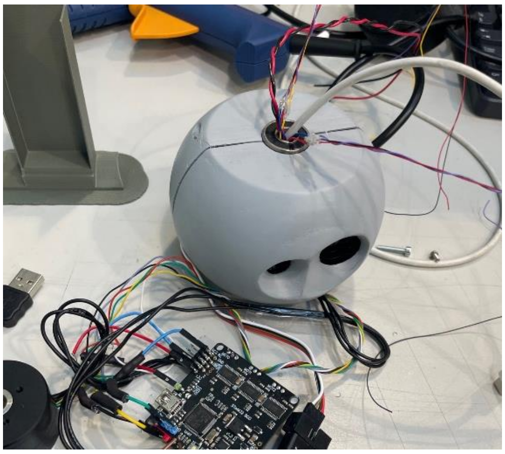
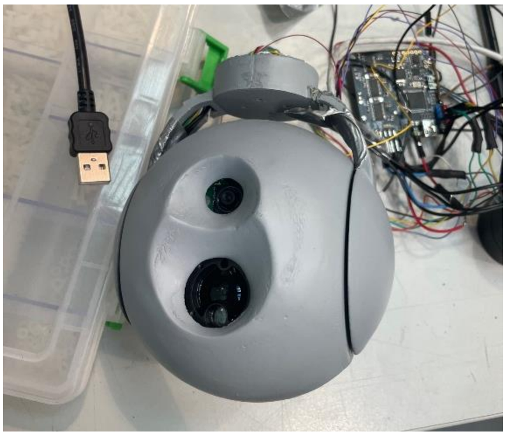
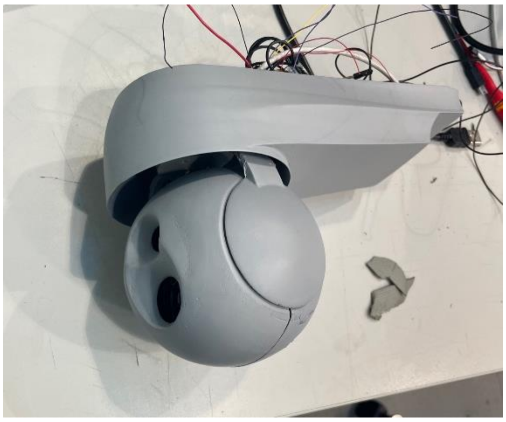


Figure 16. Explosion view of the prototype

4.5. *Assembly*

The whole assembly process of the prototype can be described by these pictures:

	Figure 17. Installation of camera module and laser into the holder (blue colored) and installation of it into the front part of the ball.
	Figure 18. Installation of the carrier board with the compute module into the back part of the ball and connecting the carrier board to the laser.
	Figure 19. Connection of the FPC camera cable to the carrier board. Pulling the wires through the bearing.

	Figure 20. Close the halves of the ball and screw them.
	Figure 21. Install the ball into the holder.
	Figure 22. Install the static part: screw the yaw motor, screw all the boards into the static part and close it with the lid.

5. Control and Communication systems

5.1. Requirements

Although the given requirements didn't specify much regarding how the system should be controlled or what kind of ranges should be achievable, quite early in the planning process it was understood that wireless communication might prove to be somewhat of a challenge. To be able to aim the system remotely, the airborne unit would need to transmit video that could be received on the ground and shown to the user. Additionally, the airborne system would have to be able to receive control communications from the user on the ground. Even though communication range was not a design priority we wanted the system to be reasonably usable when attached to a flying drone. This led to a decently challeging combination of requirements for a two-way communication link: High enough bandwith to transmit *preferably* HD quality video, low enough latency for the system to be controllable from the video feed, range of at least 100 meters but preferably more. Also, due to the multifaceted project, we were hoping to find a solution that would be as ready to work off-the-shelf as possible. Due to the communication system likely setting some requirements or limitations regarding our overall system, it was decided that we wanted to lock in on an option quite early in the planning process.

The initial solutions we investigated all had their share of issues for our use case. Protocols like ZigBee, Bluetooth and LoRa were all too low bandwidth for video transmission. Alternatively, WLAN, although high bandwidth, has too little range. Cellular would in most scenarios provide us high enough bandwidth and practically unlimited range but would be infrastructure reliant on cellular towers and at least some kind of server to handle the connection between the devices.

On top of these general use communication protocols, we decided to also research what kind of solutions are commonly used in video transmitting drones either in commercial or hobbyist situations. DJI, which holds most of the commercial drone market share (Höhrová, P., Soviar, J. and Sroka, W., 2023) at least has their proprietary technology for both control and video transmission. Although the more challenging aspect, the video transmitter and receiver would've been commercially available as a separate module, it would have been prohibitively expensive and required a separate control link. Other commercially available options shared these same issues.

On the hobbyist side on top of commercially available video transmission (VTX) modules, other common alternatives are either analog video transmission or WiFI based open-source alternatives. As analog VTX solutions would've required a separate control link, additional analog electronics on both the ground and air unit side and seemed to be in the progress losing their popularity among the FPV drone hobbyist, they we're initially put to the side as an option and additional research was focused upon the available open-source digital VTX solutions.

We were able to find two different drone usage focused open-source VTX projects: RubyFPV and OpenHD. Both projects provided software that could do long-range video transmission and additional telemetry and control link using 2.4Ghz or 5.8Ghz off-the-shelf Wi-Fi radios, but RubyFPV could not be considered for our use case due to its restrictive licensing.

5.2. OpenHD

Figure 23. The ground station running OpenHD and QOpenHD where the system can be controlled from

Ultimately we decided to choose OpenHD as our communication link and build our own software around it. OpenHD supported running on Raspberry Pi's, which we were already considering and supported a wide variety of common camera modules and a reasonably well available set of Wi-Fi adapters. It would then provide us with a long range two-directional communications link that should work to at least up to a few kilometers of range (OpenHD, 2024). OpenHD uses a modified version of Wi-Fi based on the Wifibroadcast project (befinitiv). The OpenHD link differs from normal utilising the monitor mode and packet injection offered by some Wi-Fi adapters to gain more analogue like link properties such as no required association between devices. This in addition to features like adaptive bitrate and forward error correction provides a link that can function at significantly higher ranges than normal Wi-Fi.

When paired with supported hardware such as our Rasberry Pi CM4, X86 laptop running Ubuntu 24.04 and supported Wi-Fi adapters (TP-LINK Archer T4U and ALFA AWUS036ACH v.2) OpenHD is relatively easy to setup using the image flashing utilities available on their Github.

We additionally utilized their companion application QOpenHD on the ground station to display the received video feed and telemetry, including our custom telemetry packages containing the target coordinates. OpenHD and QOpenHD are intended to be easily integrated with flight controllers using Mavlink protocol over UART which is common in hobbyist drone usage. To avoid having to modify OpenHD at all we connected our custom software to OpenHD using a pseudo terminal. This way we could pretend to be a flight controller and receive the required remote-control messages from OpenHD as well as send our own data down to the ground station. In the end we neede to worry very little about the datalink, including the video transmission or handling of the controller input. We only needed to pass our data in mavlink messages to the OpenHD air unit and if the link was stable it arrived at the ground unit.

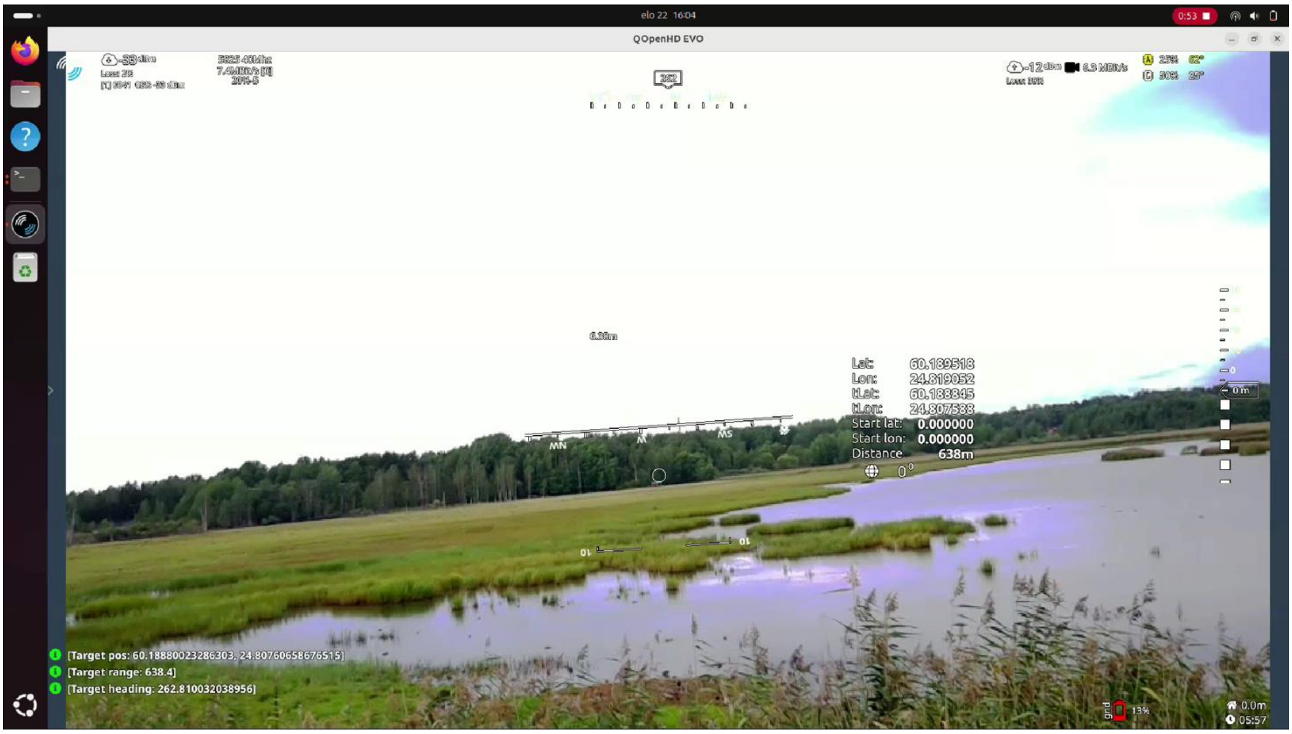


Figure 24. Ground unit interface

We did some minor modifications to the QOpenHD application to visualize the coordinate data, but the OpenHD air and ground unit ran otherwise unmodified OpenHD 2.6.2 software. The modified QOpenHD software is included in the project attachments.

5.3. Custom Python software

Due to OpenHD handling a significant portion of aspects for us, including some of the more compute heavy operations such as video processing, our custom software did not need to be very complex in the end. We mostly needed to communicate with sensors, parse their data, package in the Mavlink format OpenHD was expecting and pass it to openHD over our established virtual serial port. Additionally, we parsed incoming remote-control packages and forwarded user commands to the necessary components.

Due to an expected light compute load, ease of development and good availability of libraries for sensor parsing and using mavlink protocol, we decided to implement our custom software in python. This proved to be a good decision as we didn't run into any performance issues and python allowed us to iterate quickly on the software by updating the source code directly on the machine using a ssh connection over the Wi-Fi hotspot provided by OpenHD.

6. Other hardware

6.1. Controller

A standard Xbox 360 controller was chosen due to its ease of integration with OpenHD's remote control and its good availability. OpenHD should also support any other standard USB game controllers.

The following control mapping was implemented for the system:

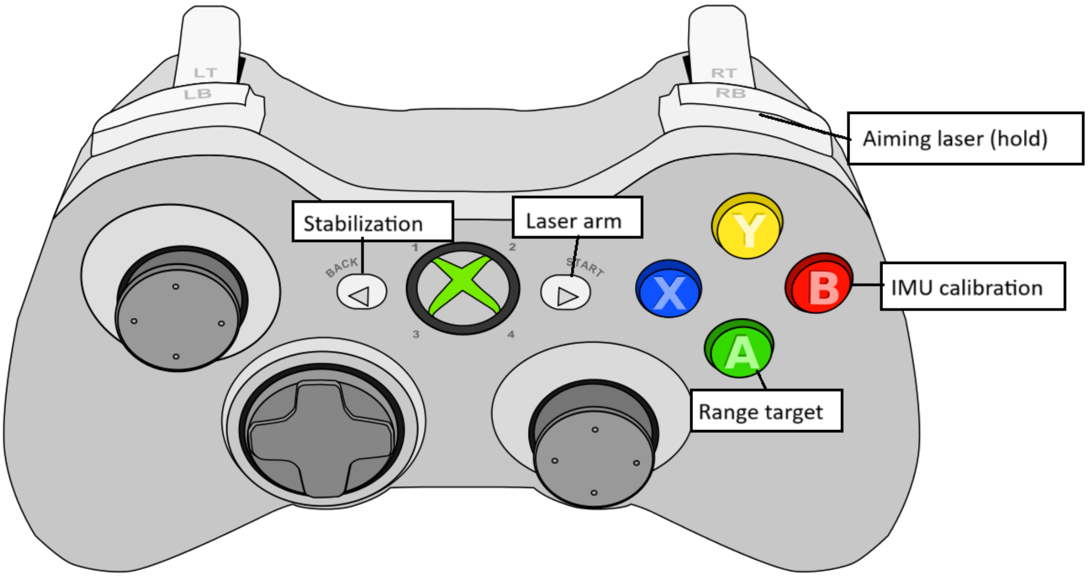


Figure 25. Labels of the joystick actuators

7. Reflection of the Project

7.1. Reaching objective

On a general level the system is functional, but there are two major problems impeding its usability. Firstly, the magnetometer on the main board's IMU provides unreliable heading readings and thus our target coordinates are inaccurate as well. This could be improved by using a higher quality magnetometer and, more importantly, having the magnetometer in a more isolated location where magnetic fields from the system's own electronics wouldn't affect the magnetometer. Secondly, friction from the enclosure and the wires makes aiming the system less than ideal. Possible improvements could come from having the system be less modular, which would reduce the number of wires and wire friction, smoothing the enclosure and having a more open design with less touching surfaces.

Also, due to time constraints, the system has not been properly tested with the GNSS module integrated into it. While the system performed well with a placeholder module and the GNSS module itself was extensively tested, a connection issue arose during integration. This problem appears to originate from the UART connection wire soldered onto the main board. Unfortunately, we haven't had sufficient time to resolve this issue.

Mandatory goals have been met with a caveat regarding the targeting: the system's weight is under the 1 kg goal. Targeting range has been tested up to 600 meters and could potentially work up to

1000 meters, according to the rangefinder specifications, easily meeting the 50-meter minimum requirement; however, the unreliable heading from the magnetometer affects the accuracy of the calculated target coordinates. Currently, the system displays the coordinates of the system and target, and the range. Additional data can also be added. The system is mountable with bolts and nuts, and additional mounting capabilities can be added.

Some of our own goals have been met: We quickly realized that our original goal of ±10-meter accuracy at 600 meters was overly ambitious and it was unlikely to be achieved. As of now, we haven't been able to properly test the accuracy due to unreliable heading data and the motor issues caused by friction created by inaccurate clearances of the 3D print of the prototype. While we haven't conducted a proper maximum range test for operating range, preliminary testing seemed to work up to ~200 meters with a clear line of sight between the ground and air unit and it showed promise of way higher ranges. Thus, we can consider our range goal fulfilled. Most of our deadlines were met, and the prototype is mostly functional by the final deadline.

As a team, we are generally pleased with the results. Communication among team members was mostly effective and timely, everyone got along with each other, and we worked well together. Throughout the project, we all learned new things and identified potential future upgrades and alternative solutions. Although, as of now, the system does not work exactly as intended, we consider this a successful project.

7.2. Timetable

Overall, we managed to stay decently well within the planned schedule. The early aspects of the project including planning, the overall solution and individual systems, took a bit more time than initially expected. This was partly due to the amount of time it took to find and compare alternative solutions.

Another aspect that took more time than expected was the sourcing of some specific components. This was especially time-consuming with some of the less commonly available parts, such as the laser range finder and the gimbal motor controller. In the end the gimbal motor controller proved to be one of the more time-consuming aspects of the project to plan, source and implement. It took us until the sixth week of the project to decide on a solution and even then, we ran into issues. The manufacturer of the gimbal controller + motor kit we were considering suddenly changed their minimum order quantity to five, preventing us from moving forward with that solution.

Due to the modular way the system was designed we could fortunately work on other aspects of the project while we didn't yet have the motor controller functional. In the end, a week or more of extra time to work more on fine tuning some aspects of the prototype might have proved beneficial, but with a few longer working days we managed to make do without that extra time.

7.3. Risk analysis

The risks of the project are listed below in figure 26. We managed our risks as we have written in mitigation sections. If we would do this risk table again there would be at least one additional risk which is: supplier delivers wrong products what we ordered. This happened one time when we ordered Wi-Fi adapters for our prototype. The risk itself is not very probable and the impact on the project's success is very low. This risk would go in the same kind of category as "Some ordered component doesn't arrive or arrives late" and mitigation of this risk would be the same (Figure 26.). Mostly remember to order critical components well in advance!

Risks which we realized mostly during the project (Especially at the end) were “Time runs out” and “We exceed our budget” (Figure 26.). During the project's late period, our Milestone 3 was late about 1.5 weeks, but during the planning phase we gave time buffers to every milestone. Eventually this arrangement ended up kind of saving our project for being late. So, we used our mitigation plan for this risk already in very early stage of the project. Another risk seemed to become very close to reality when we saw that PCB orders were very expensive, more expensive than we originally thought. After all we found some components from paja to use and also we decided to go without battery management system which saved money. Battery management system was not part of our project because we didn’t have time to make it. With our calculations we ended up using 1434.76 € (With VAT + shipping) so we didn’t exceed our budget limit which was 1488 (With VAT).

The risk that we would like to mention is that if some of our team members left the course in the middle of the project the probability of this prototype being delivered with the same kind of results would most likely not been possible. The idea of this project is very straight forward but the scale of this project is one of the largest if not largest in this course and five team members were needed. Also, the different backgrounds which each of us had in engineering were an asset to us: Electrical engineering, Computer science, Digital systems and Design, and Automation engineering with the hint of Industrial Engineering and Management.

The risk	Probability	Mitigation
We plan the prototype too complex and can't deliver prototype on time.	**High**	Break the project into smaller, manageable phases. Prioritize essential features for the initial prototype. Regularly review progress to ensure alignment with goals. Focus on doing what is mandatory!
We exceed our budget.	**Low**	Establish a detailed budget with contingencies for unforeseen expenses. Monitor spending closely and adjust plans promptly if costs begin to exceed projections.
Time runs out.	**Medium**	Create a realistic timeline with buffer periods for unexpected delays. Use agile project management techniques to maintain flexibility and adaptability.
Some ordered component doesn't arrive or arrives late.	**Medium**	Order critical components well in advance. Identify alternative suppliers and maintain a list of backup options. Actively keep track on deliver status.
Some planned aspect of the project needs to be changed significantly possibly affecting other parts of the project.	**Medium**	Conduct thorough planning and risk assessment at the start. Maintain a flexible design that can accommodate changes without major disruptions. Do prototyping well in advance and report immediately from possible changes.

Figure 26. Risk table

Overall, almost all the risks mentioned in the project plan’s risk table were considered except that supplier can deliver wrong products. All the risks were managed well and none of them ended up ruining our project. Some of the risks mentioned were more probable than originally expected: We exceed our budget and time runs out. Despite some setbacks, we delivered prototypes matching original mandatory requirements from risk analysis.

7.4. Challenges in the project

Balancing reactiveness and precision when choosing the best option for the mechanical design. Long range targeting requires high precision but at the same time a drone isn’t a completely stable platform, especially in adverse conditions, leading to the need of highly reactive motors that can stabilize the system's camera. We ended up using a generally recommended and used solution of BLDC gimbal motors.

We also did research about worm drive solutions, which required more mechanical proficiency compared to BLDC motors. The main challenge of the worm drive solution is that it is not that reactive compared to BLDC motors which we ended up using in our prototype. Because of this, it might be hard to build reactive enough stabilization for the gimbal itself. However, we didn't have enough time to test stabilization or motors with our enclosure.

Multiple sources of error in targeting compound quickly, making the camera accuracy and overall usability of the system in longer ranges less than ideal.

7.5. Suggestions for improvement

The magnetometer (in our case the IMU that is on the CM4 carrier board) used for getting the heading of the camera should be isolated from other electronics to get as clear of a reading of the Earth's magnetic field with no interference from the magnetic fields of the electronics on board the system. Currently the IMU on the board is placed directly under one of the BLDC motors, which we believe may be causing significant interference to the magnetometer.

A less sophisticated enclosure, at least for the beginning, would allow for faster iteration. Our design is fitting for an end product, but for prototype testing and iteration it is not ideal due to reassembly being cumbersome.

The static part of the enclosure could be redesigned so that the gimbal head would be more in the center. As of right now, the weight distribution of the system is skewed to the front due to the gimbal being there, which we have compensated for by having mounting holes closer to the center of mass in the front, but a more centered gimbal design might be more fitting to a wider range of drone designs.

Different methods for wireless communication should be researched. The bandwidth required for wireless video transmission can be hard to achieve, especially in longer ranges, and a more optimized solution probably can be found ready-made or made in-house. The choice for OpenHD was made based on it being more or less ready out of the box open-source solution for first-person-view drones with an active community and good documentation. It seemed to be and has been a good fit for our project but research into other solutions should be done. Also, video transmission and telemetry could be split into different transmission methods, which would e.g. allow telemetry to be still used when video transmission cuts off – as of now OpenHD uses the same connection for both telemetry and video.

Currently the system has no way of informing the user about the state of the battery, which means that the system could suddenly lose power without warning the user about it. To fix this, we would need some sort of battery management system that can monitor the battery life and relay that information to the main board. Additionally, the current prototype provides no means to charge the battery. This means that the battery must be removed from the enclosure and charged externally. Alternatively, in a potential future iteration the 2S LiPo could be replaced by using two 18650 Li-Ion batteries in series. These batteries would be extremely easy and fast to replace, meaning that the integrated charging circuitry could potentially be unnecessary.

8. Discussion and Conclusions

The project succeeded in developing a prototype of a drone-mounted targeting system, meeting mostly all objectives and goals of the project. However, a few certain challenges remain, particularly concerning the precise aiming required of the system and the acquisition of reliable heading data. The friction issues in the enclosure and unreliable magnetometer readings need to be addressed in future iterations.

OpenHD proved to be an excellent choice for wireless communication between the air and ground units for this project. It provides a reliable bidirectional communication method and video transmission out of the box with minimal need for adjustments. Modifying it also was a non-issue. However, further research into different video transmission solutions is recommended, as we may have overlooked some options. Developing a custom video transmission system was beyond the scope of this project and due to it we heavily focused on already tested solutions, but custom transmission should be considered in future iterations if required.

For mechanical parts, BLDC gimbal motors are a good choice. They offer good stability, accuracy, response time and smooth movement, which are all needed in a drone system. They do require more resources (cost, power and specific control electronics) but they were a non-issue in our project.

A worm drive solution has potential to be highly accurate for long range targeting. However, the stabilization would not be as good as in a more reactive BLDC motor solution. Worm drive solution itself is a great option for this similar type of system if you are looking for:

- High precision for long range
- Maximizing battery life
- Reduced weight
- High torque

Also, it would be interesting to see an integrated combination of these two mechanical solutions and how they would work together. There would be more reactive motors (ie. BLDC) which would do the stabilization and the faster movements when user is not targeting, but then there is also the precise worm drive solution integrated inside which would be used for precise targeting in longer ranges.

During the project we have improved our skills of working in both mechanical and electrical CADs, specifically SolidWorks, SolidEdge and KiCad. We have also broadened our knowledge of working with embedded systems as well as learnt a lot about different stages of making a prototype for a physical technological product.

List of Appendixes

- Archive with 3D models, PCB layouts and schematics, firmware, software and hardware installation instructions

References

Höhrová, P., Soviar, J. and Sroka, W., 2023. Market Analysis of Drones for Civil Use. LOGI–Scientific Journal on Transport and Logistics, 14(1), pp.55-65.

OpenHD, accessed 29.08.2024 at: https://openhd.gitbook.io/open-hd/general/troubleshooting/faq

Wifibroadcast, befinitiv accessed 29.08.2024 at: https://befinitiv.wordpress.com/wifibroadcast-analog-like-transmission-of-live-video-data/

u-blox MAX-M10S Integration manual: https://content.u-blox.com/sites/default/files/MAX-M10S_IntegrationManual_UBX-20053088.pdf

Title page 3D model of the quadcopter: https://www.cgtrader.com/products/qx-9

3D model of the GM3506 motor for the figure 16: https://grabcad.com/library/iflight-ipower-gbm3506-130t-with-encoder-1