BRNO UNIVERSITY OF TECHNOLOGY

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ

DEPARTMENT OF RADIO ELECTRONICS

ÚSTAV RADIOELEKTRONIKY

STEPPER MOTOR DRIVER WITH FEEDBACK FOR POSITIONING PLATFORM

DRIVER KROKOVÉHO MOTORU SE ZPĚTNOU VAZBOU PRO POLOHOVACÍ PLATFORMU

BACHELOR'S THESIS

BAKALÁŘSKÁ PRÁCE

AUTHOR AUTOR PRÁCE	Michael Doležel
SUPERVISOR VEDOUCÍ PRÁCE	Ing. Peter Barcík, Ph.D.

BRNO 2024

BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

Bachelor's Thesis

Bachelor's study program **Electronics and Communication Technologies**

Department of Radio Electronics

Student: Michael Doležel ***ID:*** 230241

Year of study: 3 ***Academic year:*** 2023/24

TITLE OF THESIS:

Stepper motor driver with feedback for positioning platform

INSTRUCTION:

The aim of the thesis is to design and construct a stepper motor driver with feedback for a positioning platform of a wireless optical link. The positioning platform allows azimuth and elevation adjustments within a range of +-5°. Compare available sensors for absolute elevation and azimuth sensing. Design the block and circuit diagram of the driver, considering the sensor's implementation into the existing platform.

Create comprehensive design documentation for the realization of the design (circuit diagram, printed circuit board design, layout, bill of materials, mechanical design, etc.). Implement the proposed device as a functional prototype and verify its operation through experimental measurements in the laboratory.

RECOMMENDED LITERATURE:

[1] ACARNLEY, Paul, 2002. Stepping Motors: A guide to theory and practice (Control, Robotics and Sensors). 4th edition. The Institution of Engineering and Technology. ISBN ↱978-0852964170.

[2] TI Designs Closed Loop Stepper Motor Design With Encoder for Stall- Detection Reference Design [online], 2017. Dallas, Texas: Texas Instruments Incorporated [cit. 2023-05-22]. Dostupné z:
https://www.ti.com/lit/ug/tiducy6/tiducy6.pdf?ts=1684716950617&ref_url=https%253A%252F%252Fwww.google.ca%252F

[3] Fundamentals of motor control: Industrial & Power conversion Training [online]. STMicroelectronics [cit.2023-05-22]. Dostupné z:
https://www.st.com/content/ccc/resource/sales_and_marketing/presentation/application_presentation/group0/23/a1/94/a3/39/cf/4c/37/introduction_to_electric_motors_pres.pdf/files/introduction_to_electric_motors_pres.pdf/jcr:content/translations/en.introduction_to_electric_motors_pres.pdf

Date of project specification: 16.2.2024 ***Deadline for submission:*** 27.5.2024

Supervisor: Ing. Peter Barcík, Ph.D.

doc. Ing. Lucie Hudcová, Ph.D.
Chair of study program board

ABSTRACT

This bachelor's thesis deals with designing and building a closed-loop control system for the altitude and azimuth axis of an opto-mechanical platform. The goal is to compare rotation sensors, and then design a driver for the stepper motor. The control system prototype was built using the Arduino platform, DRV8825 motor driver, NEMA-17 stepper motor, and AEAT-9922 hall effect rotational sensor. The prototype can rotate the stepper motor with a resolution of 0.056° which is the maximal accuracy of the DRV8825 driver with 1/32 microstepping mode. The board of the final driver was redesigned to include the SPI-4 interface for the AEAT-9922 hall sensor. This driver includes an L6470H stepper motor driver with 1/128 microstepping mode. This final board has a resolution of 0.014°.

KEYWORDS

AEAT-9922, Arduino, capacitive encoder, DRV8825, Hall-effect sensor, L6470H, magnetic encoder, microstepping, NEMA 17, optical encoder, Opto-mechanical platform, potentiometer, resolver, rotatory sensor, rotation, RS-485, SPI-4, stepper motor, STM32F303K6

Typeset by the thesis package, version 4.09; https://latex.fekt.vut.cz/

DOLEŽEL, Michael. *Stepper motor driver with feedback for positioning platform*. Bachelor's Thesis. Brno: Brno University of Technology, Faculty of Electrical Engineering and Communication, Department of Radio Electronics, 2024. Advised by Ing. Peter Barcík, Ph.D

Author's Declaration

Author:	Michael Doležel
Author's ID:	230241
Paper type:	Bachelor's Thesis
Academic year:	2023/24
Topic:	Stepper motor driver with feedback for positioning platform

I declare that I have written this paper independently, under the guidance of the advisor and using exclusively the technical references and other sources of information cited in the paper and listed in the comprehensive bibliography at the end of the paper.

As the author, I furthermore declare that, with respect to the creation of this paper, I have not infringed any copyright or violated anyone's personal and/or ownership rights. In this context, I am fully aware of the consequences of breaking Regulation § 11 of the Copyright Act No. 121/2000 Coll. of the Czech Republic, as amended, and of any breach of rights related to intellectual property or introduced within amendments to relevant Acts such as the Intellectual Property Act or the Criminal Code, Act No. 40/2009 Coll. of the Czech Republic, Section 2, Head VI, Part 4.

Brno ....

author's signature*

*The author signs only in the printed version.

ACKNOWLEDGEMENT

I would like to thank my bachelor's thesis supervisor, Ing. Peter Barcík, Ph.D., for his professional guidance, consultations, patience, and insightful suggestions for the work.

Contents

List of Figures

List of Tables

Introduction

This bachelor's thesis deals with movement control of the opto-mechanical platform. Movement is in the altitude and azimuth angle. Each angular coordinate is rotated with one stepper motor. Due to the absence of a sensor, the position was measured by the number of steps, which is not accurate. Inaccuracy causes the nonlinearity of the stepper motor driver, the mechanical friction of mechanisms, and the precision of the stepper motor. This problem can be solved with an additional rotational sensor, that could give the driver board information about the current angle and if more or fewer steps are needed to meet the targeted angle.

The primary goal of this bachelor's thesis is to design a functional driver that will control each of the angular coordinates. Driver board will get data about the current angle from AEAT-9922 3D hall-effect sensor and communication will be carried out by SPI-4 (*Serial Peripheral Interface with 4 wires*). The secondary goal is to compare available rotational sensors.

The primary goal can be reached by a few steps. At first communication and angle data reading from an AEAT-9922 sensor must be established. The next goal is to control the stepper motor and utilize microstepping for finer step resolution. Then prototype needs to be constructed with the use of a microcontroller like the Arduino platform. After that, the driver board needs to be redesigned with an additional SPI-4 interface for the rotational sensor. The final step is to test the redesigned driver for functionality. An additional task is to design in Autodesk Fusion the holder for the magnet and AEAT-9922 sensor. This additional sensor will measure the elevation angle.

This thesis is divided into three main parts. The first is more theoretical, where the principles and properties of rotational sensors are described. Following this, all types of conventional rotational sensors are compared. The second part of this thesis is about designing of the prototype driver module, and then about the final driver board. This part contains information about the communication and data processing of the AEAT-9922 rotational sensor and stepper motor driver DRV8825 and the integration of the prototype driver module and its testing. The last part is about the design and description of the program of the final motor driver board.

1 Rotation sensors

Rotational sensors are widely used in many fields from industrial, military, scientific, and robotic to space applications. Each sensor has a different measuring method and properties. Some sensors produce output data and others need an external board to process analog signals. The main categories of rotational sensors are:

- Potentiometers
- Resolvers
- Encoders
  - Contact encoders
  - Optical encoders
  - Magnetic encoders
  - Capacitive encoders
- Hall effect sensors

1.1 Requirements for rotation sensors

In this work, one of the objectives is to measure the position of the opto-mechanical platform in altitude and azimuth angle. The movement of the opto-mechanical platform is ±5 degrees in both axes and precision should be in the ideal case hundredths of a degree.

The requirements for rotational position sensors are usually as follows:

- Accuracy (*how close a measured value is to the true or target value*)
- Resolution (*how many different positions there are for one full rotation*)
- Range of measurement (*in which angle range can senor measure*)
- Interface (*for example SPI-4, UART, I2C , or analog signal*)
- Durability (*not really that important in this case but when the sensor is used in industrial applications this is essential*)
- Size
- Cost

1.2 Types of sensors for absolute angle measurement

1.2.1 Potentiometers

Potentiometers change resistance based on the position of a sliding contact on a resistive element. This results in analog voltage output and requires analog-to-digital hardware for digital applications. The output voltage is often 0 V - V_{supply}.
Regular potentiometers offer moderate accuracy when appropriately calibrated but are prone to degradation over time due to mechanical wear. Resolution is limited by analog signal processing hardware and potentiometer accuracy. They must also be protected against vibrations, shock, and contamination of foreign matter. [1] [2]

Industrial grade potentiometers can be very precise with resolution up to 0.007° and linearity of ± 0.025 %. They also offer a long life of up to 100×10^6 movements which is more than enough for more static systems but can still be problematic for long-term high-speed applications. The protection class of these potentiometers can be up to IP65, which means fully protected against dust and jets of water. Protection against shocks and vibrations can be great with up to 10 G for vibrations in the range of 10-1500 Hz and 50 G for shocks. The maximal temperature limit for most durable ones is around 70°C.
The main disadvantage of precise industrial-grade rotary potentiometers is their high price. [4] [5] [6] [7]

Two main types of potentiometers are single-turn and multiturn. Another division is to their relationship of increase of resistivity with their angle, most often they are linear, logarithmic, or reverse logarithmic.

Fig. 1.1: Industrial grade potentiometer AWS360 [7]

1.2.2 Resolvers

Resolver is a type of special rotary transformer. The two main components of the resolver are a cylindrical rotor and stator. Both have two sets of windings. On the stator is the primary winding that inducts sine AC signal to the rotors primary winding, this part is called the rotary transformer. The secondary winding of the rotor then rotates with the measured motor shaft. The second set of stator windings is a pair of sine and cosine windings, that are phase-shifted. [8] [9]

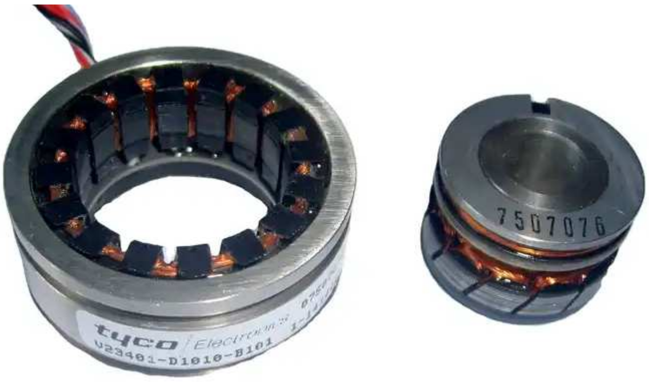


Fig. 1.2: Frameless resolver [10]

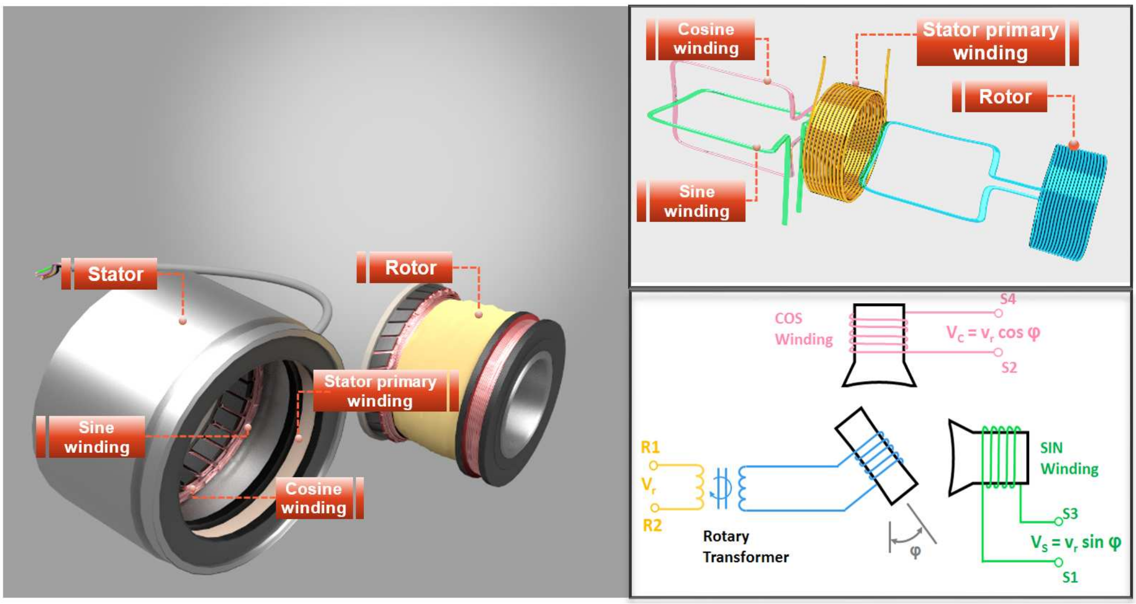


Fig. 1.3: On the left are components of a resolver, on the right top is an illustration of windings and on the right bottom is a schematic illustration of the resolver [11]

Depending on the position of the rotor a sine signal on the sine winding and a cosine signal on the cosine winding are alternately induced. If the rotor winding is parallel to the stator sine winding and perpendicular to the cosine winding, then the sin signal is maximum and the cos signal is minimal. [11]

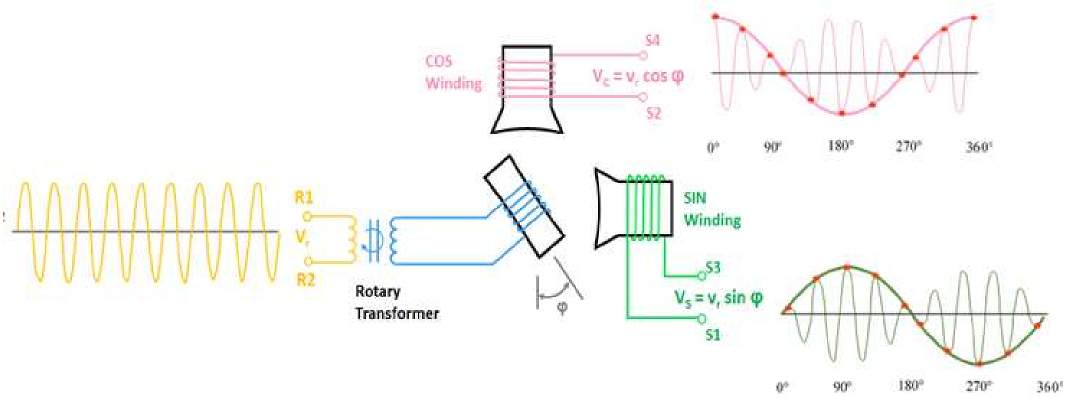


Fig. 1.4: Resolver scheme with input AC signal and output sin and cos signals from windings [11]

Position or angle is the arctangents of the voltage output of the sine winding divided by the output of the cosine winding. [12]

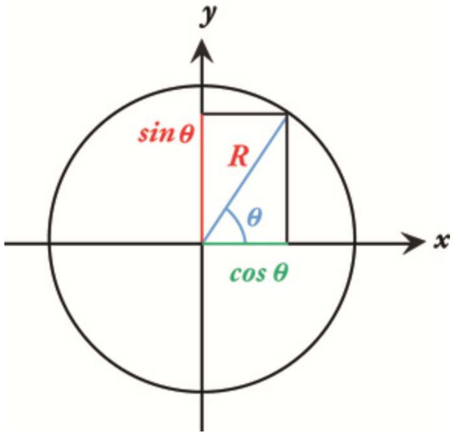


Fig. 1.5: For a given angle θ, we can express the corresponding point on a unit circle (R = 1) in terms of $\sin\theta$ and $\cos\theta$ [12]

To determine the angle of the motor shaft with the implemented resolver, we need to take the arctan of the ratio of the two voltages from sine and cosine winding of the resolver: [12]

$$\theta = tan^{-1}(\frac{sin\theta}{cos\theta})$$

Resolvers do not include any electronic components on their own. so they are much more robust than encoders and can survive violent temperatures, radiation, vibrations, and shock. Mainly for their durability and simplicity, they are ideal for industrial, military, nuclear power, and space applications, when increased durability is required. The downside is that they require separate analog to digital converter. [8]

The accuracy of resolvers can be increased by adding additional wiring, this type of resolvers is named multi-speed resolvers and they generate multiple sine/cosine cycles per mechanical revolution. [13]

Resolvers can have resolution up to 0.05° and environmental protection up to IP65 (fully protected against dust and jets of water). Resolvers can have also great temperature resistance with a maximal temperature of 200°C. They can be either housed or frameless. [15]

1.2.3 Rotary encoders

The basic principle of rotary encoders is to generate incremental signal or absolute position data with an encoded disc that can be read by a few physical principles. [2]

Contact Encoders

They use mechanical contact between the brush or pin sensor and the coded disc. On the disc is a series of metallic rings or tracks that are joined at their base to have the same logic level. As the disc rotates, sliding contacts fixed to a stationary part of the encoder create conductive contact with the disc, which creates a series of square wave patterns that represent the rotation position.
They tend to wear due to their mechanical contact principle. Contact encoders are not precise, they have most often a resolution of around 24 pulses per revolution (resolution of 15° per pulse). [16] [17]

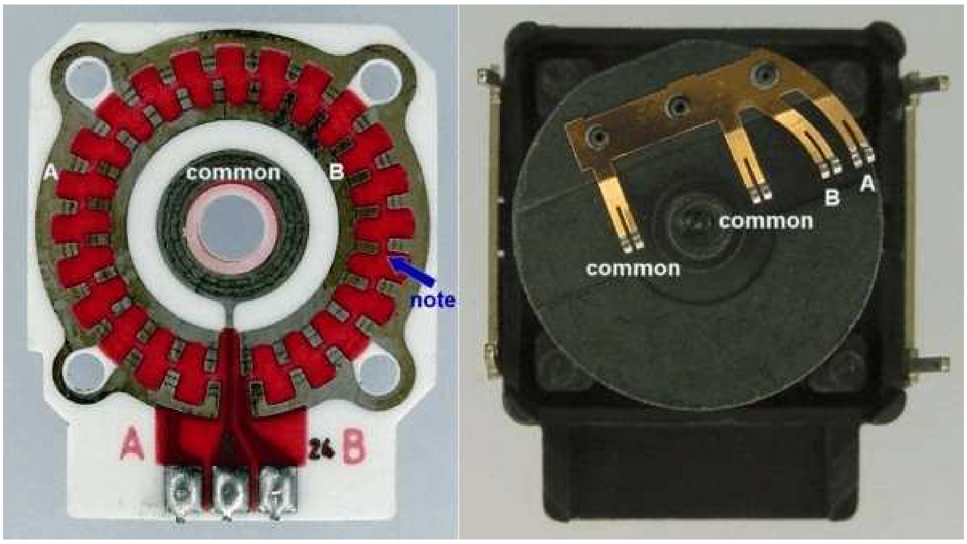


Fig. 1.6: Incremental contact encoder disc [18]

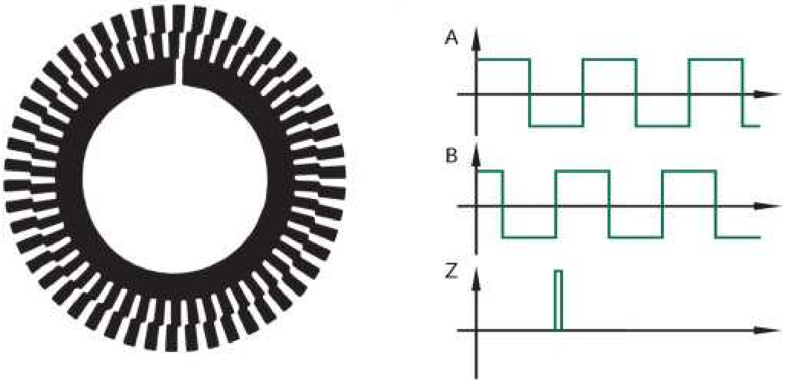


Fig. 1.7: Contact encoder output, direction of rotation can be determined by the relative position of A and B signal, and zero position by common signal. [19]

Non-contact Encoders

Non-contact encoders use different physical phenomena than conductive contact to read the coded disc. [2]

Types of non-contact encoders are:

- **Optical encoders**
- **Capacitive encoders**
- **Magnetic encoders**

Optical Encoders

Optical encoders use the patterned disc to encode position information similarly to contact encoders but the reading method differs. Their discs have transparent and non-transparent segments. A light source is usually on one side and sensors are on the other. Each sensor reads one ring track. Each separate ring track produces one determined bit in the binary string. If the encoder has 15 separate ring tracks then the encoder has $2^{15} = 32768$ unique binary combinations, with a resolution of $\frac{360°}{32768} = 0.011°$ per binary value.

They can use light sources like LEDs and light sensors are usually photodiodes. This type of non-contact encoder was used to eliminate mechanical wear problems that can be problematic at high-speed applications. [2] [20]

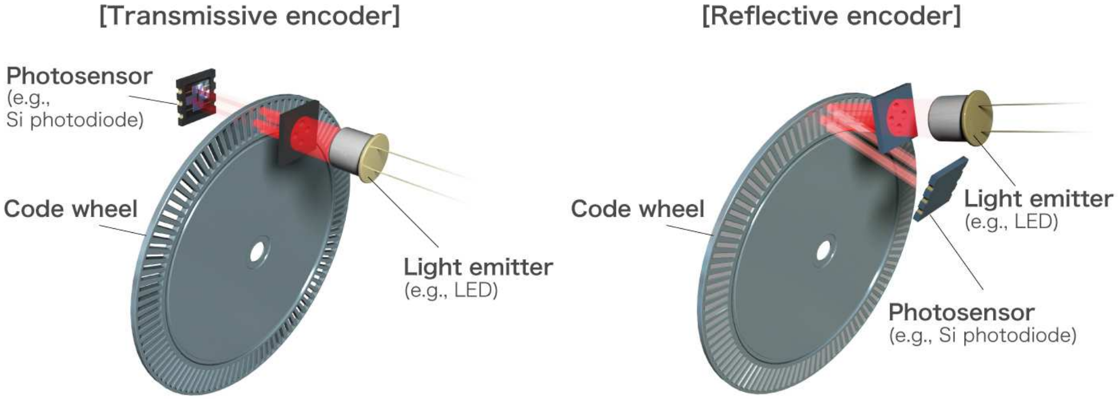


Fig. 1.8: Incremental optical encoder with different reading methods [21]

Optical encoders are available in both incremental and absolute configurations.

- **Absolute output:** Coded disc patterns where the specific section on the disc represents absolute angular position with unique binary value. [22]
- **Incremental output:** If the disc code pattern is replaced with a series of equally spaced transparent segments in one or two circles, encoder output becomes a series of incremental pulses. They provide positional information based on a number of pulses from some zero reference point. Usually, incremental encoders include dual channel output for direction sensing function and can include the third one as reference zero position. The weakness of this type of optic encoder is that it needs to be homed to zero position after power loss. They are simpler to use and less expensive than absolute optic encoders. [22]

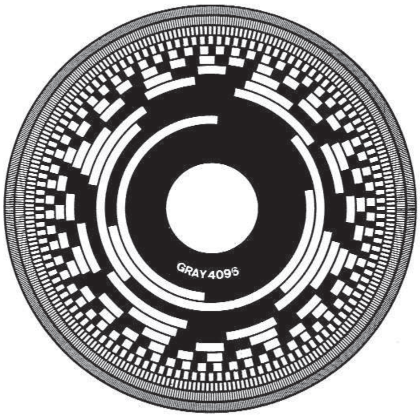


Fig. 1.9: 12bit Absolute optical encoder with gray code [22]

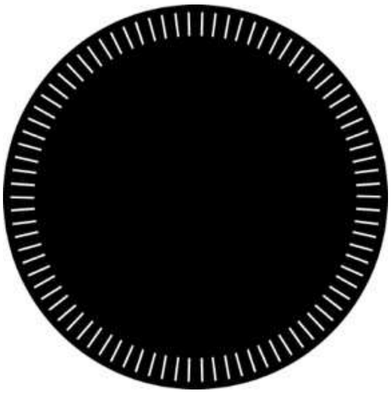


Fig. 1.10: 1bit Incremental optical encoder [22]

Two codes can be used on the encoder disc:

- **Binary code:** We can use standard binary code. This code has one major issue. When the disc rotates and transits between logical 1 and logical 0, there can be uncertainty in specific positions. The encoder could read improper positions and cause false readings. The worst situation is when disc transits between 111 to 000. [23]
- **Gray code:** Gray code was named after *Frank Gray* who patented its use for shaft encoders in 1953. The main difference against binary code is that only one bit is changing per position. This feature reduces amount false readings linked with binary counterparts. [23]

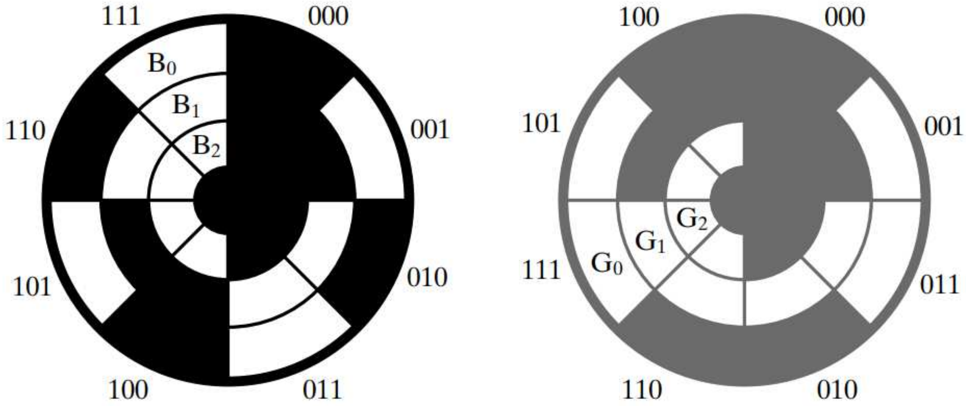


Fig. 1.11: Gray and Binary encoder disc [23]

Decimal	Binary	Gray
0	**000**	**0**00
1	00**1**	00**1**
2	0**10**	0**1**1
3	01**1**	01**0**
4	**100**	**1**10
5	10**1**	11**1**
6	1**10**	1**0**1
7	11**1**	10**0**

Table 1.1: Comparison table of encoder code types (*changing bits are marked*) [23]

Magnetic Encoders

Magnetic encoders use magnetized discs with magnetic flux patterns. This encoder can read changes in flux and generate sinusoidal signal output. Two main components are a disc that is magnetized with south and north poles that are lined around the perimeter of the disc. The second component is the sensor which can be either a Hall-effect sensor.
The output signal is a sine wave that is afterward converted into a square wave signal to be fed to the driver. The resolution mainly depends on magnetized pole pairs (can be from 2 up to 14 pole commutation tracks - most useful for absolute position measurement) and a number of sensors. [24]

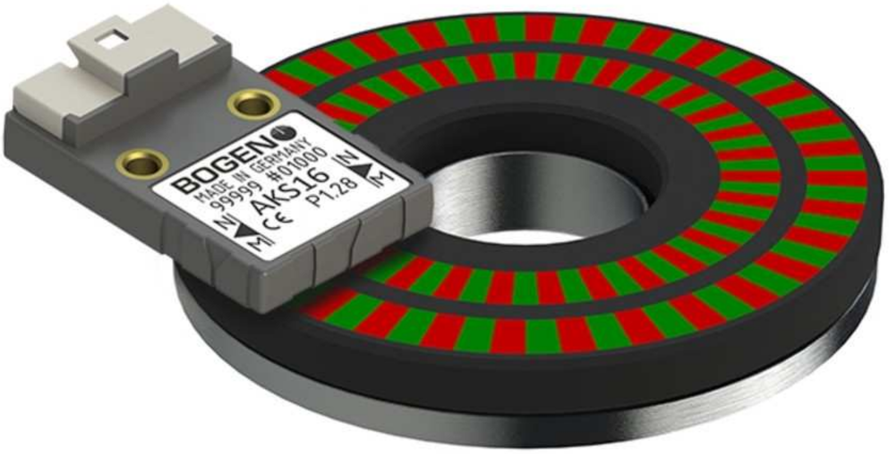


Fig. 1.12: Magnetic encoder AKS16 [25]

The resolution and accuracy of magnetic encoders is generally lower than of optical encoders. The main advantage of magnetic encoders is durability.
Optical encoders can be damaged by 3 main kinds of damage.

- Optical discs can be shuttered by vibrations or shocks
- Oil, moisture, dust, or water can get inside and cause improper readings
- LEDs or sensors can malfunction

All of these failures the magnetic encoder effectively eliminates by its design. For example, most contaminants do not have any effects on the magnetic field. They also can withstand higher temperatures and radiation levels.
In conclusion, magnetic encoders have lower resolution than optical ones but can be used in more demanding industrial environments. [24]

Capacitive Encoders

Capacitive encoders can identify changes in the capacitance of a disc using a high-frequency signal to measure rotational angle. They are composed of three major components. A stationary transmitter, that transmits a high-frequency electrical signal. Disc stamped with a sinusoidal pattern, that modulates the transmitted signal. The stationary receiver reads modulations and converts modulated signals to increments of rotatory motion. [26]

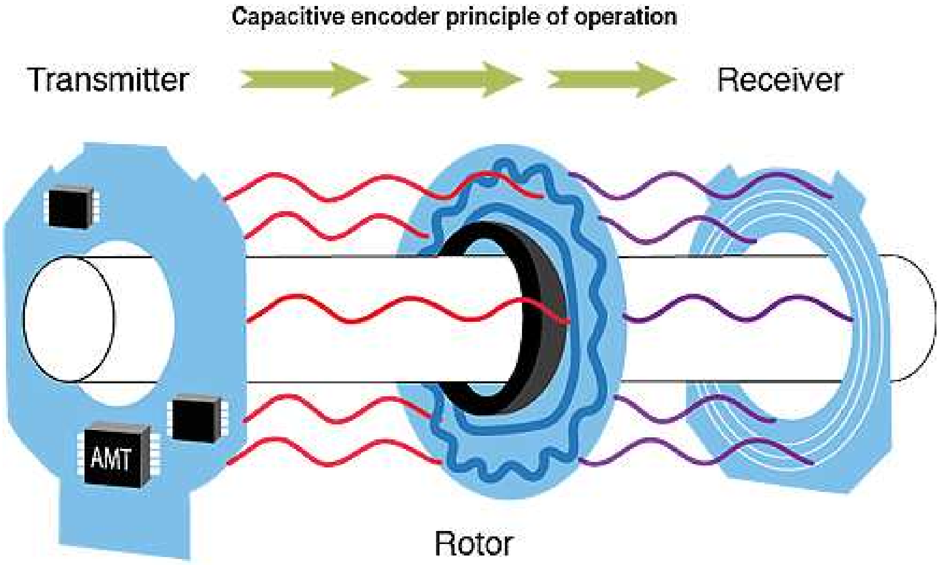


Fig. 1.13: Capacitive encoder [29]

Magnetic encoders are durable but are unable to provide resolution and accuracy of optical encoders. Capacitive encoders combine the accuracy and resolution of optical encoders and the environmental resistance of magnetic encoders. so they take the best properties of both. They are often protected against dust, dirt, and oils. They can operate in wider temperature ranges of minimal -40°C up to maximal 125°C. They are also more robust against vibrations and shocks. On average capacitive encoders draw less than one-tenth of current comparable to optical encoders and also less than magnetic ones allowing them to be effectively used in battery-powered applications.

Another advantage is the capability to change resolution without the need to change components (*In capacitive encoder AMT11 resolution can be set from 48 to 4096 Cycles Per Revolution (CPR)*). The capacitive encoder size is smaller than the optical ones. [26]

Real encoders properties

Series	Encoder type	Operating temperature	Resolution
A5SI	Optical	-40 - 100°C	50 - 5000 CPR
A6SIH	Optical	-40 - 100°C	100 - 10000 CPR
EC35	Optical	-20 - 105°C	500 - 60000 CPR
M15l	Magnetic	-40 - 125°C	100 - 2048 CPR
AMT10	Capacitive	-40 - 100°C	48 - 2048 PPR
AMT11	Capacitive	-40 - 125°C	500 - 4096 PPR

Table 1.2: Real encoder properties. Capacitive encoder resolutions are programmable, all other types are model-dependent and cannot be changed. [26] [27] [28]

Note: CPR-Cycles Per Revolution, PPR-Pulses Per Revolution, ATM models are Incremental

Encoder type	Capacitive	Optical	Magnetic
Contaminant Resistance	High	High	Low
Accuracy	High	High	Low
Temperature Range	Wide	Medium	Narrow
Current Consumption	Low	High	Medium
Programmability of resolution	Yes	No	No
Package Size	Small	Medium	Medium
EMC Immunity	High	High	High
Magnetic Immunity	High	High	Low
Resolution Range	Wide	Wide	Narrow

Table 1.3: Comparison table of encoder types [30]

1.2.4 Hall Effect Sensors

Hall-effect sensors are integrated circuits that transform changes in magnetic fields into electrical signals.

- **Hall-effect latch:** Indicates most recently measured magnetic flux density. These are used in rotary applications and incremental encoding (*Sensor toggles low in the presence of the south pole and high in the presence of the north pole*). They can be used in magnetical encoders.
- **Hall-effect switch:** Indicates the presence or absence of magnetic flux density compared to a defined threshold. In principle, they can work as on/off

switches. The unipolar type reacts to the south magnetic pole and the omnipolar type reacts to both poles.

- **Linear Hall-effect sensor:** Output is a signal that is proportional to the strength of magnetic flux density. They measure precise movement to detect the absolute position on one axis.

Hall-effect absolute angle measurement using 3D position sensor

Absolute angle measurement can be achieved by measuring the magnetic flux of the diamagnetic magnet using multiple sensor elements in one sensor. They measure the strength of magnetic flux in multiple axes (X, Y, Z). 3D Hall-effect position sensors can be placed on-axis which is simple to implement. iÍn-plane can measure only two field components. Out-off-plane, also called off-axis describes any other placement. [31] [32]

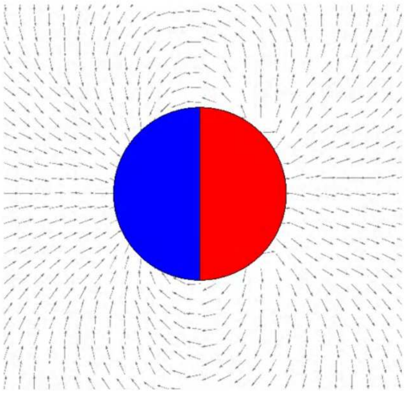


Fig. 1.14: Diametric magnet [32]

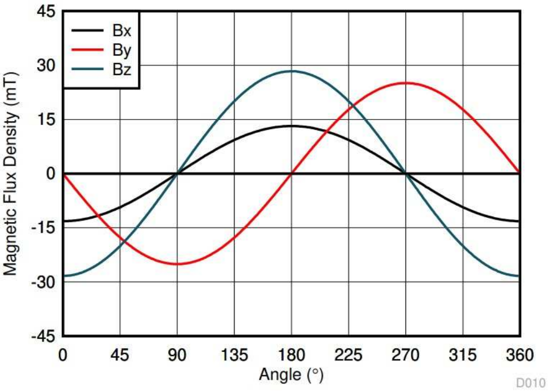


Fig. 1.15: 3D Hall-effect graph [32]

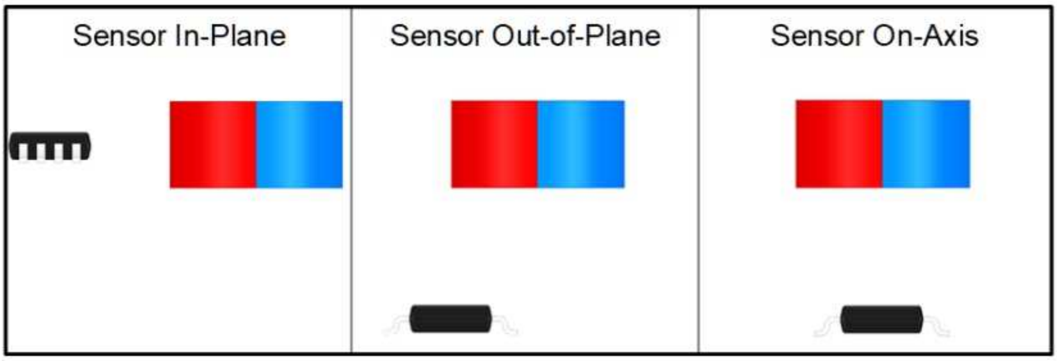


Fig. 1.16: 3D Hall-effect magnetic sensor - possible placements [32]

1.3 Comparison of rotary sensors

Potentiometers change their resistance based on the angle of a sliding contact on a resistive element. Output is usually 0 V - V_{supply}. Cheap potentiometers are not suitable for precise measurement, more expensive industrial ones can be very precise and have some environmental protection. They need external hardware to convert their output analog signal to information about angle. The problem with potentiometers is that they tend to wear out after extensive use and also have a limited measuring range.

Resolvers have two sets of windings in the stator. One produces a sine signal and the second a cosine signal. The signal is induced on them in different ratios depending on the position of the rotor. Output signals are analog and also need external electronics to calculate angle data. They are very durable and can withstand high temperatures, high doses of radiation, shocks, and vibrations.

Contact Encoders have brush sensors and coded discs and generate pulses based on conductive contact of brushes with metal disc segments. They have low resolution and they can wear out due to mechanical contact.

Optical Encoders work on the same basic principle of a disc with code, as contact encoders. The difference is that the disc is coded with transparent segments, a pair of LEDs and photodiodes read every bit. The absolute angle information is coded often with Gray code, to reduce false angle readings. They can have high resolution. Optical encoders are not as durable as their magnetic and capacitive counterparts

Magnetic Encoders have discs encoded with tracks of magnetic segments with alternating poles. They have smaller resolution than optical encoders but they are much more durable.

Capacitive Encoders read changes in the capacitance of the disc using high-frequency signals. They combine the resolution of optical encoders with the durability of magnetic encoders. The greatest advantage is their power effectiveness compared to optical and magnetic encoders.

Hall-effect sensors can read changes in the magnetic field of the magnet. For absolute angle measurement a 3D Hall-effect sensor is often used. This type includes three Hall-effect sensors, one for each axis (X, Y, Z). Output is composed of three phase-shifted sine waves that are for each angle in the same ratio. 3D Hall-effect sensor can be very precise. AEAT-9922, the sensor used in the practical part of this thesis has 18 bits and a resolution of one-thousandth of a degree per binary state.

2 Stepper Motors

Stepper motor is a special type of motor, that can be moved to a precise angle. Unlike other motors is unable to stop at a specific position. The movement of the stepper motor is controlled by electrical pulses. Each pulse represents a fixed incitement of the angular position of the motor shaft. Increment can be in any direction independently on the previous step. This type of motor by itself lacks feedback about the current angular position making it by itself an open-loop system. For most applications is needed to include a control system with a sensor for angular position measurement. Stepper motors are usually controlled by stepper motor drivers. [33]

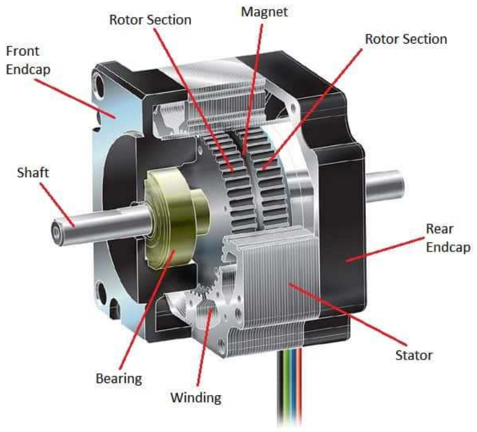


Fig. 2.1: Stepper motor parts [34]

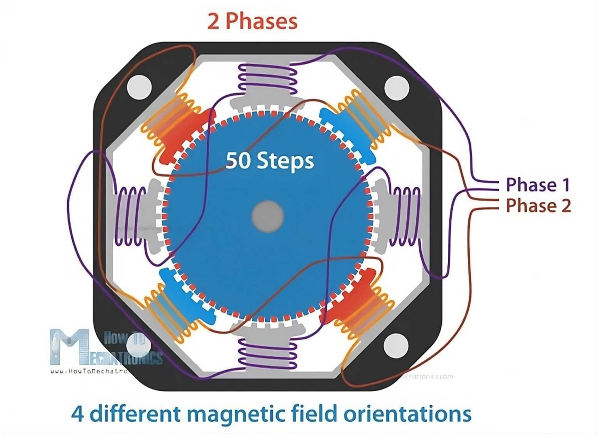


Fig. 2.2: Bipolar stepper motor [35]

- **Variable Reluctance (VR) Stepper Motors:** Simplest and cheapest type of stepper motor. This motor is made from an iron-toothed rotor and stator with paired stator windings. They have no detent torque (torque between steps that helps the motor to stay in position when not powered). [33]
- **Permanent Magnet (PM) Stepper Motors:** They consist of a permanent magnet rotor with no teeth and stator from windings. The rotor rotates due to the attraction of magnetic poles and can have more than one pair of poles. They exhibit high torques. [33]
- **Hybrid Stepper Motors:** They incorporate the qualities of both VR and PM designs. They have rotors from multi-toothed permanent magnets and electromagnets around their shafts. They have increased detent torque and smaller step angles. [33]

2.1 Microstepping

The usual number of steps per revolution is 200 but can be 400 (*it means that one step corresponds to 360°/200 steps = 1.8° per step for 200 and 0.9° for 400*). A number of steps can be significantly increased by a stepper motor driver feature called micro-stepping. If microstepping 1/32 is enabled on the driver, each step is divided into 32 microsteps. Steps are then increased from 200 steps per revolution to 200*32=6400 steps per revolution (360°/6400=0.05625° per step). Microsteps are produced by proportioning the current into two windings according to sine and cosine functions. This feature is widely used to increase resolution and smoother motion.

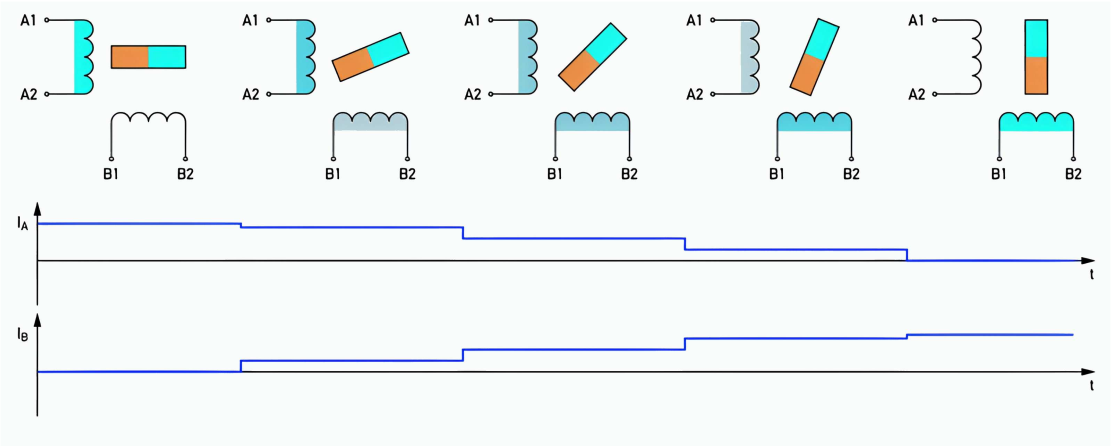


Fig. 2.3: Quarter microstepping on the bipolar stepper motor [36]

Older industrial stepper motor drivers featured only 4 micro-step setting, also called quarter-step mode. Modern drivers can have from 16 up to 256 micro-steps per full step. So with these values in theory we can get to a very satisfying resolution. In practice, the stepper motor is still an open-loop system meaning that it has no information about its current position and cannot correct step position errors. Friction, stepper motor step accuracy, external load, and nonlinearity of stepper motor driver can cause major deviations in motor movement. This error is also called step loss. Solving this problem requires closing the open-loop system with an absolute angular sensor. [36]

The table below shows the relationship between a number of micro-steps and torque per micro-steps. As microstep resolution is increased, incremental torque drastically drops. A consequence of this is if the motor friction, detent torque, and weight of load are greater than the incremental torque of the micro-step, the motor can become unmovable. [37]

Microsteps per full step	Holding Torque per micro-step	Angle per micro-step
1 (Full step)	100.00%	1.8°
2	70.71%	0.9°
4	38.27%	0.45°
8	19.51%	0.225°
16	9.80%	0.1125°
32	4.91%	0.05625°
64	2.45%	0.028125°
128	1.23%	0.0140625°
256	0.61%	0.00702125°

Table 2.1: Incremental torque per micro-step dropping as the number of micro steps per full step increases (torque 100% means maximal torque of full step on motor) [37]

In summary, microsteping reduces mechanical noise, makes rotation smoother, reduces vibrations, and mainly increases the resolution of individual steps. The downside is that with an increased microstepping setting torque per micro step decreases.

2.2 NEMA 17

NEMA 17 is a bipolar stepper motor with a step size of 1.8° or 0.9° available in single or dual-shaft versions. The driving voltage is 12-24 V. Maximum speed can be as high as 2000 rpm. Nema 17 is used in 3D printers, robots, CNC machines, and Telescopes. Accuracy of NEMA-17 is around ±0.2%. [38]

3 Explanation of Problem

The opto-mechanical platform FSO/RF Hybrid Link has to be precisely controlled in altitude and azimuth axes. This platform enables ±5° rotation in both axes. Motion is performed by two NEMA-17 stepper motors. For each axis, one hall-effect angle measurement sensor AEAT-9922 for closed-loop control needs to be implemented.

At first, the platform prototype will be controlled by the Arduino Uno platform with a DRV8825 stepper motor driver and AEAT-9922 sensor. Communication with AEAT-9922 sensors will be carried out by a 4-wire SPI interface. DRV8825 Drivers will utilize 1/32 micro-stepping. Later the final board for the opto-mechanical platform will be designed and integrated.

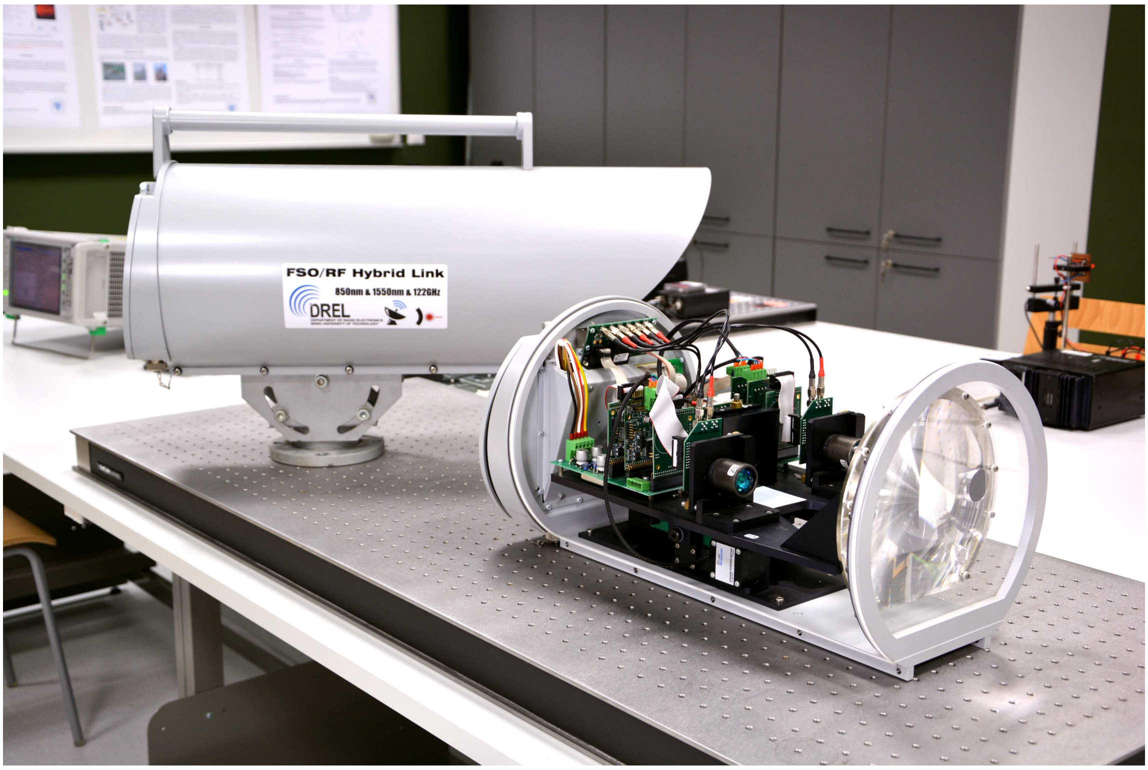


Fig. 3.1: Opto-mechanical platform

4 Thesis Results

4.1 Measuring angle with AEAT-9922

The AEAT-9922 from Broadcom company is an angular magnetic rotary sensor, that uses a 3D Hall effect sensor element with complex analog and digital signal processing within a single device, capable of precise absolute angular measurement. AEAT-9922 also includes incremental output. This sensor only requires one diametrically magnetized magnet. The sensor also supports both on-axis and off-axis modes. Absolute angle measurement supports 10-bit to 18-bit measurement resolution. The driver board will be used in 18-bit resolution mode. On this setting, the sensor can detect changes up to 0.0014°. Absolute position data can be read using SSI-2, SSI-3, SPI-3, and SPI-4 (with CRC and parity bit options) or with a PWM-encoded output signal. Incremental signals can be read with ABI and UVW signals with a wide user-configurable resolution. AEAT-9922 can operate with both 5 V and 3.3 V. It has a calibration mode and a user-programmable zero position. The sensor can operate in temperatures from -40°C to 125°C. [39]

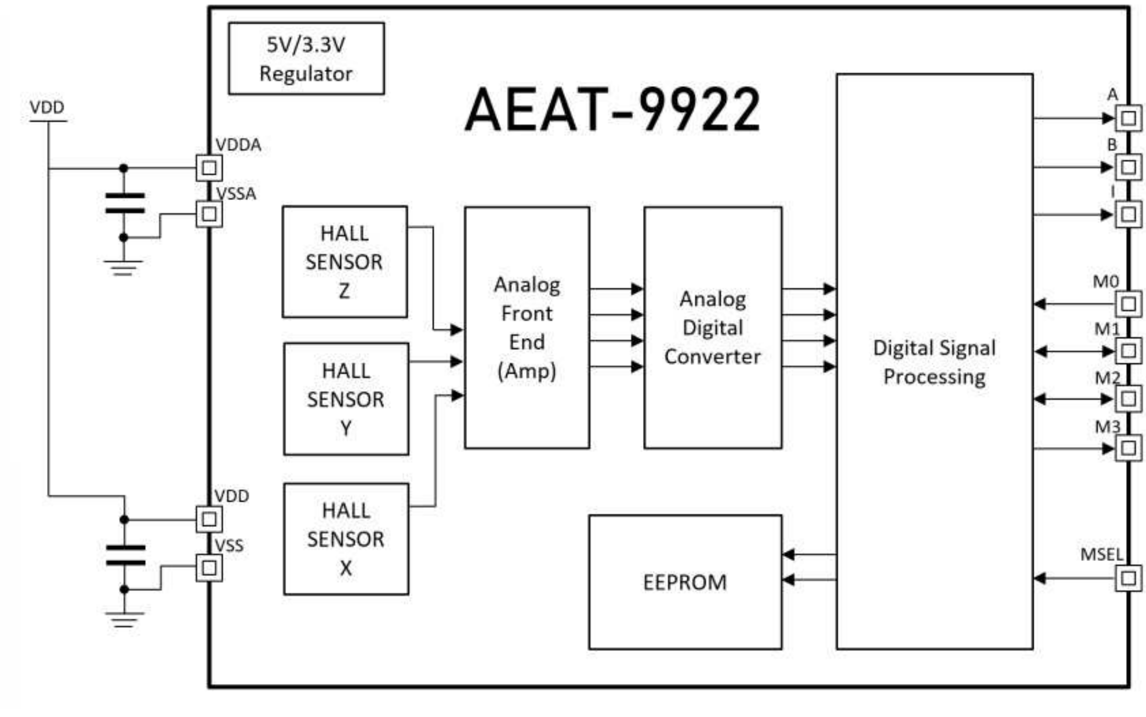
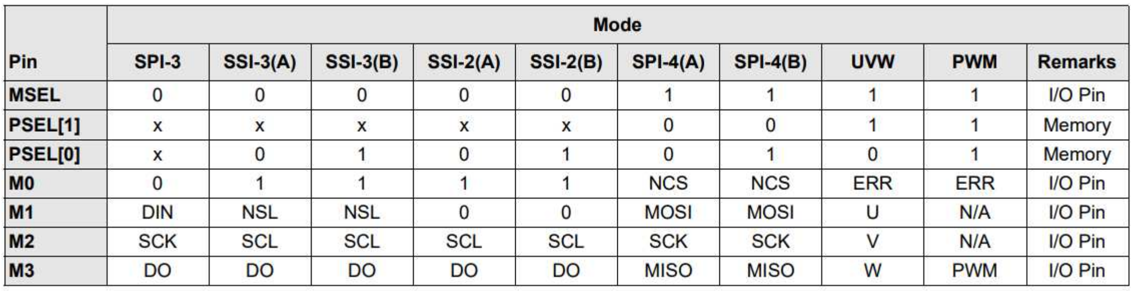


Fig. 4.1: AEAT-9922 Block diagram [40]

MSEL pin is used for changing output modes (*logical 0 is used for three and two-wire communication and also for both SPI-4, UVW, and PWM*). Further communication mode selection is adjustable by changing internal registers PSEL[0] and PSEL[1]. Pins M0 to M3 are used for data transfer. ABI has its own set of pins. [41] [40]



Pin	Mode									
	SPI-3	SSI-3(A)	SSI-3(B)	SSI-2(A)	SSI-2(B)	SPI-4(A)	SPI-4(B)	UVW	PWM	Remarks
MSEL	0	0	0	0	0	1	1	1	1	I/O Pin
PSEL[1]	x	x	x	x	x	0	0	1	1	Memory
PSEL[0]	x	0	1	0	1	0	1	0	1	Memory
M0	0	1	1	1	1	NCS	NCS	ERR	ERR	I/O Pin
M1	DIN	NSL	NSL	0	0	MOSI	MOSI	U	N/A	I/O Pin
M2	SCK	SCL	SCL	SCL	SCL	SCK	SCK	V	N/A	I/O Pin
M3	DO	DO	DO	DO	DO	MISO	MISO	W	PWM	I/O Pin

Fig. 4.2: Communication mode selection table [40]

AEAT-9922-SPI4(A) communication

(A) means with parity bit, (B) is with cyclic redundancy check (CRC) In this communication mode pins are assigned as follows:

- **M0**-NCS (Negate Chip Select)
- **M1**-MOSI (Master out Slave in)
- **M2**-SCK (Clock)
- **M3**-MISO (Master in Slave out)

AEAT-9922 SPI-4 uses CPOL = 0 and CPHA = 1 for triggering which corresponds with SPI-4 Mode 1.

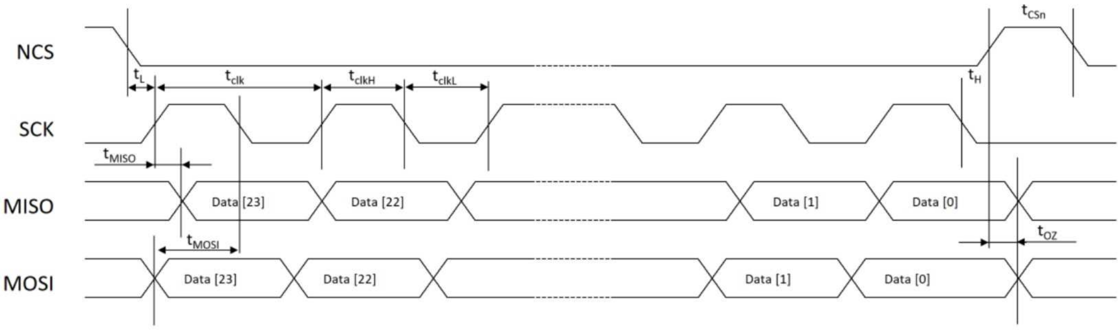
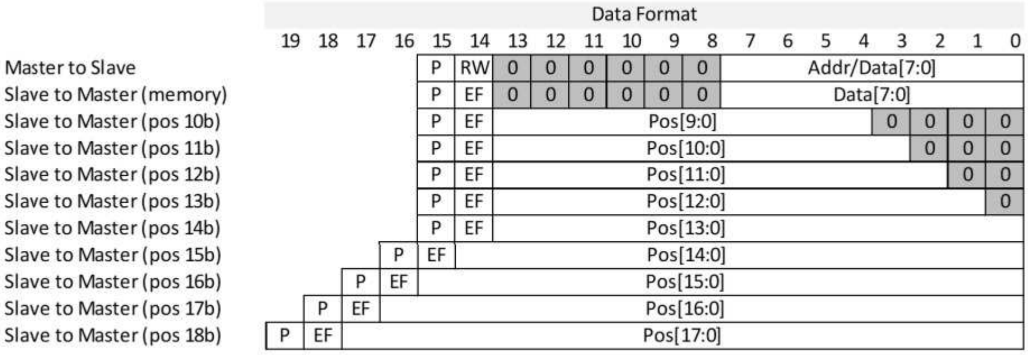


Fig. 4.3: SPI-4 communication [40]

Slave to master message is composed of a bit for odd parity, read/write bit, 6 zeroes, and 8 bits that can be used to call to register address or send some data to that register.



	Data Format																			
	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Master to Slave					P	RW	0	0	0	0	0	0	Addr/Data[7:0]							
Slave to Master (memory)					P	EF	0	0	0	0	0	0	Data[7:0]							
Slave to Master (pos 10b)					P	EF	Pos[9:0]										0	0	0	0
Slave to Master (pos 11b)					P	EF	Pos[10:0]											0	0	0
Slave to Master (pos 12b)					P	EF	Pos[11:0]												0	0
Slave to Master (pos 13b)					P	EF	Pos[12:0]													0
Slave to Master (pos 14b)					P	EF	Pos[13:0]													
Slave to Master (pos 15b)				P	EF	Pos[14:0]														
Slave to Master (pos 16b)			P	EF	Pos[15:0]															
Slave to Master (pos 17b)		P	EF	Pos[16:0]																
Slave to Master (pos 18b)	P	EF	Pos[17:0]																	

Fig. 4.4: SPI-4 data format [40]

Note: P-parity bit, EF-error flag, RW: Read = 1, Write = 0

Reading of absolute position Data

In the figure below is communication with the sensor through SPI-4 communication protocol with the parity bit.

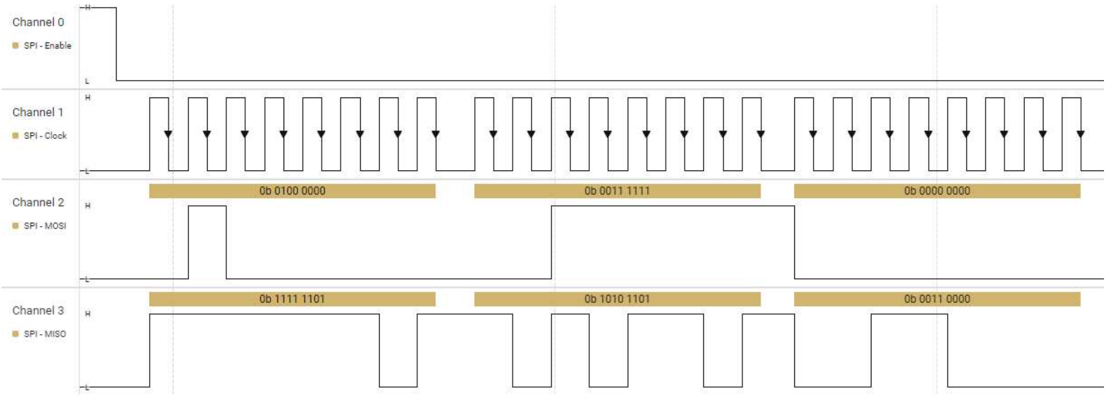


Fig. 4.5: Absolute position data

MOSI (Master out slave in)

The first bit sent is an odd parity bit (*in this case 0*). The second bit is the read/write bit (*1 - to read*). The rest of the first byte are zeroes. The second byte corresponds to the register address. Absolute position data are stored in a register with address 0x3F (*0b00111111*). The third byte is full of zeroes and it is used to read the last 4 bits of 18-bit positional data.

MISO (Master in slave out)

The first two bits of the first byte are parity and error flag. The next 18 bits correspond to the current position. The last 4 bits are empty. In this case, the position of the shaft can be calculated from the 18 positional bits. Position *0b111101101011010011*, which corresponds to 252627 in decimal out of the maximal 262144 (2^{18}). This maximal value is 360°. [40]
The motor shaft's current position can be calculated from value 252627 as:

$$\frac{360^\circ}{262144} \times 252627 = 346.9304^\circ$$

AEAT-9922 Important registers

AEAT-9922 sensor has a lot of settings. They are for zero position calibration, accuracy calibration, hysteresis, communication modes, absolute and incremental resolution, magnet position, and the direction of rotation (*counting up or down on clockwise rotation*). In the table below are the most important ones.

Address	Name	Settings
0x07	Customer config 0 MSB	Accuracy calibration, Axis mode, ABI settings
0x08	Customer config 0 LSB	Hysteresis, Direction, Absolute Resolution 10-18b
0x0B	Customer config 2	PSEL[0], PSEL[1], UVW, PWM
0x10	Register unlock	Write: 0xAB to unlock all registers
0x11	Register program	Write: 0xA1 to program all registers to EEPROM
0x21	Error bits	Chip ready, magnet high/low, memory error
0x12	Calibration	Write: 0x02 to start calibration sequence
		Write: 0x08 to set new zero position
		Write: 0x01 to erase calibration
		Write: 0x04 to erase zero position
		Write: 0x00 to return to operation mode
0x22	Calibration state	Write 0x00 to end calibration/zero position calibration
0x3F	Absolute position	Data can be obtained by sending read command

Table 4.1: Important registers [41]

4.2 Control of NEMA-17 step motor with DRV8825

DRV8825 is a stepper motor driver. This device has two H-bridge drivers and microstepping indexer. This driver is for control of bipolar stepper motors. It can withstand up to 45 V and 2.5 A. It is capable of 1/32 microstepping mode. [42]

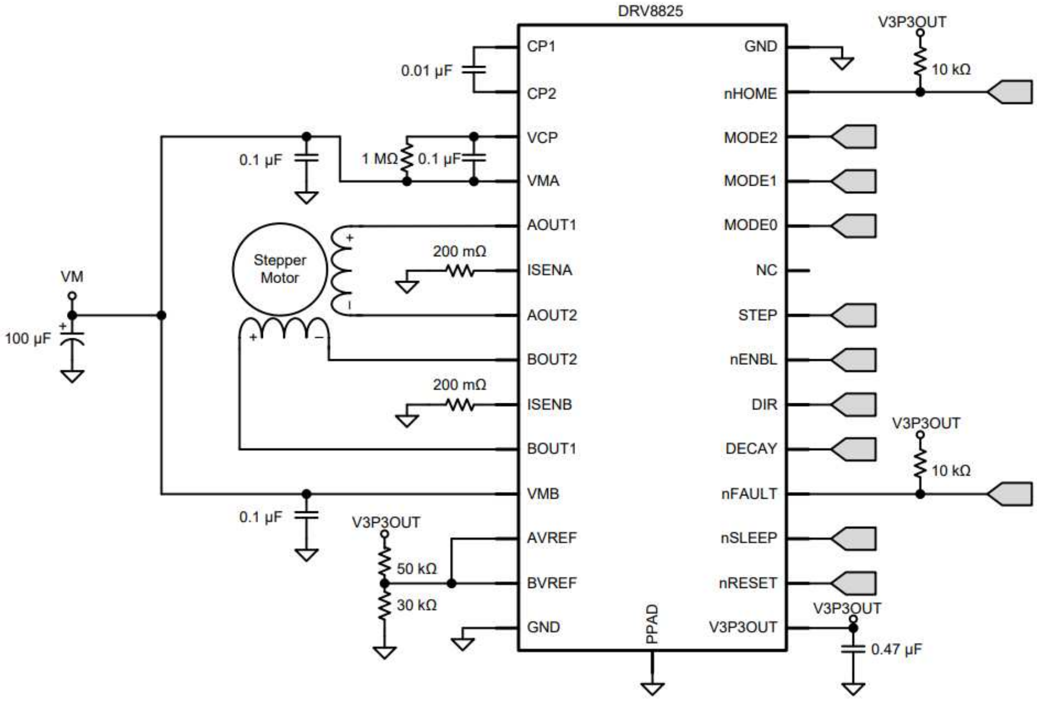


Fig. 4.6: Typical application of DRV8825 [42]

This integrated circuit is controlled by STEP and DIR pins. Every time the square pulse arrives at the STEP pin the driver turns the stepper motor by one step. With the NEMA-17 stepper motor with 1.8° rotation per step, it is 200 steps per full rotation and with 1/32 microstepping setting, it is 6400 steps per full revolution. This means that the maximal resolution of the system with this stepper motor and driver is 0.05625° per step. For better resolution of the system, a better driver with more delicate microstepping is needed. DIR pin controls the direction of steps. The DRV8825 is fully protected against undervoltage, overcurrent, and overtemperature. Pin nSLEEP can be used in battery-powered applications. [42]

4.2.1 Microsteping modes

Microstepping modes can be set by setting M0, M1, and M2 to specific combinations of logic inputs. For the 1/32 microstepping setting, they need to be put in a high logic state.

MODE0	MODE1	MODE2	Micro-step Resolution
Low	Low	Low	Full step
High	Low	Low	Half step
Low	High	Low	1/4 step
High	High	Low	1/8 step
Low	Low	High	1/16 step
High	High	High	1/32 step

Table 4.2: Microstepping setup [42]

4.2.2 Connection of DRV8825

Nema-17 step motor should have a voltage of 12 V, which must be supplied separately from the external source and connected to VMOT. To protect the board against voltage spikes that could damage the driver it is advised to use at least a 47 µF electrolytic capacitor between VMOT and the ground somewhere close to the board. In this case, a 100 µF 50 V electrolytic capacitor is being used. Pins nRESET and nSLEEP are not used, so they are permanently in inactive state log 1. Pins A1, A2, B1, and B2 are used for the connection of the stepper motor with B1, B2 for first winding and A1, A2 for the second winding. To protect the driver from overheating heatsink is installed on top of the chip. [42]

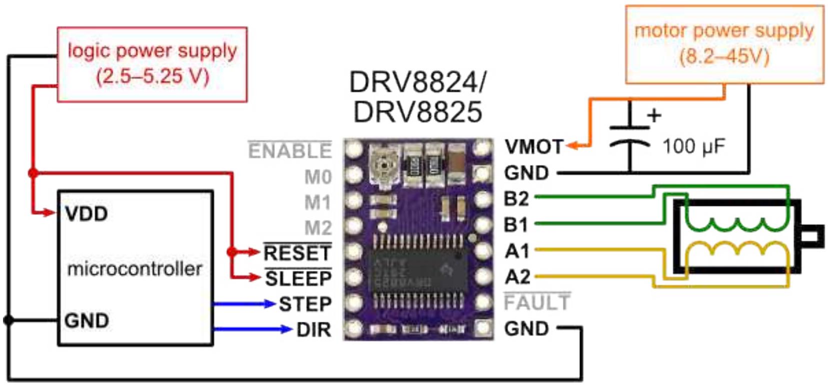


Fig. 4.7: DRV8825 connection [43]

4.3 Implementation on one axis prototype

The prototype of a closed-loop control system for one-axis movement. The prototype is built from a DRV8825 driver for control of the NEMA-17 stepper motor and AEAT-9922 rotational sensor. The whole system is controlled by the Arduino UNO R3 platform.

4.3.1 Breadboard prototype

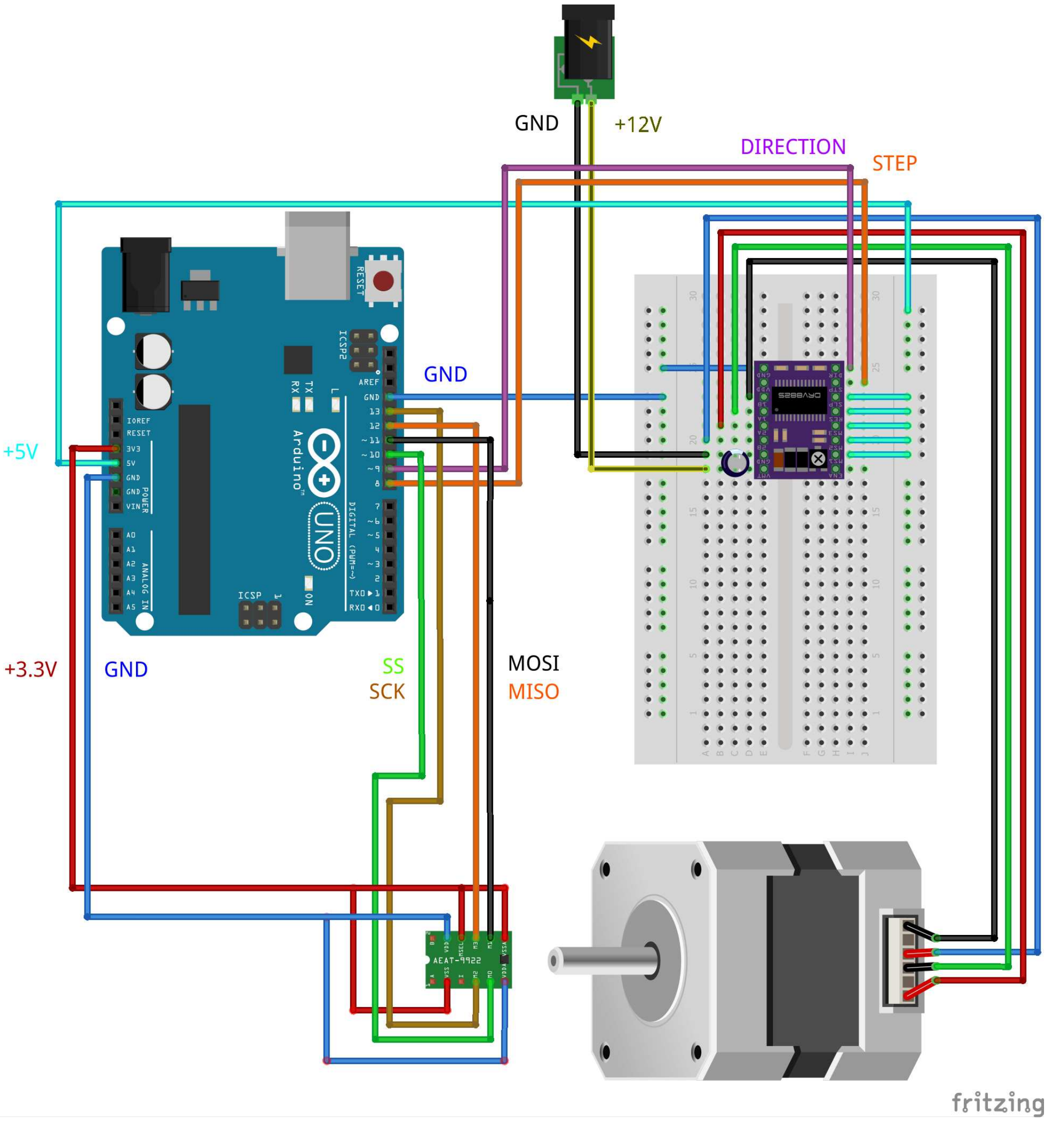


Fig. 4.8: Prototype of the driver (on the motor shaft is a diametrical magnet, green IC next to the motor shaft is AEAT-9922 sensor)

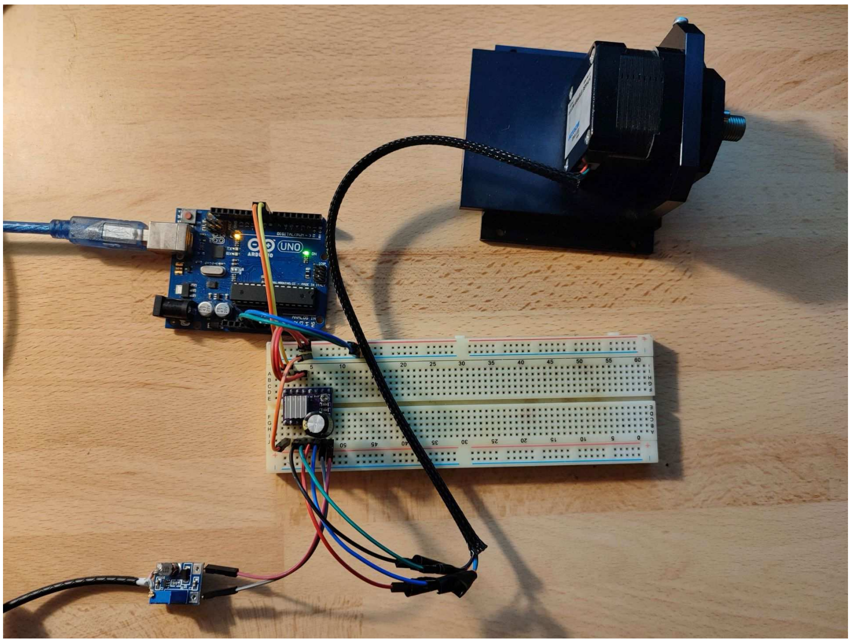


Fig. 4.9: DRV8825 connected with step motor and Arduino platform

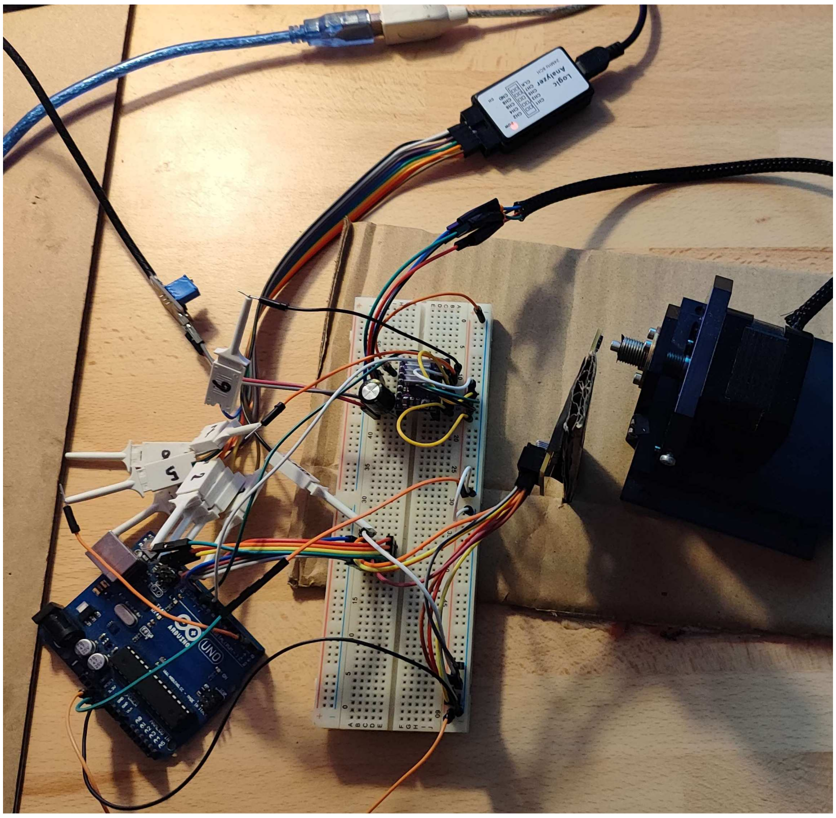


Fig. 4.10: One-axis prototype of the driver

4.3.2 Tests and Evaluation

In this test, the first angle of the motor shaft was 135.24°. The desired final angle was 340.5°. The driver did approximately 3603 steps. The last 3 values were in "check the position" mode. The final measured angle was 340.52°. The position where the stepper motor ended was with a deviation of 0.0254°, which is smaller than the size of one step. The size of one step was in this case 0.05625°. Higher accuracy could be achieved only with a driver with a finer microstepping setting.
The nonlinearity of rotation that can be seen on the second graph was probably caused by friction of the platform in which the stepper motor is placed and the nonlinearity of the DRV8825 driver. This nonlinearity of the driver also caused the driver to do about 46 fewer steps than it theoretically should. The average micro step on 1/32 microsteping setting on this driver was 0.056975° which is 0.000725° error per step on average.

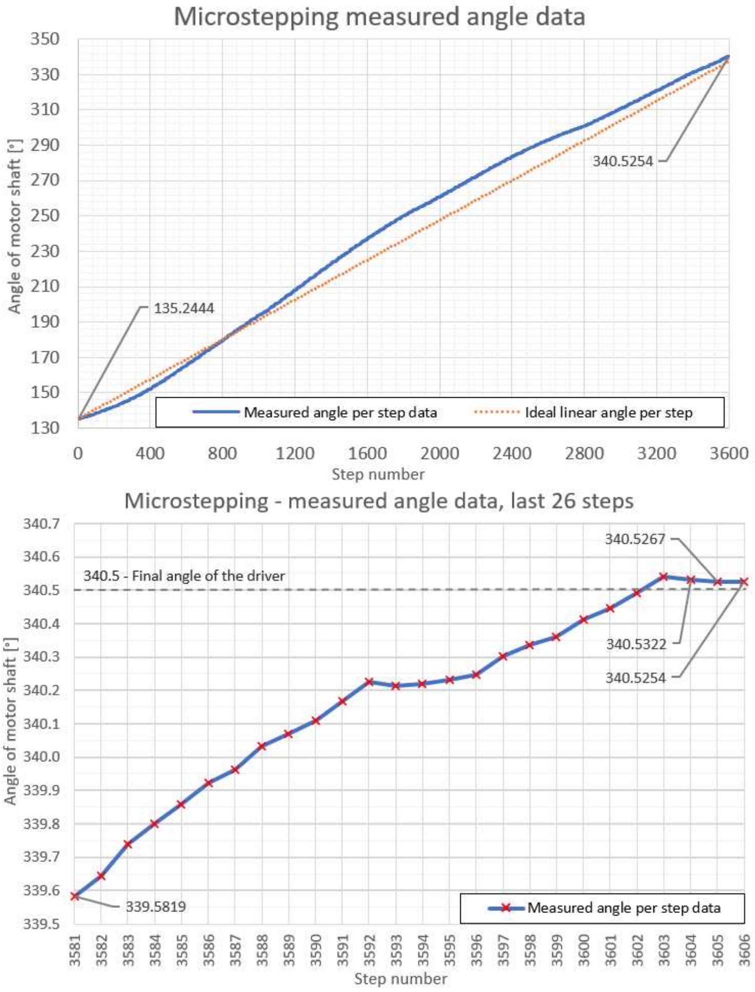


Fig. 4.11: Graph from test of driver prototype

4.3.3 Flowchart of the prototype driver program

The program for the prototype driver works as follows. At first, the program calculates the final position in steps from zero position from the user-inputted variable by dividing it by the size of one step 0.05625°. The current angle is recalculated the same way. Both of them are compared at each program cycle. The result of comparison gives the program direction in clockwise or counterclockwise. Then program sends a step signal to the DRV8825 stepper motor driver which moves the motor by one step. This continues until the target step and current step are the same. Then the program does this cycle every 10 s to verify if equality is still true to stay at the same position.

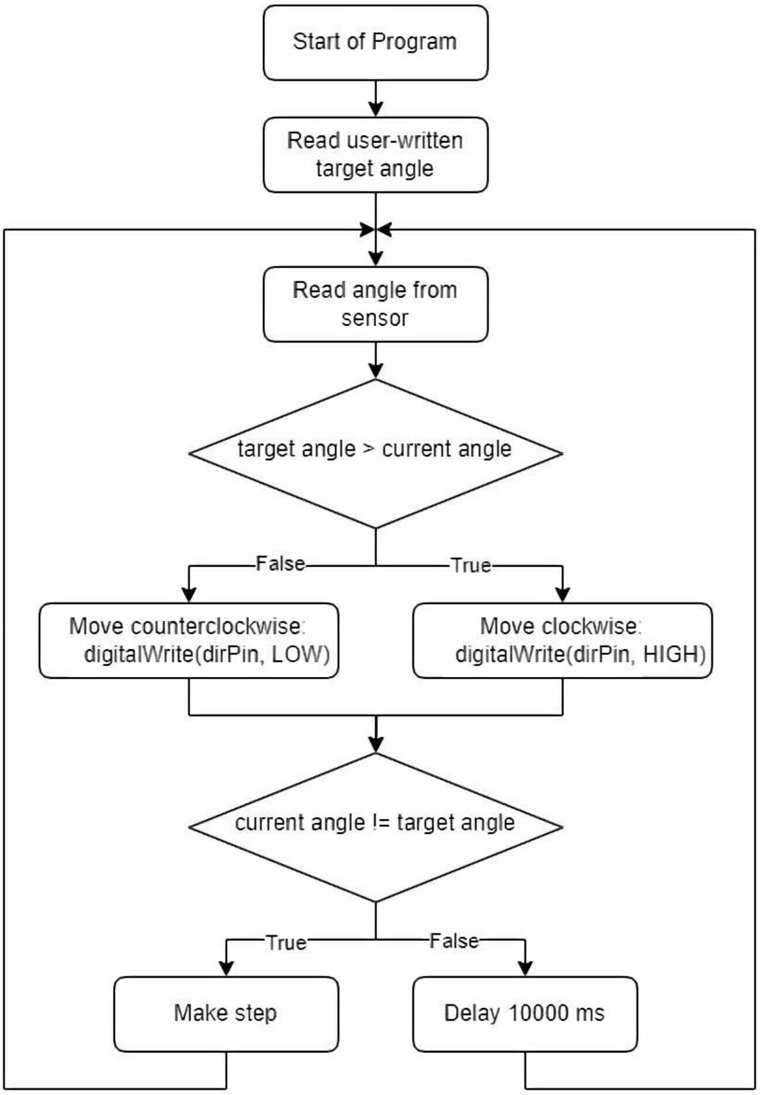

Start of Program
Read user-written target angle
Read angle from sensor
target angle > current angle
False
True
Move counterclockwise: digitalWrite(dirPin, LOW)
Move clockwise: digitalWrite(dirPin, HIGH)
current angle != target angle
True
False
Make step
Delay 10000 ms

Fig. 4.12: Prototype driver board program flowchart

4.3.4 Accuracy of the opto-mechanical platform

The accuracy of the laser beam hitting the target at some distance is mainly dependent on the size of the step on the Nema-17 stepper motor and microstepping resolution of the motor driver. The current driver prototype has a maximum of 32 microsteps per step and the stepper motor has 1.8° per step. The final platform driver will use 128 microsteps per step. The accuracy of the laser on target distance can be calculated using trigonometry.

$$\tan(\Phi) = \frac{\Delta[\mathrm{m}]}{\text{Distance [m]}}$$

$$\Delta[\mathrm{m}] = \tan(\Phi) \times \text{Distance [m]} = \tan(0.05625°) \times 1000 \text{ m} = 0.982\,\mathrm{m}$$

Note: Δ is a deviation in meters, per one micro step, Φ is the angle change per one micro step

The current accuracy of the prototype driver with 1/32 microstepping and 1.8° step size on NEMA-17 is 98.2 cm per micro-step per 1 km. The accuracy of the final platform will be with around 24.5 cm per step at 1 km using 1/128 microstepping. Even better accuracy of 12.3 cm per step at 1 km or 61.4 cm per step at 5 km can be achieved if stepper motors were changed for NEMA-17 with 0.9° per step.

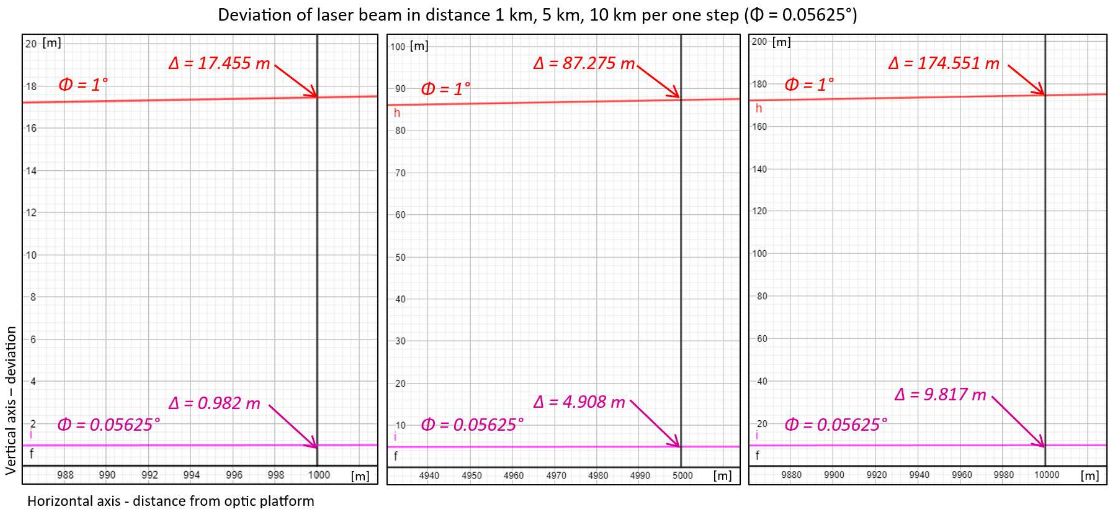


Fig. 4.13: Deviation Δ of the laser beam in distances 1 km, 5 km, and 10 km on a current prototype for micro-step size 0.05625°. For comparison, red beam is a theoretical beam with a step size of 1°.

Micro step mode	Angle of step	Deviation/step at 1 km	Deviation/step at 5 km
1/32 (1.8° step)	$\Phi = 0.05625°$	$\Delta_{1\ \mathrm{km}} = 0.982\,\mathrm{m}$	$\Delta_{5\ \mathrm{km}} = 4.908\,\mathrm{m}$
1/64 (1.8° step)	$\Phi = 0.02813°$	$\Delta_{1\ \mathrm{km}} = 0.491\,\mathrm{m}$	$\Delta_{5\ \mathrm{km}} = 2.454\,\mathrm{m}$
1/128 (1.8° step)	$\Phi = 0.0140625°$	$\Delta_{1\ \mathrm{km}} = 0.245\,\mathrm{m}$	$\Delta_{5\ \mathrm{km}} = 1.227\,\mathrm{m}$
1/128 (0.9° step)	$\Phi = 0.00703125°$	$\Delta_{1\ \mathrm{km}} = 0.123\,\mathrm{m}$	$\Delta_{5\ \mathrm{km}} = 0.614\,\mathrm{m}$

Table 4.3: Deviations on different distances with different parameters

4.4 Design of the final driver board

The design of the new stepper motor driver board is almost the same as the old version, with the addition of SPI-4 output for the AEAT-9922 sensor used in the previous Arduino prototype.

Both versions of the final driver board use an STM32F303K6T6 microcontroller and L6470H stepper motor drive. The address of the driver board is stored in 24LC64SN EEPROM memory. Commands from both the PC and the control board already implemented in the opto-mechanical platform are transmitted with the use of industrial-grade bus RS-485. This signal is converted into the USART with the use of an onboard MAC483ECSA chip. Pinout for SWD (Serial Wire Debug) is implemented also for programming and debugging. Pinout for two optical gates measuring end positions on the azimuth axis.

Logic value for AEAT-9922 sensor, STM32F303K6T6, and L6470H is 3.3 V. 5 V is supplied to the board and converted to 3.3 V with the use of low-dropout voltage regulator LP38693SD-3.3. MAC483ECSA chip for USART/RS-485 uses 5 V. The Stepper motor uses 12 V. One control board controls only one stepper motor, so two are needed for full opto-mechanical platform control.

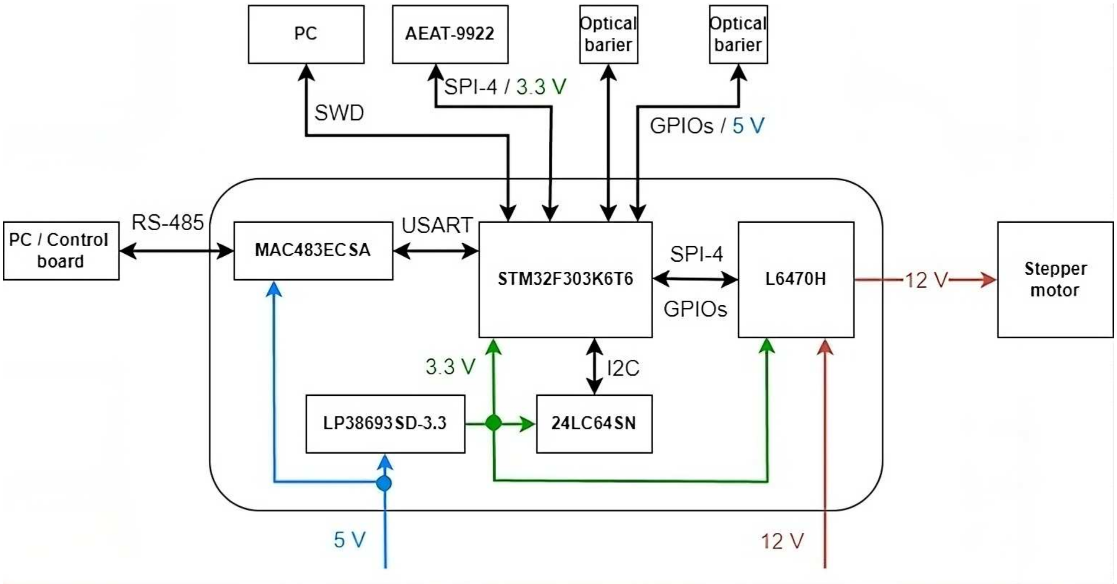


Fig. 4.14: Driver board block diagram

The board schematic and board layout can be found in the Appendix.

4.4.1 Microcontroller STM32F303K6T6

Microprocessor STM32F303K6T6 from company STMicroelectronics will be used in the new driver board. It incorporates the high-performance Arm® Cortex®-M4 32-bit CPU operating at up to 72 MHz frequency, up to 32 KB Flash, 16 KB SRAM, 2 ADCs, 3 DACs and 1 op-amp. STM32F303K6 is in a 32-pin package. It also supports 1 SPI, 1 I^2C, 2 USART, and 1 CAN bus, and also 26 GPIOs. [44]

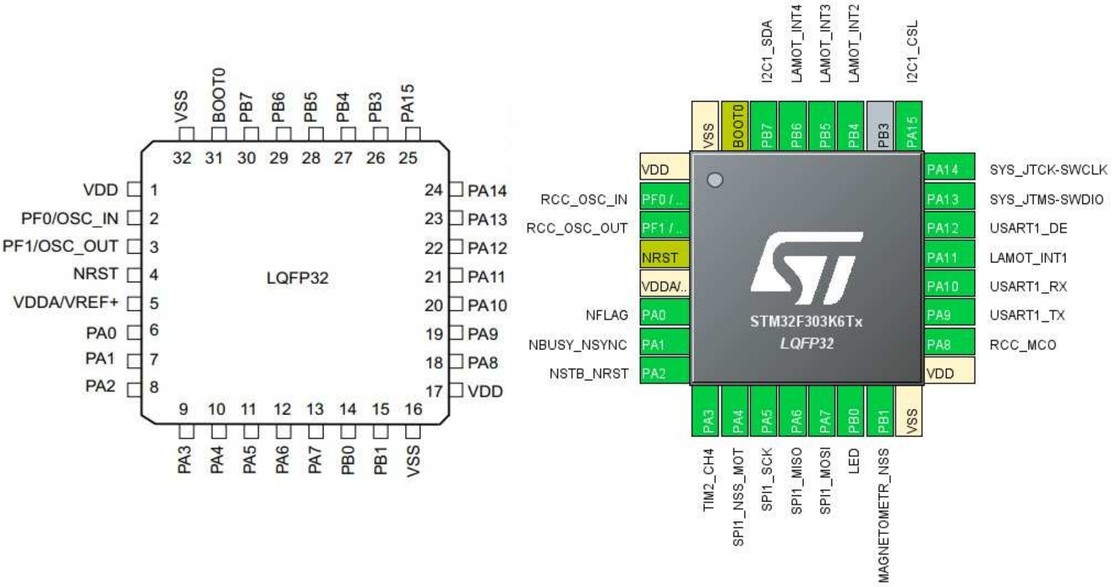


Fig. 4.15: STM32F303K6 pinout [45] and pin connection in this project (from the program STM32CubeIDE)

STM32F303K6T6 is connected to an external 16 MHz oscillator on pins 2 and 3. Then RS485 to USART converter MAX483ECSA on pins 19, 20, and 22. EEPROM memory 24LC64SN is programmed with the use of I2C and stores the address of the motor driver board. It is connected with pins 25 and 30. Motor driver L6470 is connected with pins 18, 9, 8, 7 and 6. Pins 23, 24, 4 and 26 are used for debugging and programming with ST-LINK V2. Pins 29, 28, 27 and 21 are connected to optical sensors at both ends of the axis. Pins 10, 11, 12, 13 and 15 are for SPI, both for AEAT-9922 and L6470. Pin 14 controls LED. The rest of the pins are for VSS and VDD.

4.4.2 Stepper motor driver L6470H

A motor driver L6470H, also from the company STMicroelectronics will be used. L6470H is suitable for driving two-phase bipolar stepper motors with microstepping. It has a true 1/128 steps resolution corresponding to 0.0141°, which is 4 times smaller than 1/32 microstepping. This driver has a very rich set of protections (thermal, low bus voltage, overcurrent, motor stall). An SPI-4 interface drives this motor driver. [46] [47]

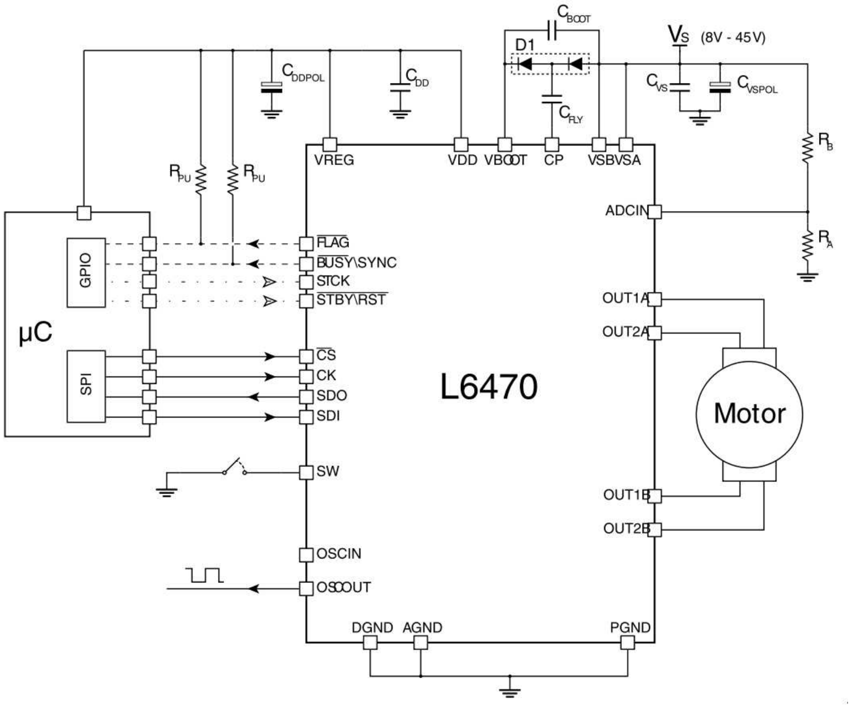


Fig. 4.16: Typical bipolar stepper motor control using L6470H [47]

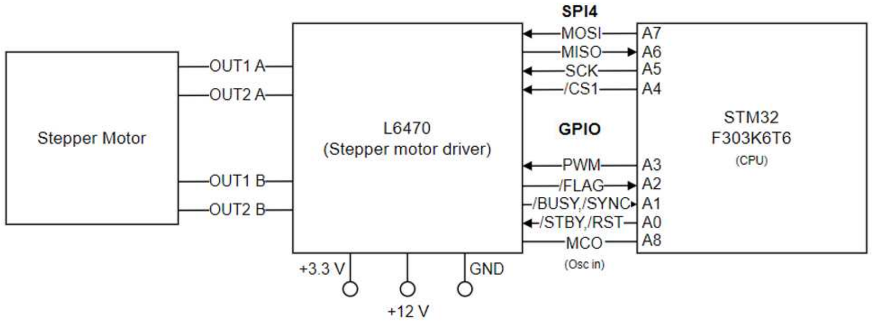


Fig. 4.17: L6470H connection to STM32F303K6T6

4.5 Finished driver board

The two driver boards were soldered and functionality was tested. Both of them are fully functional.

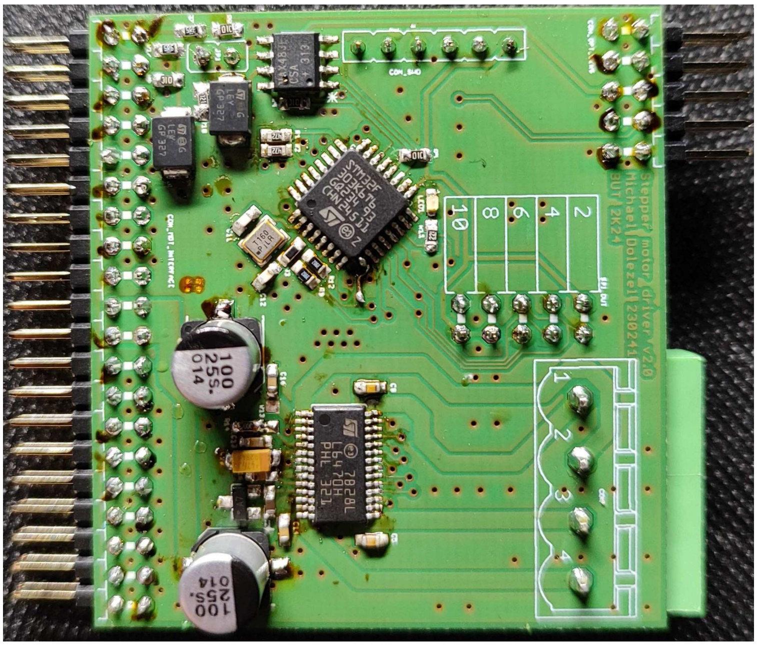


Fig. 4.18: Top side of finished driver board

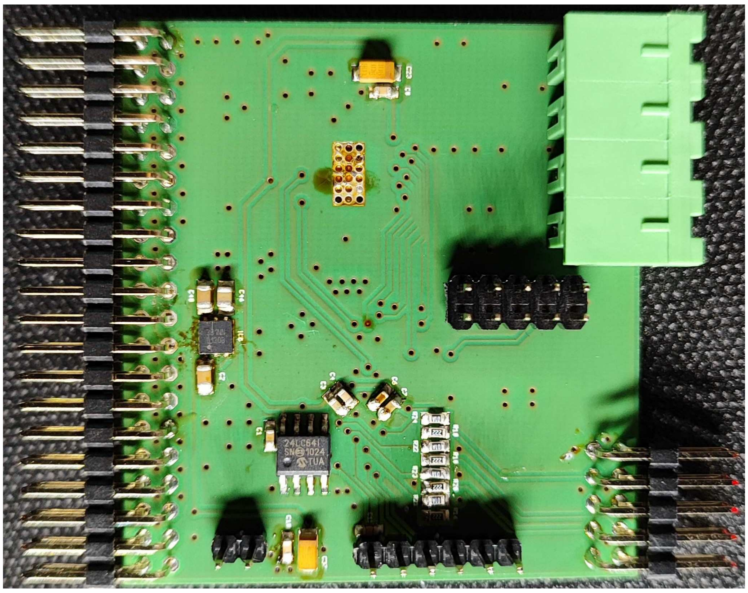


Fig. 4.19: Bottom side of finished driver board

4.6 Implementation to platform

Driver board can be implemented into an existing opto-mechanical platform board or can be used standalone with PC and USB to RS-485 convertor.

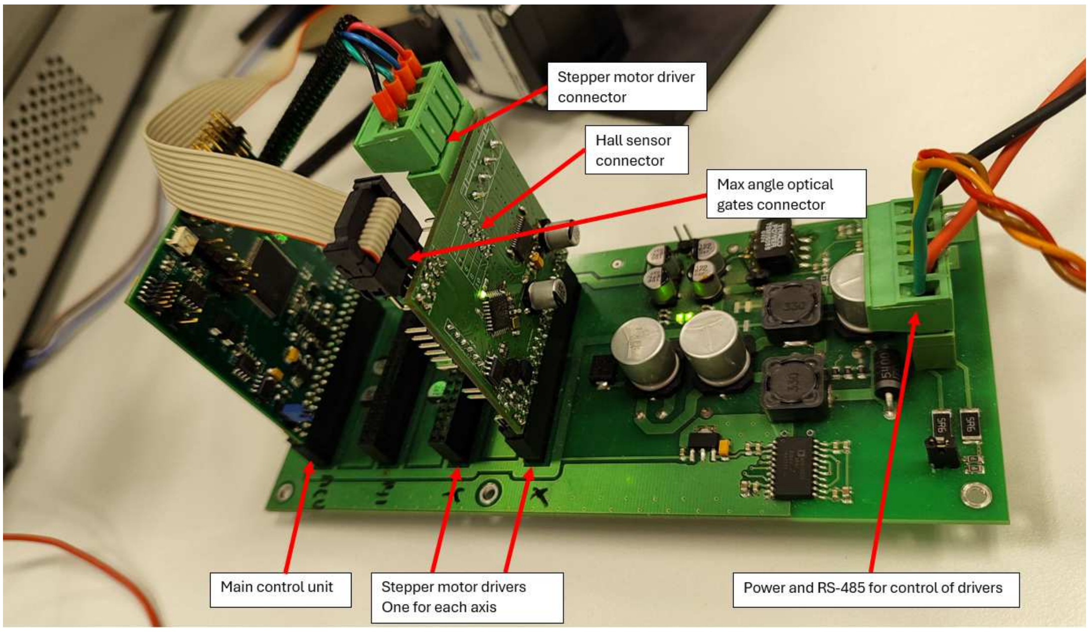

Stepper motor driver connector
Hall sensor connector
Max angle optical gates connector
Main control unit
Stepper motor drivers One for each axis
Power and RS-485 for control of drivers

Fig. 4.20: Opto-mechanical platform implementation

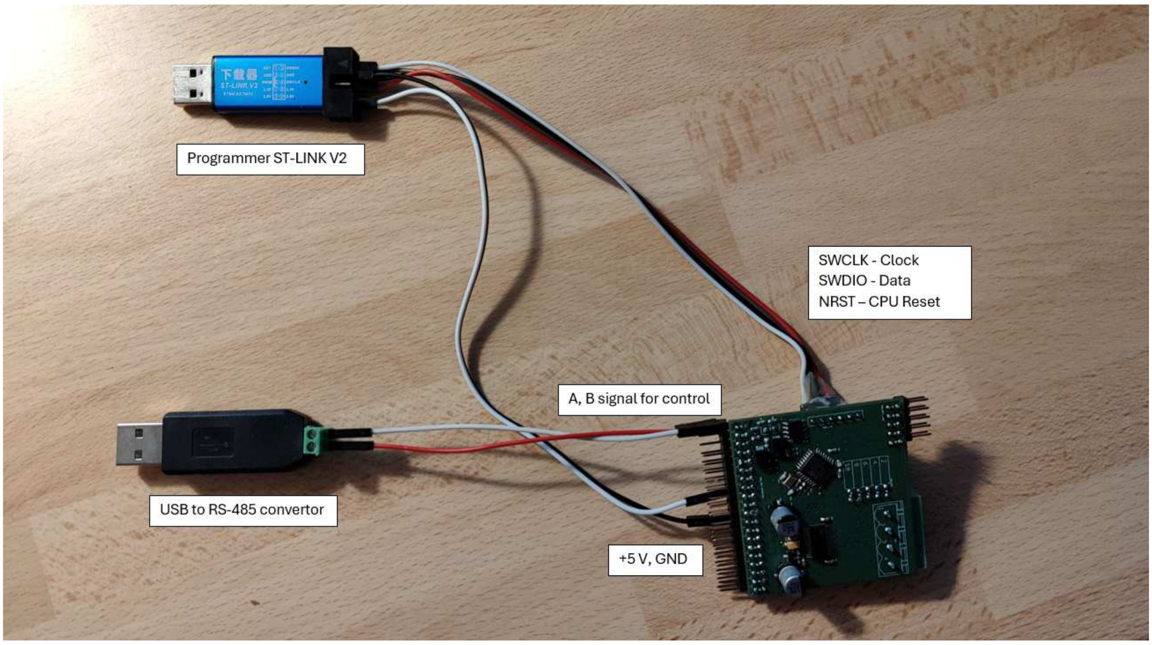


Fig. 4.21: Standalone implementation

4.6.1 Programming of driver board

The driver board is programmed with ST-LINK V2 with SWD (Serial Wire Debug) interface through the STM32CubeIDE program from STMicroelectronics in C language with the help of HAL (Hardware Abstraction Layer). Reading and graphic display of important values from the board is done with the help of STM32CubeMonitor. Control of the driver board manages the USB to RS-485 converter. The creation of correct control data packages is managed by a Python script.

4.6.2 Communication between driver board and central control board or PC

Commands for the driver board are sent with the RS-485 communication protocol. For communication with a PC, the USB to RS485 converter is used. The driver board has a MAX483ECSA integrated circuit that converts RS-485 commands into USART. USART is then transferred into the main uPC.

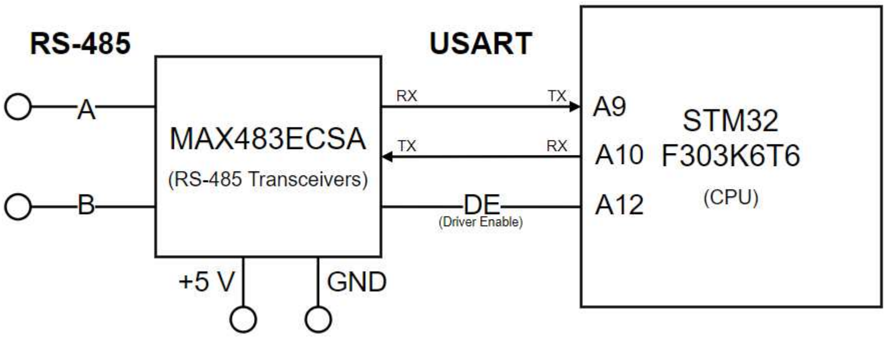


Fig. 4.22: Connection from control device into CPU

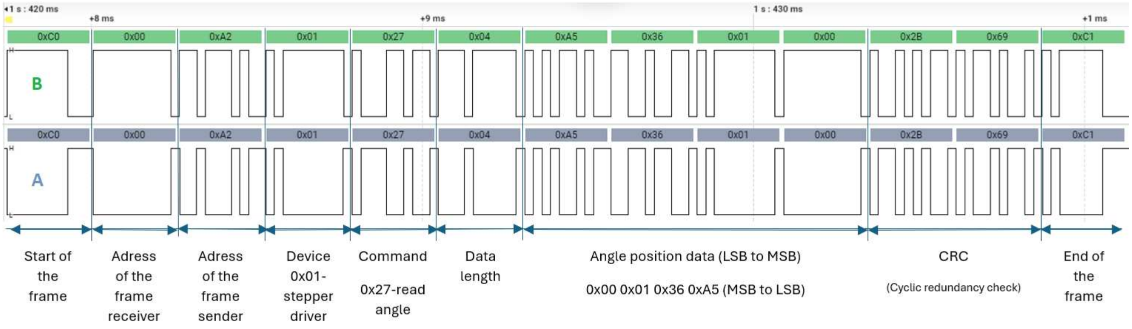


Fig. 4.23: RS-485 read angle incoming data frame

RS-485 Protocol

RS-485 is an asynchronous serial communication industrial standard. It supports half-duplex double-wire communication. It is effective for communication over long distances: over a kilometer even in electrically noisy environments. For full-duplex communication, 4 cables can be used. [48]

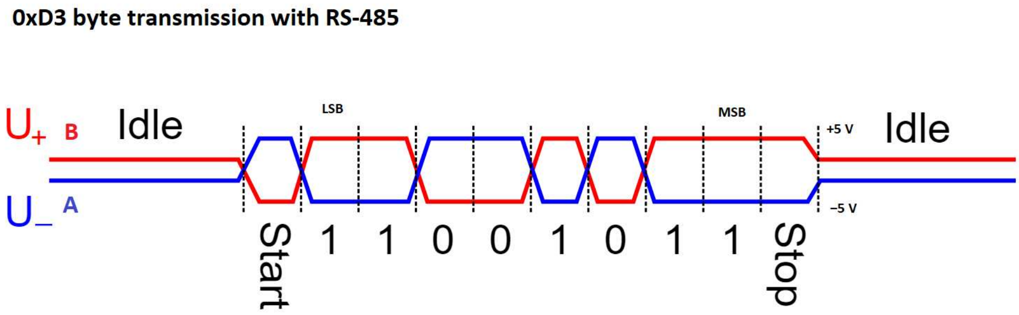


Fig. 4.24: RS-485 protocol [48]

Logical 0 is transmitted with B wire at Low and A wire at High. Logical 1 is transmitted with B wire at High and A wire at Low.

4.7 Driver board program

The driver program was already partially made with functional core elements, such as reading from the AEAT-9922 sensor, reading and writing to EEPROM, control of L6470H, and reading and sending of USART commands. Regulation for going to specific angles was partly implemented but mostly nonfunctional and needed to be remade. Reading from AEAT-9922 was simplified and now works in the same way as in the previous Arduino prototype.

The core of Python code for the creation and sending of RS-485 data packages from PC also existed. For this program, the graphical user interface with the use of the Tkinter library was created.

Driver board program's main tasks are to do first initialization, then to find out if the motor is moving, and next to check if the motor is outside of optical barriers limits. The program then waits for the incoming RS-485 data frame, when the data frame comes program can check CRC (CRC was also already implemented, is currently off, and ready for future use). It separates the two most important parts from the incoming data frame "**code2**" and "**data**".

Code2 includes commands like moving a specific number of steps or going to a specific position. The commands are:

Listing 4.1: Usable commands of driver

```
CMD2_LAMOT2_FORWARD                  0x21
CMD2_LAMOT2_BACKWARD                 0x22
CMD2_LAMOT2_STOP                     0x23

CMD2_LAMOT2_MOVE_RELATIVE_ANGLE      0x25
CMD2_LAMOT2_GO_TO_ANGLE              0x26
CMD2_LAMOT2_READ_ANGLE               0x27
```

Data part of RS-485 data frame includes a specific number of steps (1000 = 1000 steps for FORWARD and BACKWARD) or going to a specific angle (180325 = 180.325° for GO TO ANGLE and MOVE RELATIVE ANGLE).
AEAT-9922 angle reading, filtration of angle values, move commands, and RS-485 data frames will be described in the next few chapters.

4.7.1 Reading of values from AEAT-9922

Values from the AEAT-9922 sensor are read the same way as in the previous Arduino prototype as described in chapter 4.1.

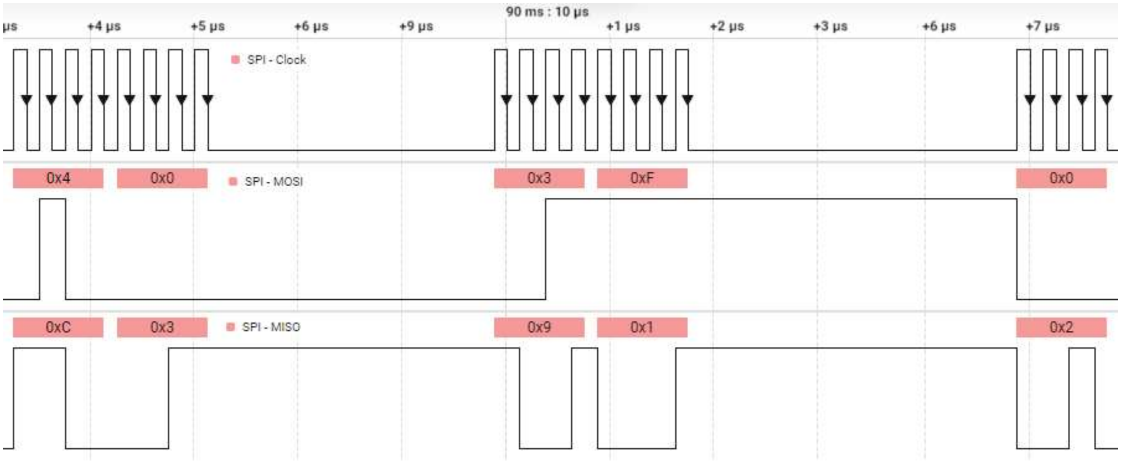


Fig. 4.25: AEAT-9922 SPI-4 communication

In the beginning, two variables are created (**data tx**, **data rx**) with a length of 20 bits each that will be used for sending and transmitting data. Variable **Value** is for storage of angle data. **Req** (0x403F) 4-0100 The first bit is parity and the second contains information that we want to read from a register with address "3F" that stores position data. Then those 16 bites are transmitted and add 4 more zero bits to fill the clock for our incoming 20 bites. Next, the chip select for AEAT-9922 is switched to low, and read angle data are sent. Angle data are received from the sensor at the same time, the incoming data are then saved into a **value** variable. Then the first two bites (error flag and parity) are masked and only angle data stays. Angles are in special data format (0 - means 0° and $2^{18} = 262144$ - means 360°). For conversion, data are multiplied by 36000 and then divided by 2^{18}. Finally, the output value resembles angle data ($180325 = 180.325°$).

Listing 4.2: Program for reading AEAT-9922 angle data

```c
uint32_t aeat9922_read(void)
{
    uint8_t data_tx[20] = {0};
    uint8_t data_rx[20] = {0};

    uint64_t value;

    uint16_t req =  0x403F;

    data_tx[0] = ((req >> 12) & 0xf);
    data_tx[1] = ((req >> 8) & 0xf);
    data_tx[2] = ((req >> 4) & 0xf);
    data_tx[3] = ((req) & 0xf);
    data_tx[4] = 0x0;

    SPI_MAGNETOMETER_NSS_LOW();
    HAL_SPI_TransmitReceive(&hspi1, data_tx, data_rx, 5, 10);
    SPI_MAGNETOMETER_NSS_HIGH();

    value = ((data_rx[0] & 0xf) << 16) |
    ((data_rx[1] & 0xf) << 12) | ((data_rx[2] & 0xf) << 8) |
    ((data_rx[3] & 0xf) << 4) | ((data_rx[4] & 0xf));

    value = value  & 0x3FFFF;
    value = value * 360000;

    value = value / (1<<18);

    return (uint32_t)value;
}
```

4.7.2 Filtration of values

Data from the AEAT-9922 sensor are noisy and using raw input data would cause oscillations. And final precise value would be possible only with some hysteresis ± some number of steps. This caused a need for filtration with averaging of incoming data.

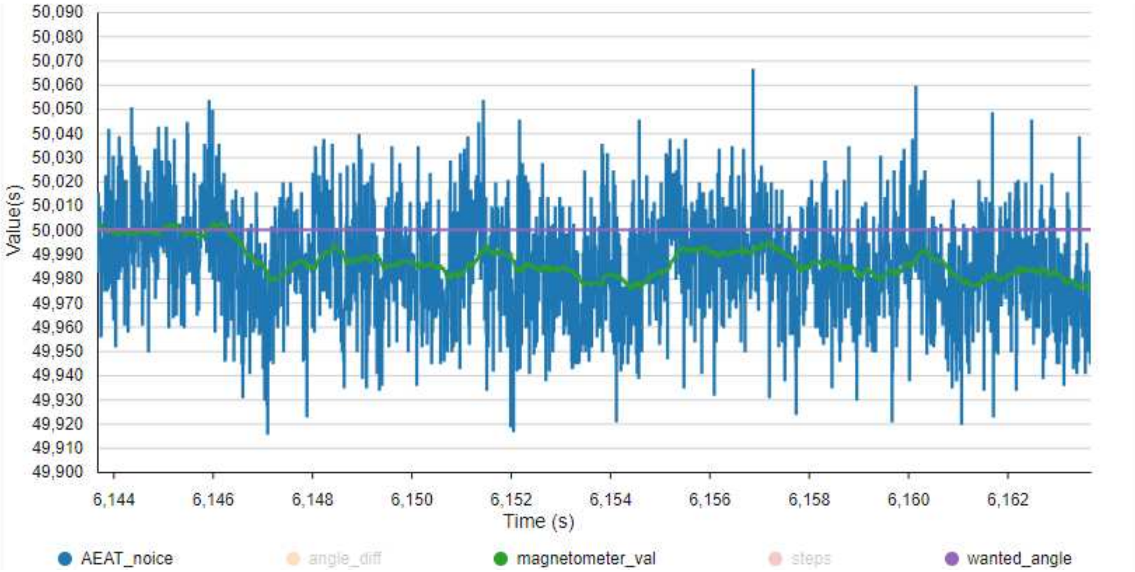


Fig. 4.26: Filtred value *green*, noice from sensor *blue* and wanted angle *purple*. On the vertical axis is angle in (°)

Filtration is done by using a cumulative filter that averages values out over time. Initially, the current filtered angle is saved as the averaged value. Then from this averaged value, the same value shifted by n bits to the right (n defines the strength of the filter) is subtracted. Then newly acquired value from the sensor is added to the averaged value. This value is then again shifted by n bits to the right and the process repeats itself. This algorithm smooths out the sensor readings and reduces pseudo-random noise from AEAT-9922.

Listing 4.3: Cumulative filter code

```
#define APPLY_Q(x)                  ((x)>>MAG_Q)
#define USE_CUMULATIVE_FILTER       (1)
#define MAG_Q                       (6)

#if defined(USE_CUMULATIVE_FILTER) &&
          (USE_CUMULATIVE_FILTER == 1)
        magnetometer_val = APPLY_Q(mag_avg);
        mag_avg -= magnetometer_val;
        mag_avg += aeat9922_read();
#else
        magnetometer_val = aeat9922_read();
#endif
```

Different values of Q (strength of filter) can be found in Appendix B. Generally the more stronger is the filter, the more stable is the averaged value, but on the other hand, more time is needed for this value to settle and stabilize. The ideal value (Q) for the filter turns out to be 6 and will be used further. Q = 6 has a value swing in a stable motor position under the size of one step (under 0.014°).
Additional filter settings can be found in the table lower (Dif P-P (°) means difference between maximal and minimal values in the stable position, Dif steps (-) is recalculated into the size of steps, Settle time (s) means time for filtered value to settle at current value for move of 1000 steps). These values are also in graphic form under the table.

Filtr Q	Off	2	3	4	5	6	7	8	9	10
Dif P-P (°)	0.17	0.075	0.029	0.024	0.023	0.012	0.007	0.004	0.001	0.001
Dif steps (-)	12.1	5.36	2	1.71	1.64	0.85	0,5	0.29	0.07	0.07
Settle time (s)	0.17	0.33	0.58	1.2	4	8	18	25	35	80

Table 4.4: Size of signal noise and time to settle, with different filter parameters

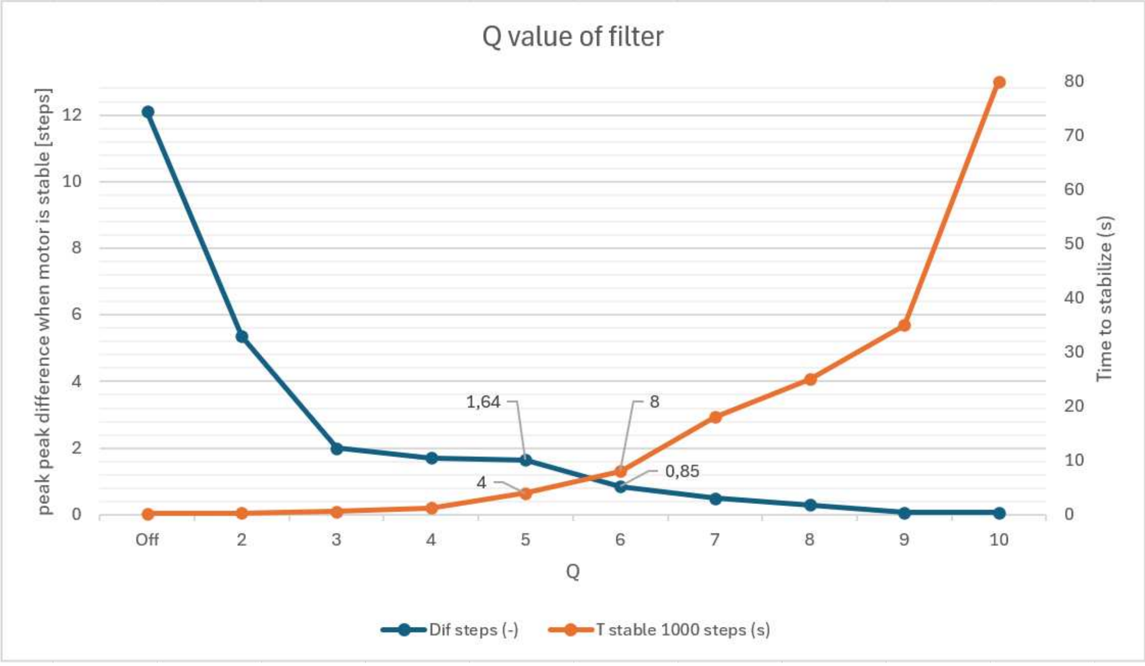
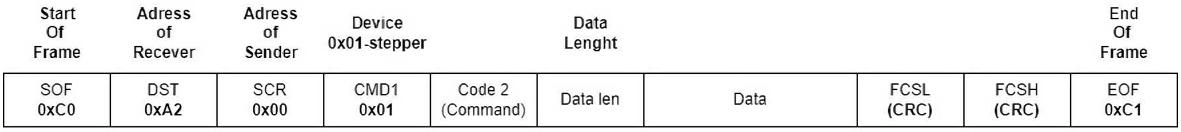


Fig. 4.27: Size of signal noise and time to settle, with different filter parameters

4.7.3 RS-485 control data frames

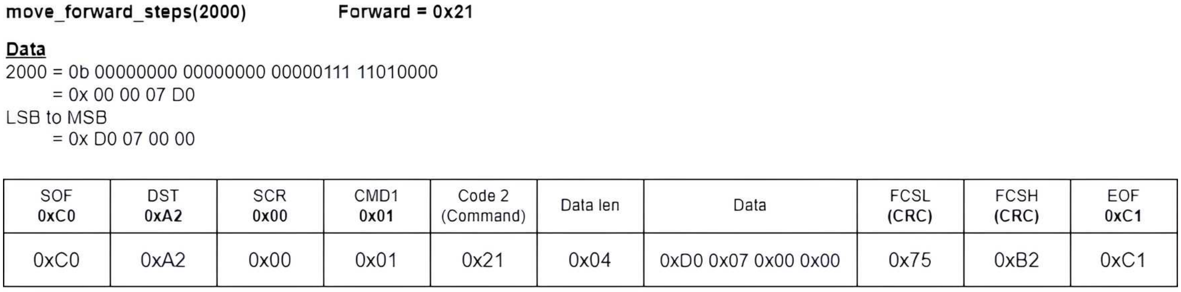
The Stepper motor driver board can be controlled with RS-485 both from the PC and from the master control platform itself. In this case, the PC control will be used. The data frame begins with the start of the frame represented with hexadecimal value *0xC0*. The second part is the address of the receiver (PC has address *0x00* and driver board *0xa2*) Each axis driver will have its unique address for simultaneous control of both axes. Next is the address of the sender, then the device type (*0x01* is for motor drive boards). Next is the command (e.g. rotate forward for a number of steps is *0x21*). after that is the length of data (usually 4 bytes), and after that data with e.g. number of steps, current angle, or wanted angle. The last part of the control frame are two CRC check bytes and an end byte *0xC1*.



Start Of Frame	Adress of Recever	Adress of Sender	Device 0x01-stepper		Data Lenght				End Of Frame
SOF 0xC0	DST 0xA2	SCR 0x00	CMD1 0x01	Code 2 (Command)	Data len	Data	FCSL (CRC)	FCSH (CRC)	EOF 0xC1

Fig. 4.28: General frame format

Format of move forward by 2000 steps frame:



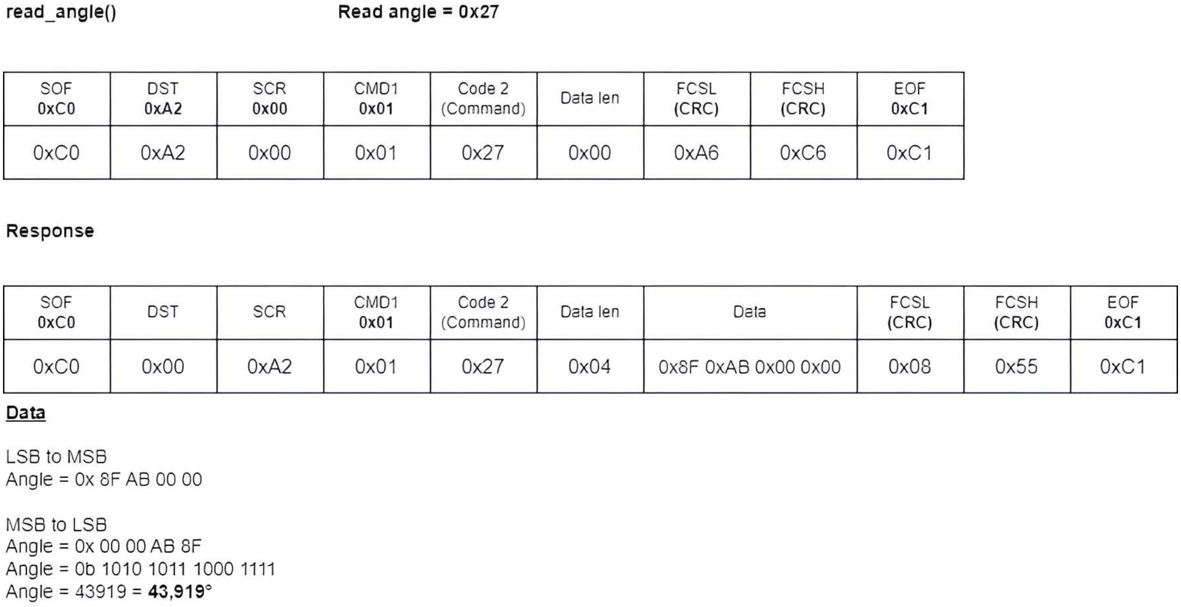
move_forward_steps(2000) **Forward = 0x21**

Data
2000 = 0b 00000000 00000000 00000111 11010000
= 0x 00 00 07 D0
LSB to MSB
= 0x D0 07 00 00

SOF **0xC0**	DST **0xA2**	SCR **0x00**	CMD1 **0x01**	Code 2 (Command)	Data len	Data	FCSL **(CRC)**	FCSH **(CRC)**	EOF **0xC1**
0xC0	0xA2	0x00	0x01	0x21	0x04	0xD0 0x07 0x00 0x00	0x75	0xB2	0xC1

Fig. 4.29: Move forward 2000 steps frame

Format of read angle frame with response 43.919°:



read_angle() **Read angle = 0x27**

SOF **0xC0**	DST **0xA2**	SCR **0x00**	CMD1 **0x01**	Code 2 (Command)	Data len	FCSL (CRC)	FCSH (CRC)	EOF **0xC1**
0xC0	0xA2	0x00	0x01	0x27	0x00	0xA6	0xC6	0xC1

Response

SOF **0xC0**	DST	SCR	CMD1 **0x01**	Code 2 (Command)	Data len	Data	FCSL **(CRC)**	FCSH **(CRC)**	EOF **0xC1**
0xC0	0x00	0xA2	0x01	0x27	0x04	0x8F 0xAB 0x00 0x00	0x08	0x55	0xC1

Data

LSB to MSB
Angle = 0x 8F AB 00 00

MSB to LSB
Angle = 0x 00 00 AB 8F
Angle = 0b 1010 1011 1000 1111
Angle = 43919 = **43,919°**

Fig. 4.30: Read angle output and input frame

Angle data are transmitted from LSB to MSB. When they are being decoded they need to be structured from MOB to LSB.

4.7.4 Control modes

The Stepper driver board currently has 5 control modes. The first one is for reading of the current angle, the second that stops the motor movement, two that move the motor by a specific number of steps, and two that can specify a precise wanted angle.

- **Read angle (0x27)** The read angle command takes the current angle value and calls the function that constructs the data frame and then sends it to the PC or control board.
- **STOP (0x23)** calls function *dSPIN_Hard_Stop();* that sends into the L6470H Stepper motor driver chip instruction with SPI-4 that stops movement and cancels the current task.

Move number of steps

- **Forward (0x21)** This function calls the function: *dSPIN_Move(FWD, rx_data);* and inputs a specified number of steps from received data frame. It also specifies the direction of rotation. This function then sends this command into L6470H.
- **Backward (0x22)** This command works the same way as the previous one but in the opposite direction. Function: *dSPIN_Move(REV, rx_data);*.

Go to specific angle

- **Go to angle (0x26)** This function reads the wanted angle value from the received data frame and then *move_by_angle(wanted_angle);* function is called after *move_by_angle_in_progress = true;*. As input value to this function is *wanted_angle* variable.

Listing 4.4: Go to angle (0x26) function

```
value = *(uint32_t*)(rx_frame->data);

wanted_angle = value;
move_by_angle_in_progress = true;
```

- **Move relative angle (0x25)** This function works the same way as "Go to angle" with the difference that the received data frame value is subtracted from the former wanted angle value.

Listing 4.5: Move relative angle (0x25) function

```
value = *(uint32_t*)(rx_frame->data);

wanted_angle = magnetometer_val - value;
move_by_angle_in_progress = true;
```

Move by angle function

This function is called after *move_by_angle_in_progress = true;* which can be changed with **Go to angle** or **Move relative angle** and takes the wanted angle acquired from those functions.
This regulation algorithm first calculates the angle difference between the current angle and the wanted angle value and sets the stabilized bool variable to false.

Listing 4.6: First part of move by angle function

```
void move_by_angle(uint32_t wanted_angle)
{
    HAL_Delay(Stabilization_time);
    uint32_t current_angle = magnetometer_val;
    angle_diff = current_angle - wanted_angle;
```

Suppose the angle difference is smaller than the negative angle tolerance. In that case, the program calculates how many steps this angle difference corresponds to and divides it by a dividing constant so it does not overshoot, and moves the stepper motor forward for a calculated amount of steps. This process is repeated until the angle difference is above the negative angle tolerance. If the angle difference is bigger than the positive angle tolerance, the stepper motor moves a calculated amount of steps in the backward direction.

Listing 4.7: Second part of move by angle function

```
void move_by_angle(uint32_t wanted_angle)
    if (angle_diff < -angle_tollerance)
    {
        steps = -1*round((angle_diff/(360000/
        (200*Microstepp_setting)))/Dividing_constant);
        spi1_reinit_to_stepper();
        dSPIN_Move(FWD,steps);
        spi1_reinit_to_magnetometer();
        stabilized = false;
    }
    else if (angle_diff > angle_tollerance)
    {
        steps = round(((angle_diff/(360000
        /(200*Microstepp_setting))))/Dividing_constant);
        spi1_reinit_to_stepper();
        dSPIN_Move(REV, steps);
        spi1_reinit_to_magnetometer();
        stabilized = false;
```

The last part of this code waits for 0.5 seconds and then changes the stabilized to true and then stops movement.

Listing 4.8: Third part of move by angle function

```
    else
    {
        if (stabilized)
        {
            move_by_angle_in_progress = false;
        }
        else
        {
            HAL_Delay(500);
            stabilized = true;
        }
    }
}
```

4.7.5 Program for control driver boards from PC

Data packages for the driver module board can be created on paper but a better alternative is to create a program that can do it automatically. The core of this Python program was already completed but usage was impractical so a Tkinter GUI was created for easier manipulation and lesser time consumption
The code under is creating data frames by combining its parts as described in the previous subchapter 4.7.3. The address of the wanted device is copied from an entry window in the GUI.

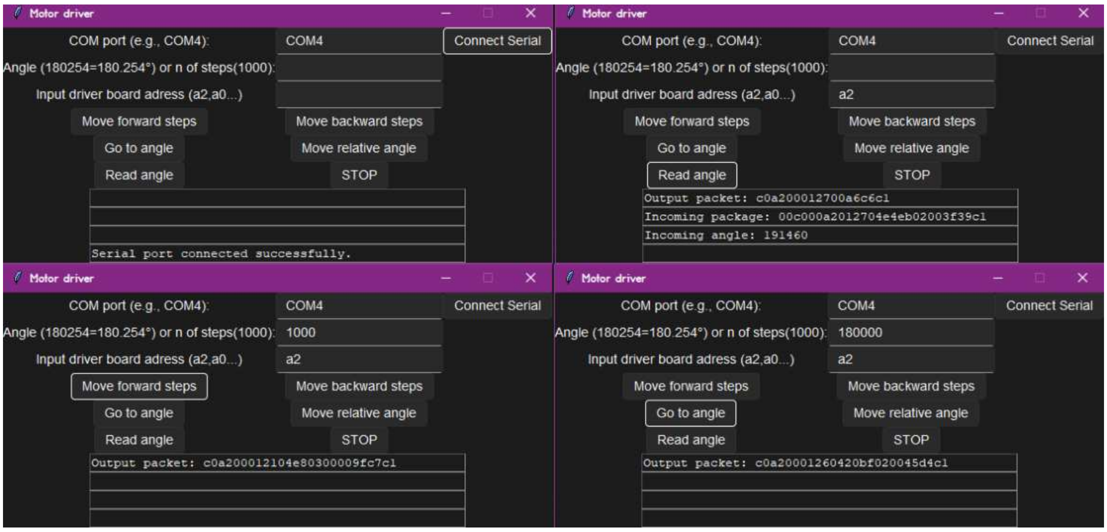
Listing 4.9: Python program for RS-485 data frames generation

```
FRAME_SOF = bytes([0xC0])
FRAME_EOF = bytes([0xC1])
SCR = bytes([0x00])
CMD1_STEPPER = bytes([0x01])

data_len = bytes([data_len & 0xFF])

crc_calculator = Calculator(Crc16.XMODEM)
address_str = entry3.get()
address_int = int(address_str, 16)
address = bytes([address_int])

frame = address + SCR + CMD1_STEPPER + code2 + data_len + data

checksum = crc_calculator.checksum(frame)
fcsl = bytes([checksum & 0xFF])
fcsh = bytes([(checksum >> 8) & 0xFF])

frame = FRAME_SOF + frame + fcsl + fcsh + FRAME_EOF
ser.write(frame)
```

At first, the user is supposed to write a port in which the USB to RS-485 convertor is connected and press the "Connect Serial" button. The last field in the code above informs the user of a successful connection. If the connection was successful, the user should write an address of a wanted driver board (0xa2 was used as the address). The last information that the user inputs is an angle (for "go to" and "move relative" buttons) or a number of steps (for "move forward" or " move backward" buttons). For all the buttons, the sent data frame is displayed in the first data field, and the incoming data frame is displayed in the second data field.

4.7.6 Control program interface

This interface has 3 entry fields (first for COM port name, second to input angle or steps, third for the address of the board), one button that connects to the COM port, 4 buttons for movement, one button for reading the current angle and one that stops movement. Next, it has two output fields with outcoming and incoming frames, a third for reading angle and the last for confirmation of connection to COM.



```
COM port (e.g., COM4): COM4
Angle (180254=180.254°) or n of steps(1000):
Input driver board adress (a2,a0...)
Serial port connected successfully.
```

```
COM port (e.g., COM4): COM4
Angle (180254=180.254°) or n of steps(1000):
Input driver board adress (a2,a0...) a2
Output packet: c0a200012700a6c6c1
Incoming package: 00c000a2012704e4eb02003f39c1
Incoming angle: 191460
```

```
COM port (e.g., COM4): COM4
Angle (180254=180.254°) or n of steps(1000): 1000
Input driver board adress (a2,a0...) a2
Output packet: c0a200012104e80300009fc7c1
```

```
COM port (e.g., COM4): COM4
Angle (180254=180.254°) or n of steps(1000): 180000
Input driver board adress (a2,a0...) a2
Output packet: c0a20001260420bf020045d4c1
```

Fig. 4.31: Graphical user interface of python code for driver control

4.7.7 Parameters affecting regulation

The parameters that change regulation behavior are angle tolerance (±angle that will be tolerated around the wanted angle value, can help with stability), the strength of filter Q (MAG_Q - Number of bits to shift to right), dividing constant makes sure that the wanted angle is not overshot (100 steps move = 100/1.2 = 83 steps move) and stabilization time (time between iterations, the stronger and slower filter is, the longer it needs). For filter strength 6 ideal time is around 2.5 seconds.

Listing 4.10: Parameters affecting regulation

```
#define APPLY_Q(x)              ((x)>>MAG_Q)
#define angle_tollerance        (0)   // 1 = 0.001 degrees
#define Microstepp_setting      (128)
#define USE_CUMULATIVE_FILTER   (1)   // 1 = on, 0 = off
#define MAG_Q                   (6)   // (2-10)
#define Dividing_constant       (1.2)
#define Stabilization_time      (2500)
```

4.7.8 Tests and Evaluation of Final Motor Driver board regulation and accuracy

The program can get to maximal accuracy with few iterations in a matter of seconds. The driver is capable of holding this wanted angle even if an external force moves with the platform, in which case the program comes back into the wanted angle. The accuracy of this method is 24.5 cm per step per 1 km. This accuracy could be improved to 12.3 cm per step if the current 1.8° per step stepper motors were replaced with 0.9° per step stepper motors. In the figures below 10° movement can be seen (which is the maximal angle distance on this platform in both axes). The platform took about 10 seconds to reach the maximal accuracy (with an error of around 0.008°).

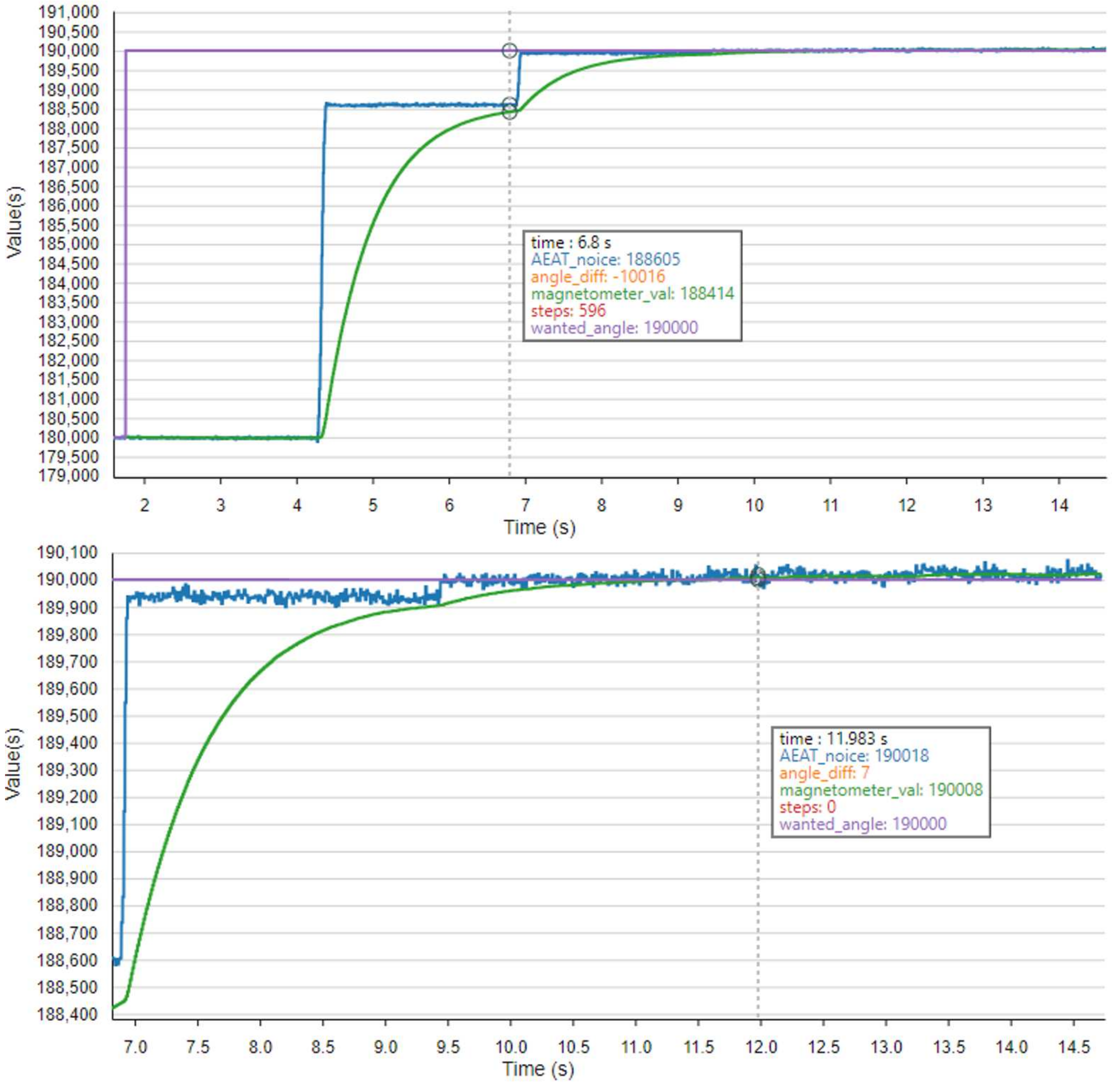


Fig. 4.32: Movement of 10° (Time on x axis and angle in degrees on y axis)

4.7.9 Implementation to platform

In the base of the opto-mechanical platform, the holder for the AEAT-9922 sensor was already implemented at the azimuth axis but the holder of the sensor at the altitude axis wasn't implemented yet. Two new holders were Designed in Autodesk Fusion, one for the magnet and a second for the sensor.

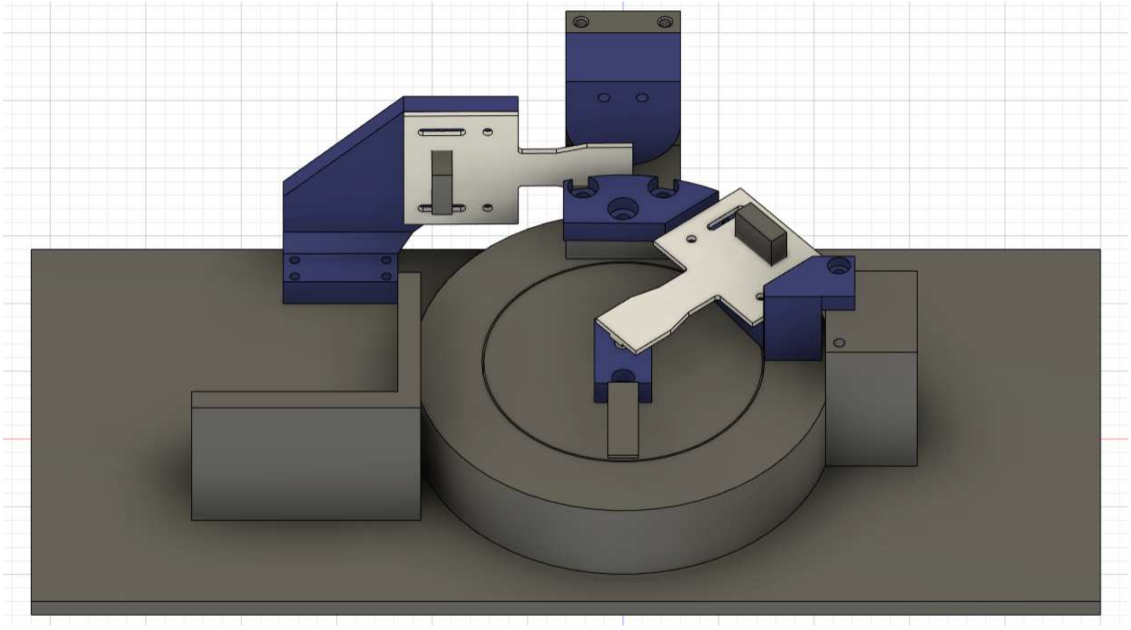


Fig. 4.33: Holders for sensor and magnet top view

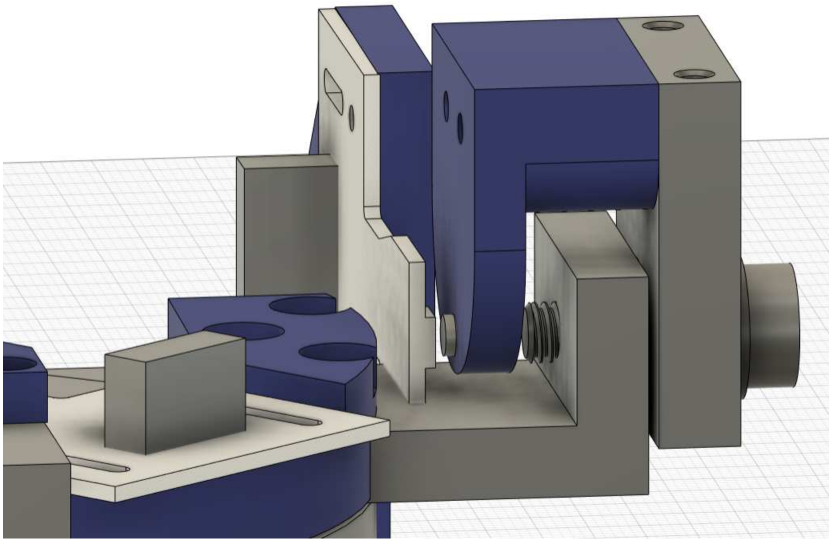


Fig. 4.34: Holders for sensor and magnet side view

Conclusion

My bachelor's thesis had a few major tasks. The first one was aimed at describing and comparing different types of absolute rotary sensors. A second tasks was to design the final driver board for the opto-mechanical platform.
The first tasks was achieved in a theoretical part where potentiometers, resolvers, encoders, and hall effect sensors were described and compared. The second tasks was also achieved. The first step was to learn how to communicate and read angle data from the 3D hall-effect sensor AEAT-9922 with the use of the SPI-4. Reading of data was achieved with 18-bit resolution, which means there were 262144 binary combinations for full revolution (0.00137° per binary state). The next step was about control of the stepper motor using a stepper motor driver. I used a DRV8825 driver from the company Texas Instruments which allows 1/32 microstepping setting. This setting offers a resolution of 0.056° per step. Another challenge was to create a prototype of the driver. For this, I used the Arduino UNO platform. The prototype was successfully built and I was able to get a maximal resolution of 98.2 cm per step per km. The final step was to redesign the previous version of the opto-mechanical platform driver board to include an SPI-4 interface to communicate with the AEAT-9922 sensor. This version of the driver is controlled by a microcontroller STM32F303K6 and the stepper motor is driven by stepper motor driver L6470H. The L6470H has a microstepping resolution of 0.014° per step. With this microstepping setting, the final resolution of the optic platform could be 24.5 cm per step per km.
The next task was to integrate two of those drivers and test if they were functional and both of them were. The following task was to modify the program of the driver board for open loop application and movement to a specific angle. This task was also successful and I was able to reach the maximal resolution mentioned earlier. The graphic user interface for the Python program in order to control the driver board with RS-485 protocol from a PC was also programmed.
The last task was to model two holders in Autodesk Fusion. One for the AEAT sensor and the second for a magnet. Renders of those parts were modelled successfully and can be found in Appendix C.

Bibliography

[1] ELECTRICAL 4 U. *Potentiometer: Definition, Types, And Working Principle* Online. Available at: https://www.electrical4u.com/potentiometer/.
[cit. 2023-12-03].

[2] Dynamics Research Corporation. *Techniques For Digitizing Rotary and Linear Motion.* Online. Available at: http://irtfweb.ifa.hawaii.edu/~tcs3/tcs3/0306_conceptual_design/Docs/05_Encoders/encoder_primer.pdf.
[cit. 2023-12-03].

[3] COMPONENTS101. *Potentiometer.* Online. Available at: https://components101.com/resistors/potentiometer.
[cit. 2023-12-03].

[4] DIRECT INDUSTRY BY VIRTUALEXPO GROUP. *Rotary position sensor P-6500* Online. Available at: https://www.directindustry.com/prod/novotechnik/product-13745-518580.html.
[cit. 2023-12-25].

[5] DIRECT INDUSTRY BY VIRTUALEXPO GROUP. *Draw-wire position sensor XL80* Online. Available at: https://www.directindustry.com/prod/sentop-shanghai-sibo-m-e-co-ltd/product-235548-2543809.html.
[cit. 2023-12-25].

[6] DIRECT INDUSTRY BY VIRTUALEXPO GROUP. *Rotary position sensor AW-360* Online. Available at: https://www.directindustry.com/prod/novotechnik/product-13745-518667.html.
[cit. 2023-12-25].

[7] A-TECH INSTRUMENTS LTD. *AWS360, Hybrid Rotary Displacement Potentiometer* Online. Available at: https://www.a-tech.ca/Product/Series/1850/AWS360_Hybrid_Rotary_Displacement_Potentiometer/?tab=2.
[cit. 2023-12-25].

[8] DYNAPAR™. *What is a Resolver?.* Online. Available at: https://www.dynapar.com/technology/encoder_basics/resolvers/.
[cit. 2023-12-03].

[9] ELTRA ENCODER, 2020. *ENCODER VERSUS RESOLVER. WHAT THE DIFFERENCE BETWEEN THEM?.* Online. Available at: https://eltra-encoder.eu/news/resolver-vs-encoder.
[cit. 2023-12-03].

[10] ALIBABA.COM, 2011. *Tyco Rotary Encoder Resolver D1010-B101.* Online. Available at: https://www.alibaba.com/product-detail/V23401-D1010-B101-Tyco-Rotary-Encoder_60697014623.html.
[cit. 2023-12-25].

[11] LEARNCHANNEL-TV. *How does a resolver work.* Online. Available at: https://learnchannel-tv.com/en/drives/servomotor/resolver/.
[cit. 2023-12-03].

[12] DYNAPAR™, 2013. *Resolvers 101 Understanding the Basics* Online. Available at: https://www.jdhengstler.co.uk/admin/download-attachment.php?file_id=resolverbasicswpvf.pdf.
[cit. 2023-12-03].

[13] DYNAPAR™. *Resolver Speed and Accuracy* Online. Available at: https://www.dynapar.com/Knowledge/Resolver-Speed-Accuracy/.
[cit. 2023-12-03].

[14] RICE, Rick (ed.), CONTROL DESIGN, 2021. *Advantages of encoders vs. resolvers* Online. Available at: https://www.controldesign.com/sensing/encoders/article/11293849/advantages-of-encoders-vs-resolvers.
[cit. 2023-12-03].

[15] DYNAPAR™, 2008. *Resolver Selection Guide datasheet* Online. Available at: https://www.hengstler.de/gfx/file/shop/encoder/BRCX_series/datasheet_resolver_en.pdf.
[cit. 2023-12-25].

[16] DYNAPAR™. *What Are Absolute Encoders and When to Use Them* Online. Available at: https://www.dynapar.com/technology/absolute-rotary-encoders/.
[cit. 2023-12-03].

[17] COLLINS, Danielle, LINEAR MOTIONTIPS, 2015. *How encoder resolution is determined* Online. Available at: https://www.linearmotiontips.com/how-encoder-resolution-is-determined/.
[cit. 2023-12-03].

[18] GLOBALSPEC. *LEARN MORE ABOUT ROTARY ENCODERS* Online. Available at: https://www.globalspec.com/learnmore/sensors_transducers_detectors/encoders_resolvers/rotary_encoders.
[cit. 2023-12-03].

[19] RESEARCHGATE. *Incremental encoder principle* Online. Available at: https://www.researchgate.net/figure/Incremental-encoder-principle-25_fig19_309674149.
[cit. 2023-12-25].

[20] Wikipedia: the free encyclopedia. San Francisco (CA): Wikimedia Foundation. *Rotary encoder* Online. Available at: https://en.wikipedia.org/wiki/Rotary_encoder.
[cit. 2023-12-03].

[21] HAMAMATSU. *Encoder* Online. Available at: https://www.hamamatsu.com/jp/en/applications/industrial-equipment/optical-encoder.html.
[cit. 2023-12-03].

[22] KHAN, Zak, MOTION CONTROLTIPS, 2015. *FAQ: What is the difference between absolute and incremental encoders?* Online. Available at: https://www.motioncontroltips.com/faq-what-is-the-difference-between-absolute-and-incremental-encoders/.
[cit. 2023-12-03].

[23] MANO, M. Morris and KIME, Charles, 2015. *Logic and Computer Design Fundamentals. Fourth Edition. Pearson.* Online. Available at: https://drive.uqu.edu.sa/_/mskhayat/files/MySubjects/20178FS%20LogicAnalysisAndDesign/Logic_and_Computer_Design_Fundamentals_-_4th_International_Edition.pdf.
[cit. 2023-12-03].

[24] ANAHEIM AUTOMATION. *Logic and Computer Design Fundamentals. Fourth Edition. Pearson.* Online. Available at: https://www.anaheimautomation.com/manuals/forms/magnetic-encoder-guide.php.
[cit. 2023-12-03].

[25] MACON. *Bogen.* Online. Available at: https://www.maccon.com/rotary-linear-encoders/magnetic-encoders/bogen.html.
[cit. 2023-12-03].

[26] ANAHEIM AUTOMATION. *Introduction to Capacitive Encoders.* Online. Available at: https://www.anaheimautomation.com/manuals/forms/capacitive-encoders-guide.php.
[cit. 2023-12-03].

[27] ANAHEIM AUTOMATION. *ENC-A6SIH - Optical Encoders.* Online. Available at: https://www.anaheimautomation.com/products/encoder/

incremental-rotary-item.php?sID=894&serID=126&pt=i&tID=1054&cID=422.
[cit. 2023-12-03].

[28] ANAHEIM AUTOMATION. *ENC-EC35 - Optical Encoders.* Online. Available at: https://www.anaheimautomation.com/products/encoder/incremental-rotary-item.php?sID=399&serID=54&pt=i&tID=1054&cID=422.
[cit. 2023-12-03].

[29] COLLINS, Danielle, MOTION CONTROLTIPS, 2015. *What are capacitive encoders and where are they suitable?* Online. Available at: https://www.motioncontroltips.com/faq-what-are-capacitive-encoders-and-where-are-they-suitable/.
[cit. 2023-12-03].

[30] DEAMER, Lanna, 2019. *Choosing the right rotary encoder for ruggedness and performance.* Online. Available at: https://www.electronicspecifier.com/products/sensors/choosing-the-right-rotary-encoder-for-ruggedness-and-performance.
[cit. 2023-12-03].

[31] TEXAS INSTRUMENTS, 2022. *Introduction to Hall-Effect Sensors.* Online. Available at: https://www.ti.com/lit/ml/slyt824a/slyt824a.pdf?ts=1699359871375.
[cit. 2023-12-03].

[32] TEXAS INSTRUMENTS, 2021. *Absolute Angle Measurements for Rotational Motion Using Hall-Effect Sensors.* Online. Available at: https://www.ti.com/lit/an/sbaa503a/sbaa503a.pdf.
[cit. 2023-12-03].

[33] ANAHEIM AUTOMATION. *Stepper Motor Guide.* Online. Available at: https://www.anaheimautomation.com/manuals/forms/stepper-motor-guide.php.
[cit. 2023-12-25].

[34] ISL PRODUCTS. *STEPPER MOTOR FUNDAMENTALS.* Online. Available at: https://islproducts.com/design-note/stepper-motor-fundamentals/.
[cit. 2023-12-25].

[35] Dejan, 2022. *Stepper Motors and Arduino – The Ultimate Guide.* Online. Available at: https://howtomechatronics.com/tutorials/arduino/stepper-motors-and-arduino-the-ultimate-guide/.
[cit. 2023-12-25].

[36] WALTER, Moritz, 2016. *HOW ACCURATE IS MICROSTEPPING REALLY?.* Online. Available at: https://hackaday.com/2016/08/29/how-accurate-is-microstepping-really/.
[cit. 2023-12-25].

[37] FAULHABER. *Stepper Motor Technical Note: Microstepping Myths and Realities.* Online. Available at: https://www.faulhaber.com/en/know-how/tutorials/stepper-motor-tutorial-microstepping-myths-and-realities/.
[cit. 2023-12-25].

[38] Tessie, UTMEL ELECTRONIC, 2021. *NEMA17 Stepper Motor: Datasheet pdf, 1.5 A 1.8° Stepper Motor and Dimensions.* Online. Available at: https://www.utmel.com/components/nema17-stepper-motor-datasheet-pdf-1-5-a-1-8%C2%B0-stepper-motor-and-dimensions?id=914.
[cit. 2023-12-25].

[39] BROADCOM. *AEAT-9922 Programmable Angular Magnetic Encoder.* Online. Available at: https://www.broadcom.com/products/motion-control-encoders/absolute-encoders/single-turn-encoders/aeat-9922.
[cit. 2023-12-25].

[40] BROADCOM, 2021. *AEAT-9922 Data sheet.* Online. Available at: https://docs.broadcom.com/doc/AEAT-9922-Programmable-Angular-Magnetic-Encoder-IC-DS.
[cit. 2023-12-25].

[41] BROADCOM, 2021. *AEAT-9922-Q24 Application Note.* Online. Available at: https://docs.broadcom.com/doc/AEAT-9922-Q24-10-to-18-Bit-Programmable-Angular-Magnetic-Encoder-IC-AN.
[cit. 2023-12-25].

[42] TEXAS INSTRUMENTS, 2010. *RV8825 Stepper Motor Controller IC.* Online. Available at: https://www.ti.com/lit/ds/slvsa73f/slvsa73f.pdf.
[cit. 2023-12-25].

[43] PROTO-PIC. *DRV8825 Stepper Motor Driver Carrier, High Current.* Online. Available at: https://proto-pic.co.uk/product/drv8825-stepper-motor-driver-carrier-high-current.
[cit. 2023-12-25].

[44] STMICROELECTRONICS. *STM32F303K6 Mainstream Mixed signals MCUs Arm Cortex-M4 core with DSP and FPU, 32 Kbytes of Flash memory,*

72 MHz CPU, CCM, 12-bit ADC 5 MSPS, comparators, op-amp. Online. Available at: https://www.st.com/en/microcontrollers-microprocessors/stm32f303k6.html.
[cit. 2023-12-25].

[45] STMICROELECTRONICS. *STM32F303x6/x8 datasheet.* Online. Available at: https://www.st.com/resource/en/datasheet/stm32f303k6.pdf.
[cit. 2023-12-25].

[46] STMICROELECTRONICS. *L6470 Fully integrated microstepping motor driver with motion engine and SPI.* Online. Available at: https://www.st.com/en/motor-drivers/l6470.html.
[cit. 2023-12-25].

[47] STMICROELECTRONICS. *L6470 datasheet.* Online. Available at: https://cz.mouser.com/datasheet/2/389/l6470-1849602.pdf.
[cit. 2023-12-25].

[48] Wikipedia: the free encyclopedia. San Francisco (CA): Wikimedia Foundation. *RS-485* Online. Available at: https://en.wikipedia.org/wiki/RS-485.
[cit. 2024-05-20].

Symbols and abbreviations

ABI	A, B incremental signals and Index signal providing a reference point
AC	Alternating Current
ADC	Analog-to-Digital Converter
CAN	Controller Area Network
COM	Communication Port
CPHA	Clock Phase
CPOL	Clock Polarity
CPU	Central Processing Unit
CPR	Cycles Per Revolution
CRC	Cyclic Redundancy Check
DAC	Digital-to-Analog Converter
EEPROM	Electrically Erasable Programmable Read-Only Memory
EF	Error Flag
EMC	Electromagnetic Compatibility
FSO	Free-Space Optical
GPIO	General Purpose Input/Output
GUI	Graphical User Interface
HAL	Hardware Abstraction Layer
I2C	Inter-Integrated Circuit bus
LED	Light Emitting Diode
LSB	Least Significant Bit
MISO	Master in Slave Out
MOSI	Master Out Slave In

MSB	Most Significant Bit
NCS	Negate Chip Select
P-P	Peak to Peak
PPR	Pulses Per Revolution
PWM	Pulse-width modulation
RF	Radio Frequency
RPM	Revolutions Per Minute
RX	Received
SCK	Clock
SPI	Serial Peripheral Interface
SRAM	Static Random Access Memory
SSI	Synchronous Serial Interface
SWD	Serial Wire Debug
TX	Transmitted
UART	Universal Asynchronous Receiver Transmitter
USART	Universal Synchronous And Asynchronous Receiver Transmitter
UVW	Three-channel integrated commutation output

A Appendix

A.1 Scheme

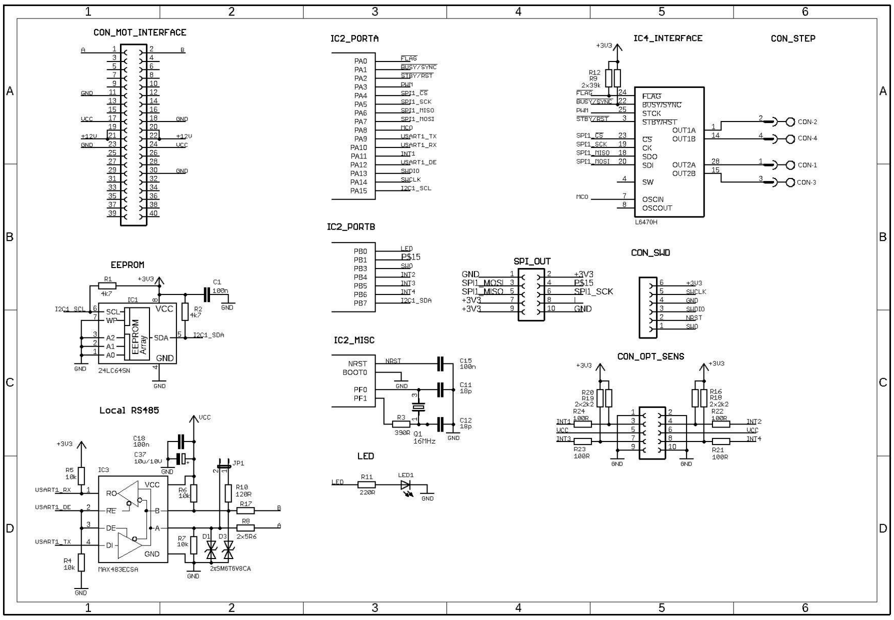


Fig. A.1: Schematic of driver page 1

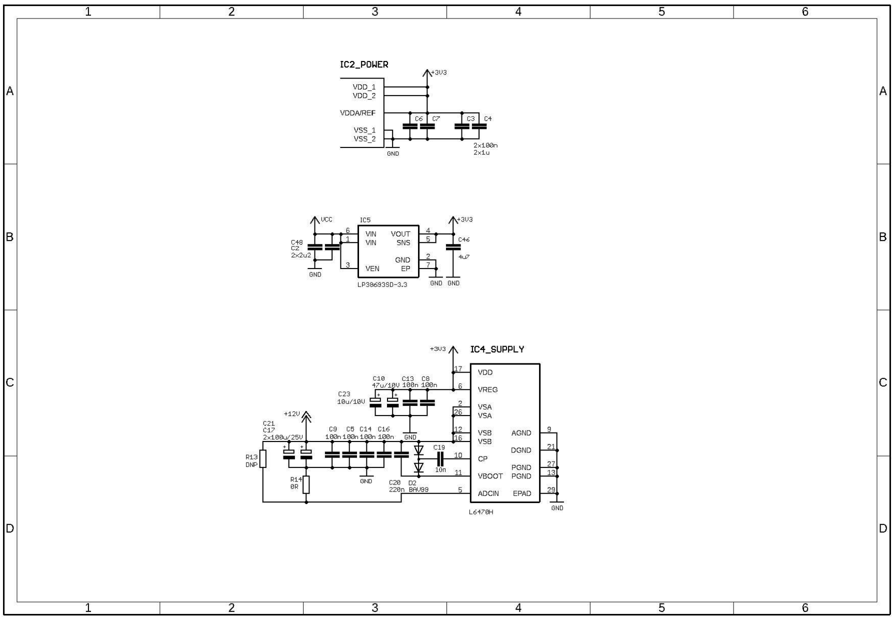


Fig. A.2: Schematic of driver page 2

A.2 Layout of routes

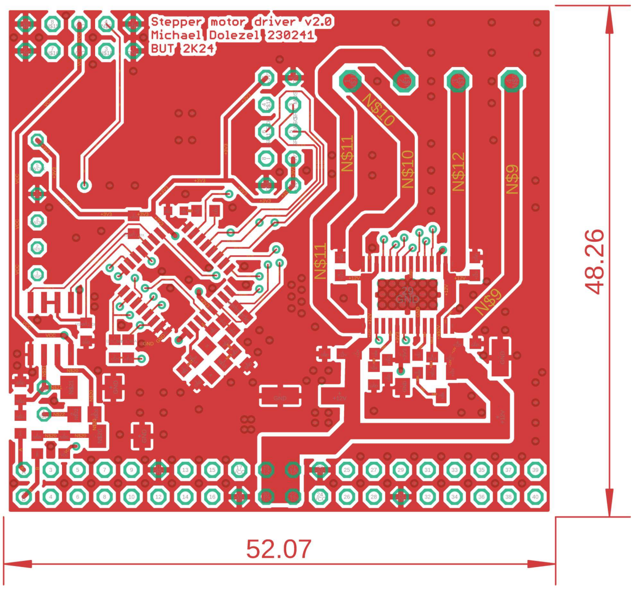


Fig. A.3: Layout of board top view

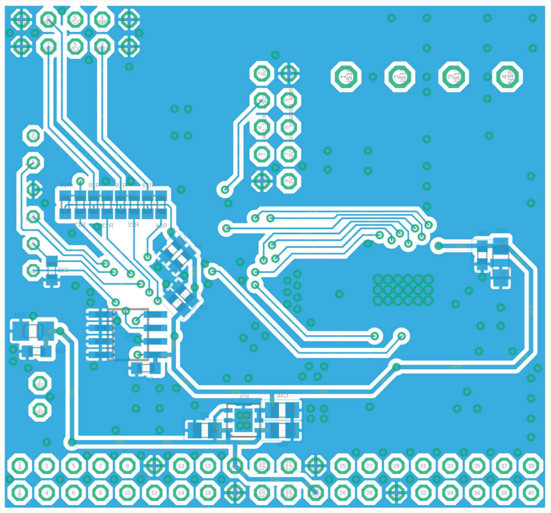


Fig. A.4: Layout of board bottom view

A.3 Layout of components

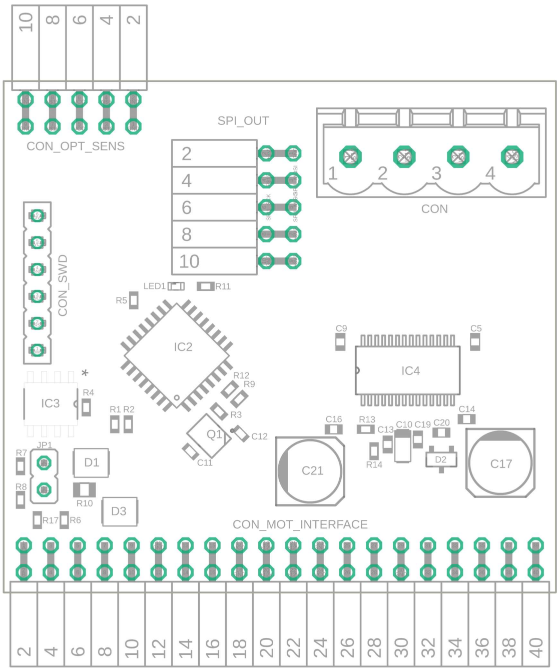


Fig. A.5: Components on board top view

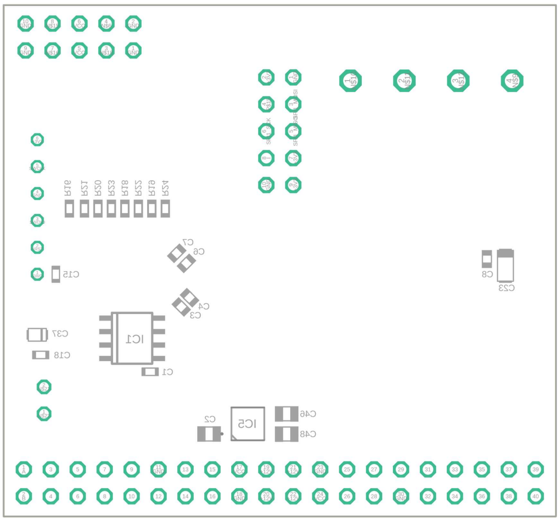


Fig. A.6: Components on board bottom view

A.4 List of the parts

Table A.1: Components list

Label	Value / Name	Package
IC1	24LC64-I/SN	SO-08
IC2	STM32F3-303K6Tb	LQFP32-7X7 (32 pin)
IC3	MAX483ECSA	SOIC127P600X175-8N
IC4	L6470H	SOP65P640X120-29N
IC5	LP38693SD-3.3	DFN300X300X80-7N
Q1	TXC-7M-16.000MAAJ-T	NA
D1, D3	SM6T6V8CA	SMB
D2	BAV99	SOT23
LED	Green	SML0603
R1, R2	4.7 K	R0603
R3	390 R	R0603
R4, R5, R6, R7	10 K	R0603
R8, R17	5.6 R	R0603
R9, R12	39 K	R0603
R10	120 R	R0603
R11	220 R	R0603
R13	Do not populate	NA
R14	0 R	R0603
R16, R18, R19, R20	2.2 K	R0603
R21, R22, R23, R24	100 R	R0603
C1, C4, C5, C6, C8, C9, C13, C14, C15, C16, C18	100 nF	C0603
C2, C48	2.2 uF	C0805
C3, C7	1 uF	C0603
C10	47uF/10V	SMCA 2917
C11, C12	18 pF	C0603
C17, C21	100 uF/25V	SMD
C19	10 nF	C0603
C20	220 nF	C0603
C23, C37	10 uF/10V	SMCA 2917
C46	4.7uF	C0805

B Appendix

B.1 Filter settings

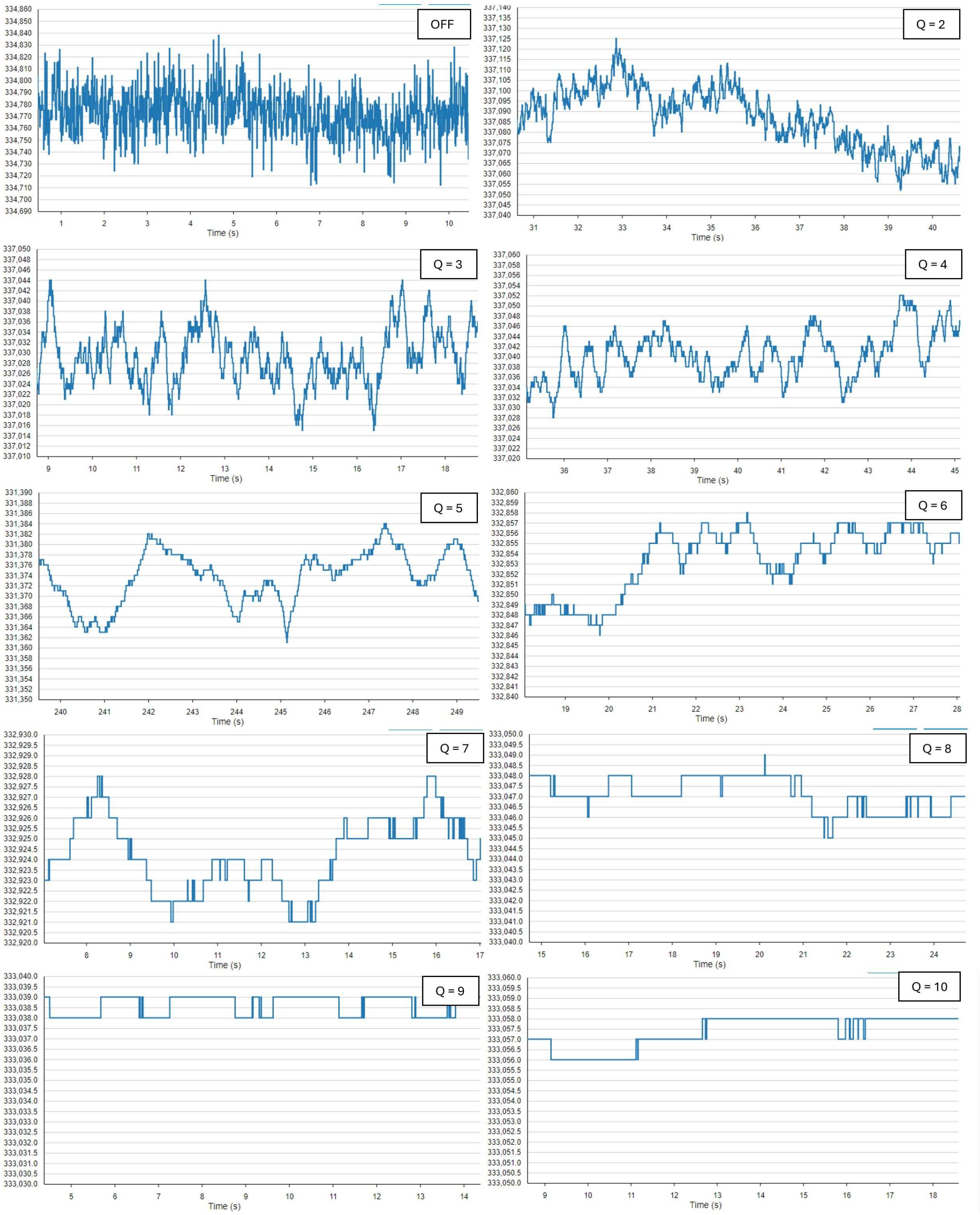


Fig. B.1: Angle value relation to filter settings

B.2 Filter settings 2

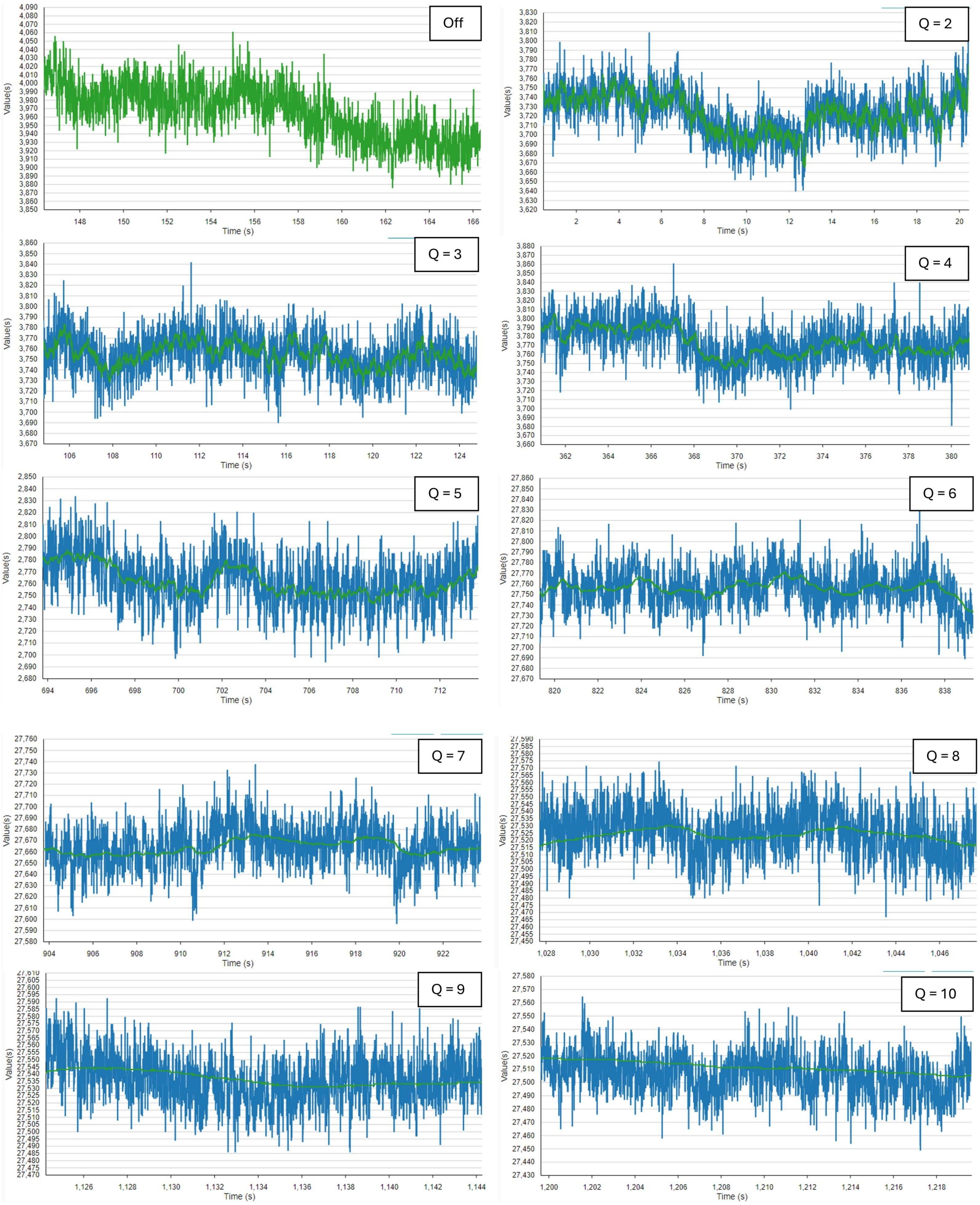


Fig. B.2: Angle value relation to filter settings with original noice

C Appendix

C.1 Renders of platform with holders for magnets and AEAT-9922 sensors

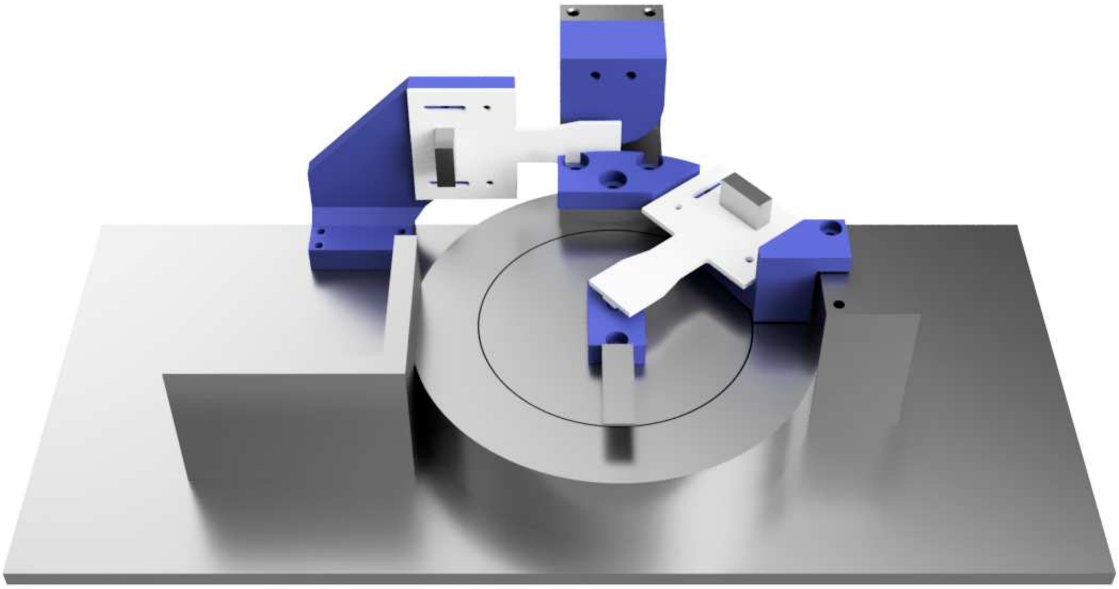


Fig. C.1: Render of platform with holders on both axes - front

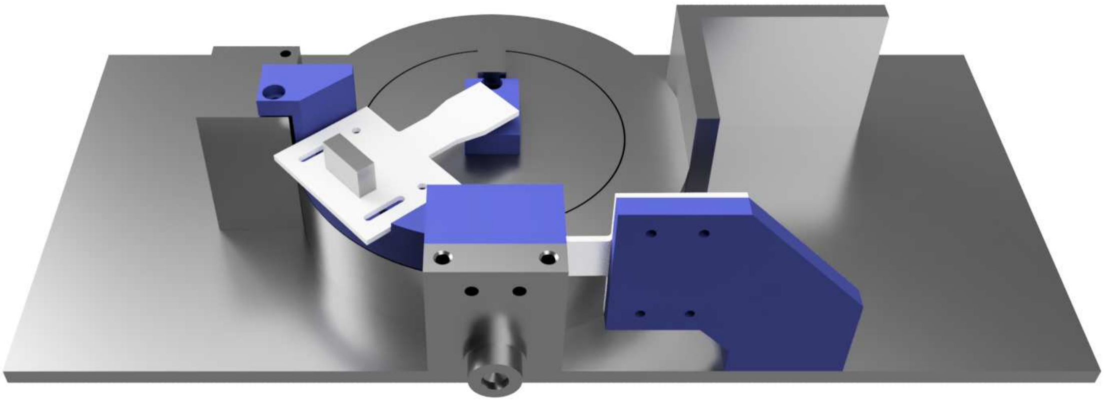


Fig. C.2: Render of platform with holders on both axes - back

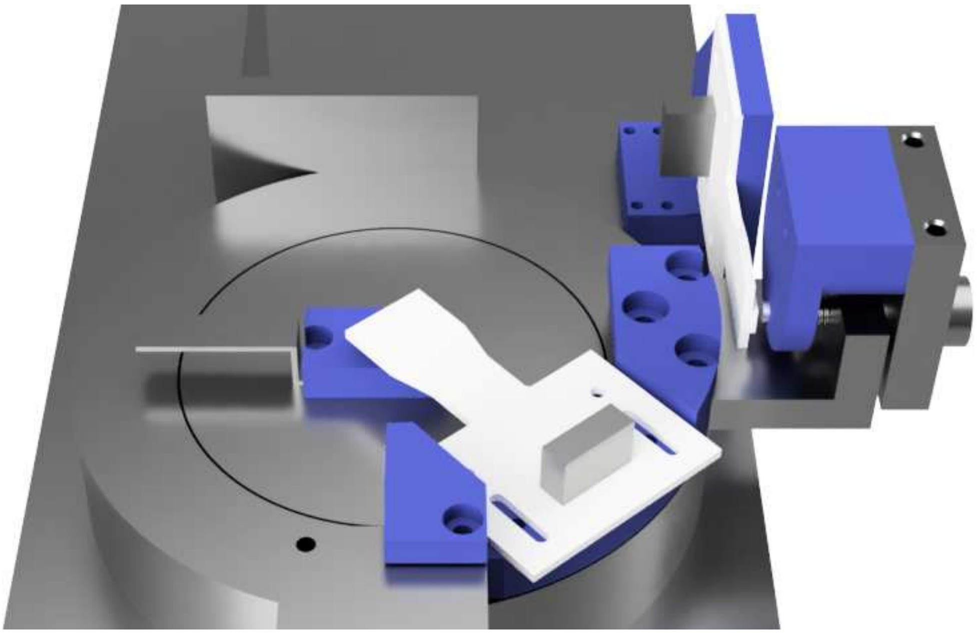


Fig. C.3: Render of platform with holders on both axes - side

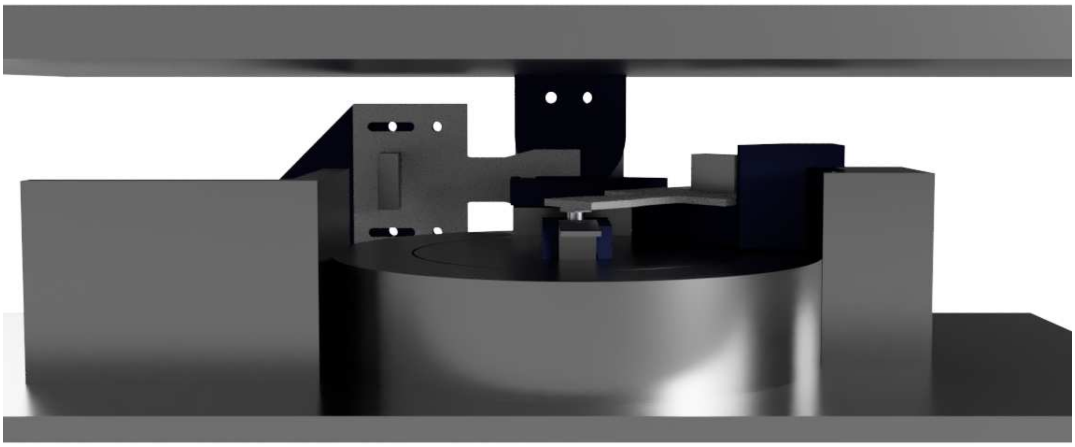


Fig. C.4: Render of platform with holders on both axes with upper optic part

C.2 Comparison of current platform and future version with holders on both axes

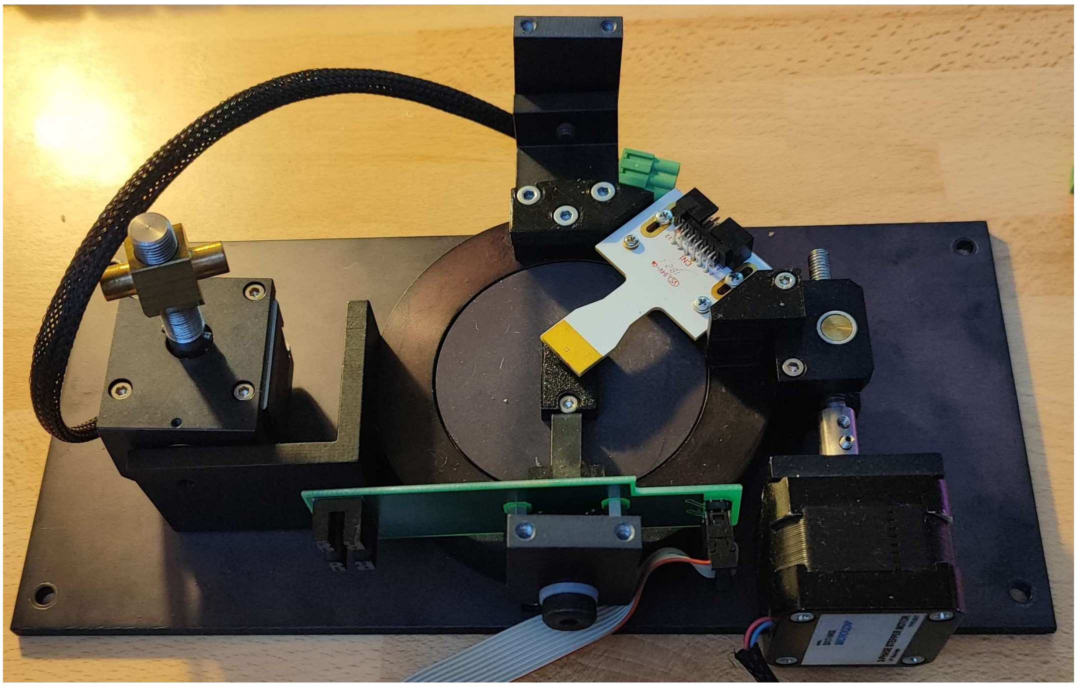


Fig. C.5: Photo of current version just with holders on rotational axis

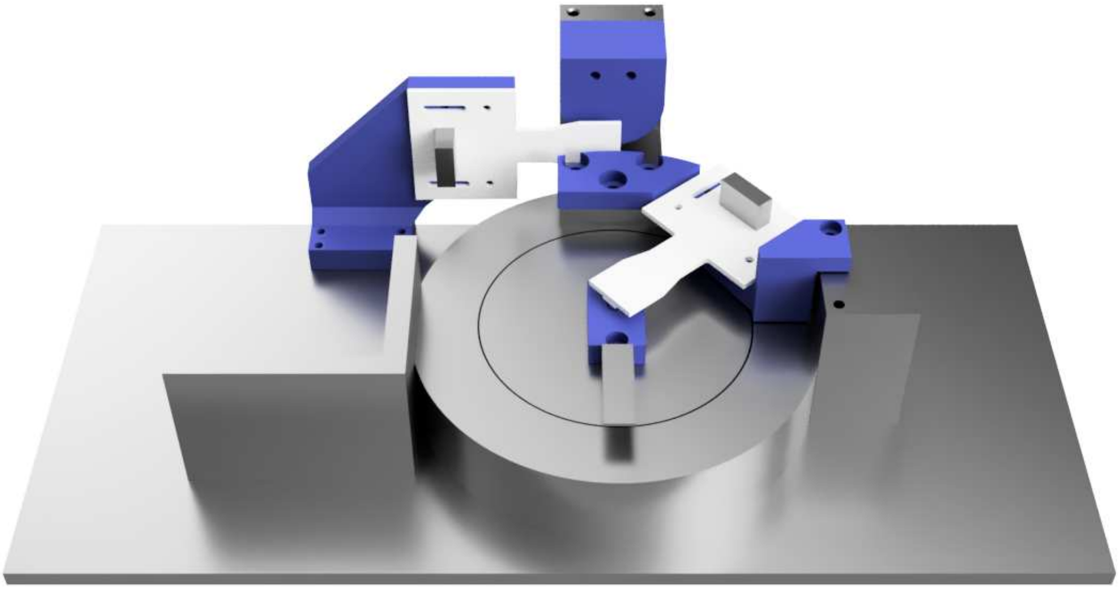


Fig. C.6: Render of platform with holders on both axes for comparison