CODEN:LUTEDX/(TEIE-3106)/1-56/(2022)

Industrial Electrical Engineering and Automation

Gimbal stabilization of superzoom cameras

Max Borglowe

Division of Industrial Electrical Engineering and Automation
Faculty of Engineering, Lund University

Gimbal stabilization of superzoom cameras

LUNDS
UNIVERSITET
Lunds Tekniska Högskola

LTH Ingenjörshögskolan vid Campus Helsingborg

Industrial Electrical Engineering and Automation

Author: Max Borglowe

Supervisors: Leonardo Colombo, Christoffer Cederberg
Examinator: Ulf Jeppsson

LTH Ingenjörshögskolan vid Campus Helsingborg
Lunds universitet
Box 882
251 08 Helsingborg

LTH School of Engineering
Lund University
Box 882
SE-251 08 Helsingborg
Sweden

Tryckt i Sverige
Media-Tryck
Biblioteksdirektionen
Lunds universitet
Lund 2021

Sammanfattning

Med dagens framsteg inom kamerateknik så är bildstabilisering ett väsentligt moment när det gäller att fånga skarpa och stadiga bilder och videosekvenser. Nuförtiden har kameror inbyggda stabiliseringsmekanismer som presterar bra inom detta område. I vissa fall kan det dock krävas extra stabilisering för att uppnå optimala resultat. Under sådana omständigheter är elektroniskt kontrollerade gimbals (sv. kardaner) ett populärt val när det gäller att stabilisera bild i realtid.

Detta projekt har utförts i samarbete med företaget Low Vision International (LVI) som utvecklar och producerar elektroniska synhjälpmedel. Syftet med projektet är att designa, konstruera och utvärdera en gimbal-stabilisator, som kontrollerar rotationen av en kamera.

Resultatet av detta projekt är en gimbal-prototyp med två rotationsaxlar (kallade *roll* och *pitch*). Kamerans rotation regleras genom att en sensor som mäter rotation placeras ovanpå kameran, medan särskilda motorer snurrar kameran på så vis att dess position blir oförändrad i förhållande till en bestämd rotationsreferens.

Prototypen är fullt fungerande, men justeringar måste göras innan den kan användas i LVIs produkter. Det huvudsakliga problemet som kvarstår är att signalerna för kamerans uppmätta rotation har ett stadigvarande brus, vilket gör att kameran blir märkbart instabil vid hög zoom.

Abstract

With today's advancements in camera technology, image-stabilization is a crucial element for capturing sharp and steady images and video. Nowadays, cameras usually have built-in stabilizers that perform well in this task. In some cases, however, an additional stabilizer may be required to achieve optimal results. In such circumstances, electronically controlled gimbals are a popular choice when it comes to real-time image stabilization.

This project has been conducted in collaboration with the company Low Vision International (LVI), which develops and produces electronic visual aid systems. The purpose of the project is to develop, assemble, and evaluate a gimbal stabilizer, which controls the rotation of a camera.

The result of this project is a gimbal prototype with two rotational axes (called *roll* and *pitch*). The camera rotation is regulated by placing rotation sensor on top of the camera, while special motors spin the camera as to keep its position unchanged in relation to a defined rotational reference.

The prototype is fully functional, but some adjustments must be made before it can be used in LVI products. The main problem that remains is that the signals for the measured rotation of the camera has a sustained noise, which makes the camera noticeably unstable during great zoom.

Keywords:

Image-stabilization

BLDC motor

Gimbal

IMU

FOC

Förord

Jag vill rikta ett stort tack till mina kollegor på LVI som gett mig möjligheten att sätta mig in i ett otroligt spännande och utvecklande arbetsområde. Ytterligare ett enormt tack ägnar jag mina handledare Leonardo Colombo och Christoffer Cederberg och examinator Ulf Jeppsson som med sin kunskap hjälpt mig hitta genom otaliga snåriga vägar.

Terminology

Analog-to-Digital Converter (ADC): Converts an analog (positive) voltage to a digital value.

Bias: In this report, bias refers to the addition or subtraction of a set DC voltage to another signal, without altering the shape of the original signal. An unbiased signal has no DC component.

Microcontroller Unit (MCU): Digital programmable controller.

Gyroscope: Electronic sensor that measures rotation.

Accelerometer: Electronic sensor that measures acceleration.

Magnetometer: Electronic sensor that measures the direction of the Earth's magnetic field.

Inertial Measurement Unit (IMU): A combination of a gyroscope, an accelerometer, and a magnetometer.

X Degrees Of Freedom (X-DOF): Refers to the number of measured signals (X) used to determine the orientation of a component. A gyroscope, an accelerometer, and a magnetometer each provide 3-DOF. An IMU containing a gyroscope and accelerometer gives 6-DOF. Adding a magnetometer would give the IMU 9-DOF.

Image-Stabilization (IS): The process of maintaining the position and clarity of a captured image or video, either at the time of capture or during post-processing.

Printed Circuit Board (PCB): Electronic circuit with copper traces printed on a glass fiber slab.

Pulse-Width Modulation (PWM): A method of representing an analog signal using DC-voltage. This is done by changing the so-called duty cycle of the DC-voltage during a certain period of time. The duty cycle can be set to 0-100%.

PID-controller: Calculates how much a signal deviates [error] from a setpoint, which is then corrected using proportional, integrating, and differential terms.

Table of contents

1 Introduction

This chapter describes the fundamentals of the project, as well as theoretical and practical concepts used when designing gimbal prototypes and their features.

1.1 Industrial background

This project has been done at the request of a company called Low Vision International (LVI), which is a company based in Växjö, Sweden. LVI designs visual aid systems for people suffering from visual impairments or hearing loss. One of their products is a *reading camera*, which could simply be described as an all-in-one standalone image projector system. This device, as shown in Figure 1.1, usually consists of a digital camera which is mounted on an adjustable mechanical arm, and aimed at a designated reading area. The image read by the camera is then digitally projected onto a computer monitor, whose purpose is to enlarge and clarify material placed within the bounds of the camera view.

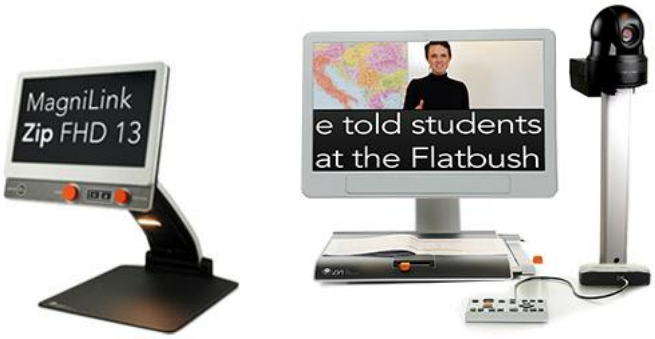

Figure 1.1: Reading camera example products. Source: https://lvi.se/catalog/products/

1.2 Problem definition

The cameras used in the current systems are capable of significant image enlargement (about 90x maximum optical zoom, commonly referred to as *superzoom* in the camera industry) with practically unaffected sharpness. However, image enlargement has a caveat: the more an image is enlarged, it gets increasingly unstable. Instability in image and video, means that the projection tends to look blurry or suffer displacement when the camera is bumped into, shaken, etc. In this thesis, the term s*haking* is used to describe the force which produces instability in projected images.

LVI has posited that image stability is important to their users since this helps the user maintain focus on the text that is being displayed by a reading camera. The company therefore wants to reduce the effects of shaking using a *gimbal*, which should stabilize the reading camera mechanically by controlling its rotation. This concept will be described in more detail in the following sections.

1.3 Image Stabilization

Image stabilization (IS) is commonly defined as the reduction of sudden, and noisy movements of an electronic image. In camera technology, such stabilization may refer to counteracting unwanted angular movement of the camera. This is often done by either panning and/or tilting the camera lens or sensor, or by stabilizing the image using software.

Figure 1.2 shows three methods of optical IS where the goal is to physically rotate the cameras' internal lens and sensor as the camera is being moved. Example *a)* shows the lens and sensor in their resting states, whereas examples *b)* and *c)* show either lenses or sensors can be adjusted for optimal visualization of the object.

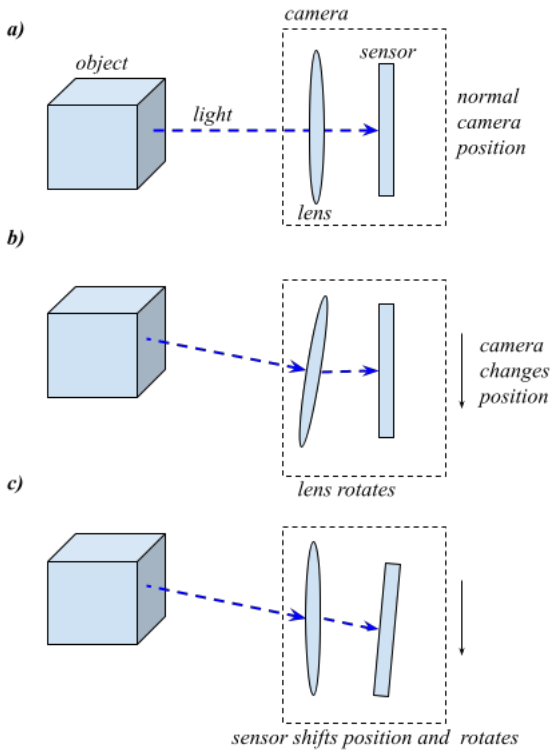

Figure 1.2: Examples of mechanical image stabilization methods.

It is also possible to stabilize images and video using software. This method is called digital (or electronic) IS, since it uses electronic operations or software to achieve what is shown in Figure 1.2 via mechanical action. Although efficient, this process may require extensive processing power, and can also cause loss of image data when dealing with real-time applications. Figure 1.3 illustrates this type of data loss, which appears as black rectangles. This affects the image resolution, since the software alters the image size when parts of the images are removed during the digital IS process.

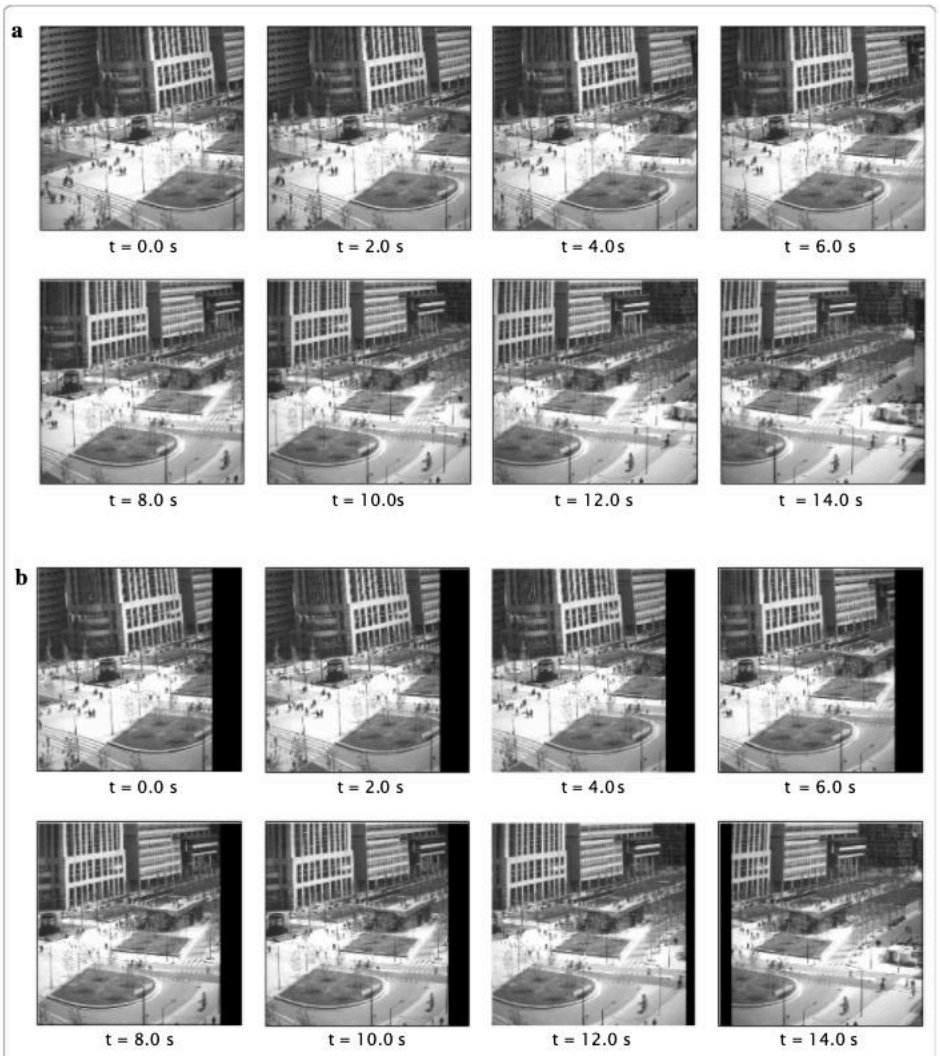

Figure 1.3: Software-based image stabilization [1, p. 14]. Section (a) consists of an un-stabilized image sequence. Section (b) shows the same image sequence with image stabilization enabled. The data loss appears as black rectangles.

1.4 Pre-study

Prior to this project, a pre-study regarding hardware-based image-stabilizer was run by LVI. The purpose of this pre-study was to assess the combination of third-party built-in sensors and a camera integrated into a motorized 3-axis gimbal.

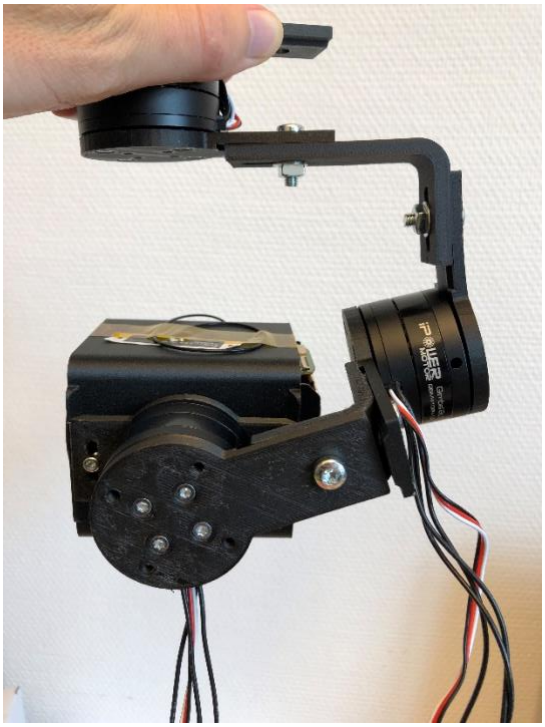

Figure 1.4: Stabilizer prototype from previous pre-study.

The authors of the pre-study built a prototype shown in Figure 1.4 using a gimbal made of 3D-printed components, Brushless Direct Current (BLDC) motors, Inertial Measurement Units (IMU), and a gimbal controller circuit called *SimpleBGC* [2] for controlling motors, measuring rotation, etc. The results, however, were deemed inadequate, since the prototype was unstable, difficult to use and configure, and lacked some desired functions.

The only component from the pre-study to be reused in this project is the camera.

1.5 Purpose

The purpose of this project is to investigate the viability of stabilizing LVI reading cameras using an electronic gimbal. This shall be done by developing an electronic control system which should keep the camera rotationally intact when it suffers shaking. Instead of relying on premade gimbal solutions, a full system shall be designed using readily made components such as motors, driver ICs, microcontrollers, etc. The purpose of this bottom-up approach is to allow for a deeper understanding of the complete system, which enables for low-level tailoring of the system and its functions.

The desired outcome of the project is to produce a working prototype consisting of an electrically motorized gimbal, custom-made mechanical parts, control electronics, and a camera used in LVI systems. The main goal is that the gimbal should project legible text when the camera is being shaken, and also during standstill. As the work progresses, components and functions may be removed or added to the demands if necessary. The prototype should preferably allow for ease of use.

1.5.1 Questions

The following questions are to be answered during the development of this thesis:

- Is an electrically controlled gimbal a viable option of achieving image stabilization for LVIs camera systems?
- Which components are suitable for gimbal image-stabilization?
- What are the limitations of the gimbal solution?

1.6 Demarcation

Although there are efficient software-based methods to stabilize digital video in real time as well as in post processing, this report will not use or focus on any such methods. LVI has done previous research on digital IS, but has concluded that this type of stabilization reduces image-quality as a result of the data loss illustrated in Figure 1.3. Since their customers have high demands on image clarity, it is not an option to use digital IS.

The project does not focus on any optimization of hardware, code, mechanics, etc., as this would be very time-consuming. The resulting prototype is a base that could be improved at later stages, should it be deemed appropriate for use by LVI.

Finally, only two axes of rotation (roll and pitch) shall be implemented for this prototype. The reasons for this are to minimize the complexity and physical size of the prototype.

2 Technical background

This section explains key concepts, components and materials that are used in this project.

2.1 Determining camera position and rotation

A critical part of the prototype is position control, which requires intake and use of positional data. The following subsections describe the methods used to determine the position and rotation of the camera.

2.1.1 Gimbal principle

A gimbal is a support bracket, which lets an object (e.g., a camera) mounted on the innermost gimbal to remain rotationally intact when the gimbals are moved or rotated. A 3-axis gimbal uses three such brackets which are mounted orthogonally to one another, *x, y* and *z* (see Figure 2.1). Rotations around *x*, *y*, and *z* are commonly referred to as *roll*, *pitch*, and *yaw,* respectively.

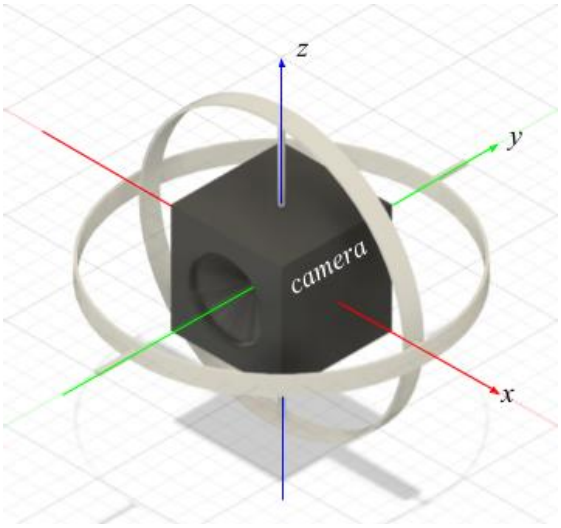

Figure 2.1: Principle of gimbal with internally mounted camera.

By connecting the gimbal brackets to the shaft of an electric motor, the rotation of the camera may be controlled. This allows for counteracting the angular change caused by shaking, as demonstrated in Figure 2.2.

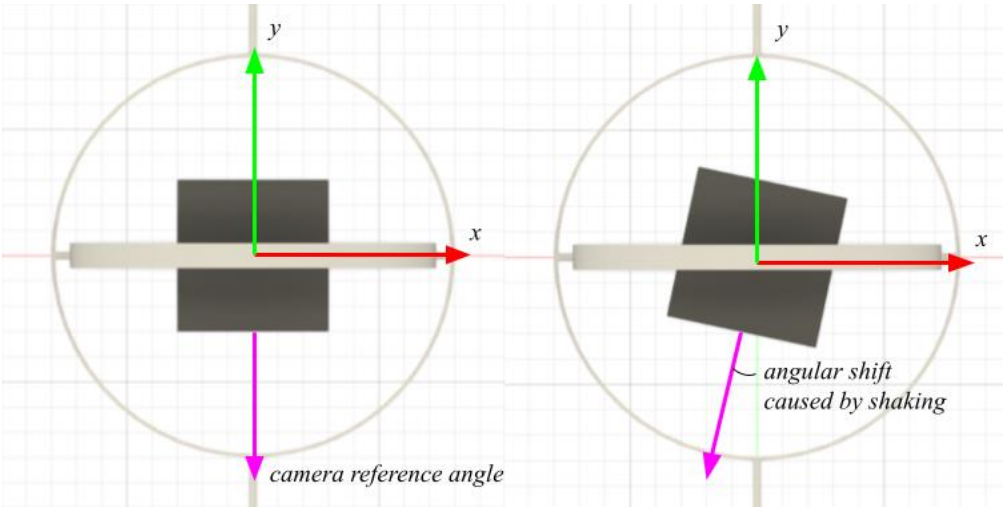

Figure 2.2: Angular shift of camera around its z-axis.

The camera rotation is acquired by an IMU fixed onto a rotating body. In this study, the IMU is placed on a camera. Nowadays, most IMUs are based on MEMS-technology (Micro-ElectroMechanical System). A 9-Degrees Of Freedom (DOF) IMU usually consists of a 3-axis gyroscope sensor, a 3-axis accelerometer, and a 3-axis magnetometer. Respectively, these sensors electronically measure the physical quantities shown in Table 2.1.

Table 2.1: Output units for each sensor type in a 9-DOF IMU.

Sensor type	Measurement unit
Gyroscope	Degrees per second [dps, °/s]
Accelerometer	Meters per second squared [m/s^2]
Magnetometer	Micro Tesla [μT]

In applications such as automobiles and aircrafts, the magnetometer is used to measure the Earth's magnetic field. This allows drones to detect its own rotation relative to the Earth's center.

In combination, the gyroscope and accelerometer provide six degrees of freedom (6-DOF), which means that six variable data points can be used to calculate the camera rotation. If a magnetometer is used, the additional degrees of freedom is added to the system. The raw data acquired from each sensor are not very useful on their own, since they only output at which rate the IMU is rotating or moving. However, when combining the data from the sensors using a method commonly known as *sensor fusion*, the absolute rotation around each axis in 3-dimensional space can be calculated.

2.1.2 Gimbal lock

A common and intuitive way to describe 3D rotation is through the use of so-called *Euler angles*. Euler angles describe the rotation around three orthogonal axes commonly referred to as x-, y- and

z-axes. The rotation around each axis is given in reference to a fixed coordinate system. Although Euler angles pose a simple method of describing rotation, they introduce some computational problems, the most severe of these problems being the *gimbal lock*.

Gimbal lock occurs when one gimbal bracket aligns with another, causing the system to lose one degree of rotational freedom. This issue is illustrated further in Figure 2.3, where each axis *xx*, *yy*, and *zz* represent the rotation axes of each gimbal component. *xx* rotates the inner bracket, *yy* rotates the outer bracket, and *zz* rotates the inner camera block. In part *a)* the gimbal system starts with each axis aligned with the given coordinate system axes. Next, in part *b)* the inner bracket is rotated around the *xx*-axis, which aligns *yy* with *zz*. Rotating the block around the *yy*- or *zz*-axis now yields the same result – the system is thusly in gimbal lock.

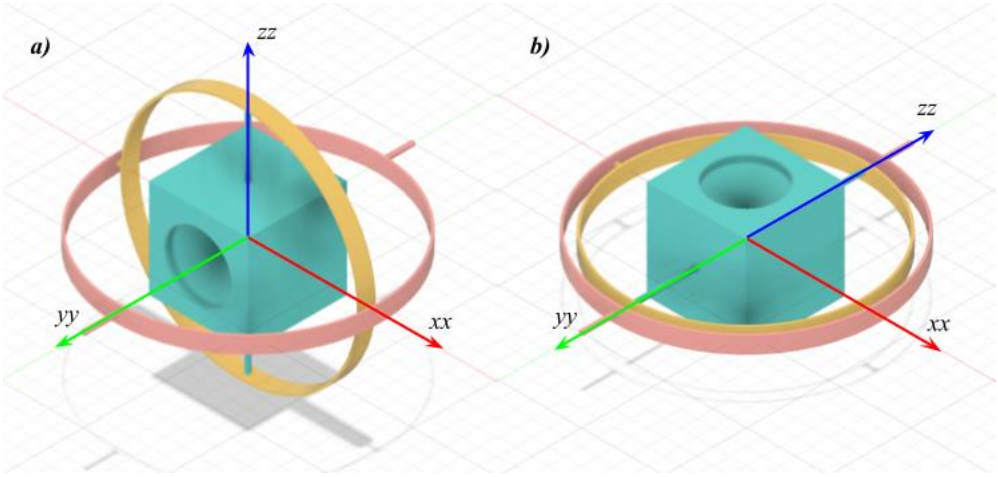

Figure 2.3: Gimbal lock principle.

Avoiding gimbal lock is crucial when designing systems such as gimbal stabilizers. Luckily, there are other methods of representing 3-dimensional rotation that prevent this issue.

2.1.3 Quaternions

A popular method of avoiding gimbal lock is through the use of *quaternions*. A quaternion is a mathematical method of representing spatial rotations using four variables. Since the math that goes into calculating quaternions is deemed outside of the scope of this report, it shall suffice to say that a quaternion is mathematically represented by one real number, and three complex numbers *i, j,* and *k* respectively. A quaternion has the form shown in (1), where q_0, q_1, q_2, and q_3 are real numbers [3].

$$q_0 + q_1 i + q_2 j + q_3 k \tag{1}$$

In this project, the data captured from the IMU will be converted algorithmically into quaternions. A mathematical and programmatic implementation of the *Madgwick-algorithm* (see [4], p. 29),

converts IMU-data, i.e., degrees per second (°/s), meters per second squared (m/s^2), and microteslas (μT) into quaternions.

2.1.4 Motors

In mechatronics and robotics, selecting the right motor for the task at hand is of crucial importance. This section is dedicated to explaining different motor types and their characteristics. This information is used to determine which motor fulfills the project requirements.

The BLDC motor is a 3-phase machine closely related to the synchronous motor. In this device, DC currents which are fed to each phase produce magnetic fields in the electro-magnetic coils inside the motor, which forces the rotor to rotate.

There are many methods of controlling the BLDC-motor, such as sinusoidal and trapezoidal control [5], and Field-Oriented Control (FOC), which will be described further in section 2.2.1. With today's availability of cheap, high-speed microcontrollers FOC is often preferred, despite being computationally expensive.

The BLDC is the motor of choice for this project, since it maintains torque at both low and high speeds, it is lightweight, and it has a relatively small physical volume.

2.2 Controlling the gimbal

Based on the insights, methods, and materials described in section 2.1, the control theory that's relevant to this project shall be presented in this chapter.

2.2.1 Field-Oriented Control (FOC)

Also known as *vector control*, FOC is a method of controlling a two-component current vector acquired from conversion of three-phase currents (e.g., in a BLDC motor). The current vector components are called i_d (direct current) and i_q (quadrature current), which respectively define magnetic flux and torque producing currents. The main goal of FOC when using BLDC motors is to achieve maximal torque. This is done by zeroing the flux reference i_{dref}, while maximizing the torque reference i_{qref}.

In FOC, certain mathematical transformations, called *Clarke* and *Park transformations,* are calculated in order to determine the voltages that will drive the 3-phase motor. When performing the Park transformation, the angle of the BLDC rotor is required. This angle is acquired using a magnetic encoder aligned with a permanent magnet that is attached to the BLDCs rotor shaft.

The block diagram in Figure 2.4 illustrates the controllers and parameters involved in realizing position control with BLDCs and an IMU, which provides a rotational setpoint for the BLDC. Note that this diagram shows the implementation for *one* BLDC and corresponding IMU axis, and that the position control for the IMU has been omitted. In total, the system contains five Proportional-Integrating-Differential-controllers (PID-controllers), including the IMU controller.

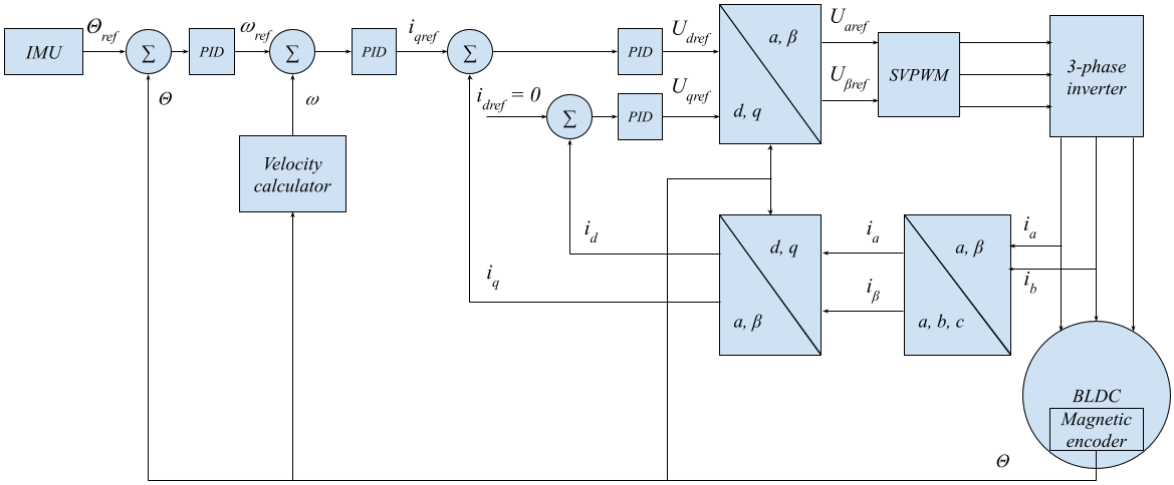

Figure 2.4: IMU-based position control and FOC block diagram.

The procedure of FOC is as follows:

1. Measure phase-currents from BLDC. Only two currents need to be measured, since the third phase-current can be calculated using Kirchhoff's current law.

$$i_a + i_b + i_c = 0 \tag{2}$$

2. Transform the phase-currents to the $\alpha\beta$-reference frame using Clarke-transform.

$$i_\alpha = i_a \tag{3}$$

$$i_\beta = \frac{1}{\sqrt{3}} i_a + \frac{2}{\sqrt{3}} i_b \tag{4}$$

3. Read the rotor position and transform the $\alpha\beta$-components to the dq-reference frame using Park-transform.

$$i_d = i_\alpha cos\ \Theta + i_\beta sin\ \Theta \tag{5}$$

$$i_q = -i_\alpha\ sin\ \Theta + i_\beta cos\ \Theta \tag{6}$$

4. Feed i_d and i_q through PID-regulators and compare with reference setpoint currents i_{dref} and i_{qref}.

5. Transform the currents from the *dq*-reference frame to three phase currents using reverse Park- and Clarke-transform.
6. Calculate Pulse Width Modulation (PWM) duty cycles for three PWM-channels connected to the three phases of a BLDC. The PWM-channels using Space Vector Modulation (SVM or SVPWM) [6].

Once the algorithm has finished, the three PWM-channels output logic-level voltages (e.g., 3.3V or 5V) with their respective duty cycles. These output signals are then fed through a power inverter which transforms the logic-level voltage into a higher voltage, whose value depends on the BLDCs requirements. The BLDC is thusly driven, and the FOC algorithm is repeated.

3 Method

3.1 Hardware requisites

Before selecting materials and components for constructing the prototype, the hardware requirements must be determined. In order to assess the specifications needed for the various components of the system, some initial tests were run.

3.1.1 Measuring shaking speed using optical flow

To get some understanding of how fast the gimbal motors need to run in order to counteract the shaking of the camera, a measurement tool is programmed using the Python-based computer vision library OpenCV [7]. Using this library, edges and corners can be detected continuously, which allows for frame-by-frame estimation of the angular speed of the red dot shown in Figure 3.2. The angular speed can then be converted into RPM (Revolutions Per Minute).

The velocity estimation program [8] is based on acquiring video from the LVI reading camera MagniLink S (MLS). The video captured by the reading camera is simply a dot of ink drawn on a paper. To ease the process of detecting the dot, a contrast filter is applied to the captured video is filtered in [9]. Figure 3.1 reports a dot of ink drawn on a paper which represents the reference for the estimation of the camera angular speed.

Figure 3.1: Left: unfiltered drawing. Right: filtered drawing.

When physically shaking the reading camera, the red dot naturally moves around on the screen. The estimated maximum rotational speeds were around 500 RPM on each axis, as seen in Figure 3.2. This estimated speed was used as a minimum requirement when selecting motors.

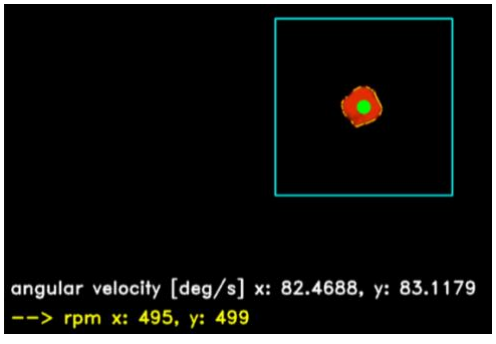

Figure 3.2 Resulting rotational velocity estimation.

The estimation of the angular velocity is calculated using **Error! Reference source not found.**,

$$\omega_s = arctan\left(\frac{v_s}{h_{cam}}\right) \tag{7}$$

where the *s* suffix denotes which axis is measured, ω_s is the angular velocity in [rad/s], v_s is the velocity of the moving red dot [mm/s], and h_{cam} is the height of the camera [mm]. The OpenCV program also prints the angular velocity in Revolutions Per Minute (RPM), which is commonly used to describe the speed of motors. This is useful, since it is important that the motors used for the gimbal can reach the same or higher speeds than the ones it is supposed to counteract. The maximum speeds detected in the program were about 500 RPM for the x- and y-axis. The basis of this calculation is illustrated in **Error! Reference source not found.**.

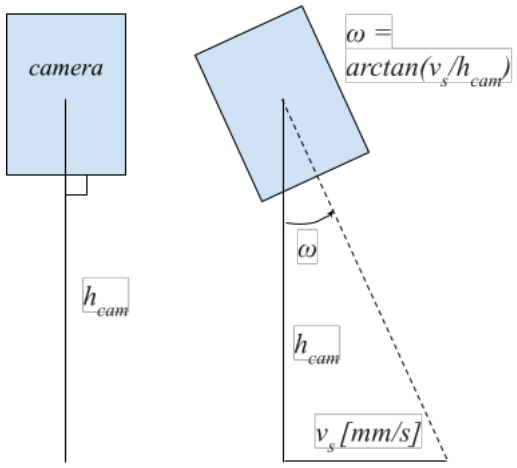

Figure 3.3: Calculating the angular velocity of the camera.

When selecting parts to be used for the gimbal system, there were certain aspects that were taken into consideration. These aspects were namely that the system should maintain:

1. Fast response – If the camera is rotated, the gimbal should counteract the rotation as quickly as possible.
2. Mechanical reliability – The gimbal motors most be able to rotate a certain mass (e.g., the weight of the camera module).
3. Precise measurements and mechanical positioning – When positioning the gimbal motors, the mechanical position should not deviate perceivably from its setpoint.

3.2 Electronic components and procedure

The following components were used to build the control system for the gimbal prototype:

- **Nucleo-F401RE:** A development board based on the high-speed microcontroller STM32F401RE. The MCU contains many useful features such as common communication protocols, a built-in 12-bit Analog-to-Digital Converter (ADC), and General-Purpose Inputs and Outputs (GPIO) with PWM capabilities, etc. [10]
- **DRV8313:** Driver circuit capable of supplying 8-60 V three-phase output using 3 logic-level voltage PWM inputs. 3 motor driver outputs, 3 logic inputs (PWM), 3 low-side current sense outputs, fault and reset logic pins. [11]
- **INA2181(A1):** Dual bidirectional current sensing amplifier. Will be used as a current-sense amplifier for the ADC. A1 = 20V/V gain. [12]
- **MAX6018**: 1.6 V voltage reference generator, used as reference voltage for the current sensor. [13]
- **BMI270:** Accurate 6-DOF IMU with SPI and I^2C-capabilities. [14]
- **AS5048A:** Magnetic encoder with 14-bit SPI output. [15]
- **GBM4108H-120T:** 3-phase BLDC-motor. Maximum speed about 1000 RPM, which is double the speed of the measured shaking speed of 500 RPM.

Using the datasheets provided by the component manufacturers, it is possible to design dedicated libraries, circuits, etc. The components were connected according to the block diagram shown in Figure 3.4. This block diagram shows how each electronic component interact with each other.

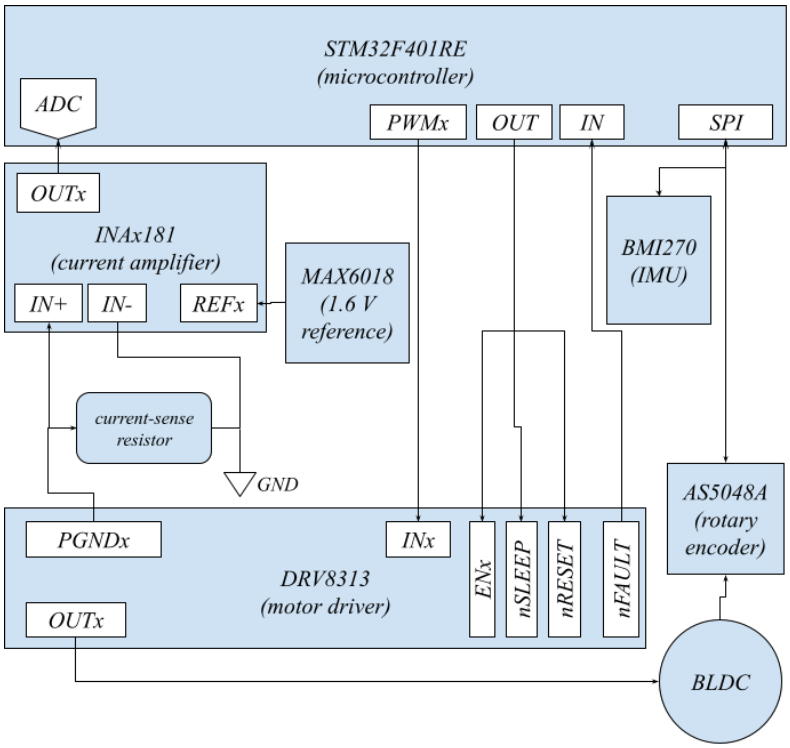

Figure 3.4: Block diagram of electronics in the gimbal system. Each electronic component is represented by the large blue blocks, and their internal inputs and outputs are shown as white blocks.

The procedure of the system electronics shown in the previous figure is described in the flow chart in Figure 3.5.

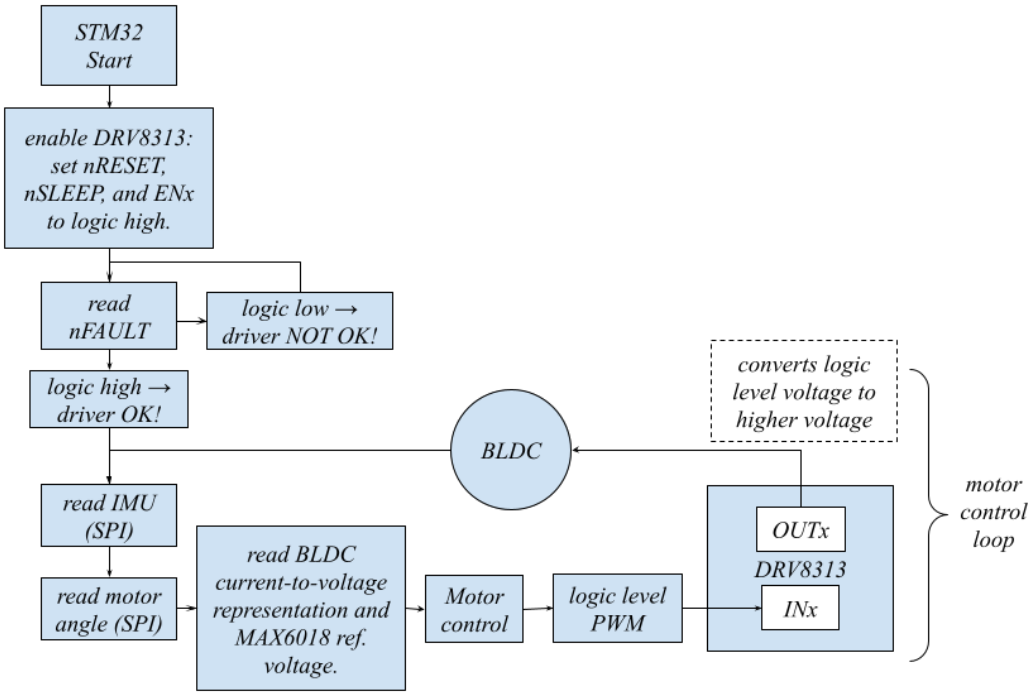

Figure 3.5: Gimbal electronics procedure.

3.3 Software - communication and procedure

By examining their respective datasheets, dedicated libraries were written for each component. The hardware code used to control the gimbal components is coded in C. The code is built, uploaded and debugged using [16].

Some components require digital interfacing through special protocols such as Inter-Integrated Circuit (I^2C) or Serial-Peripheral Interface (SPI). If such interfacing is required, SPI will be preferred since it can transfer data at higher speeds[1], compared to I^2C.

The procedure of the program is illustrated in the block diagram in Figure 3.6.

[1] Of the components used here, the maximum speeds are 10 MHz for SPI and 400 kHz for I^2C.

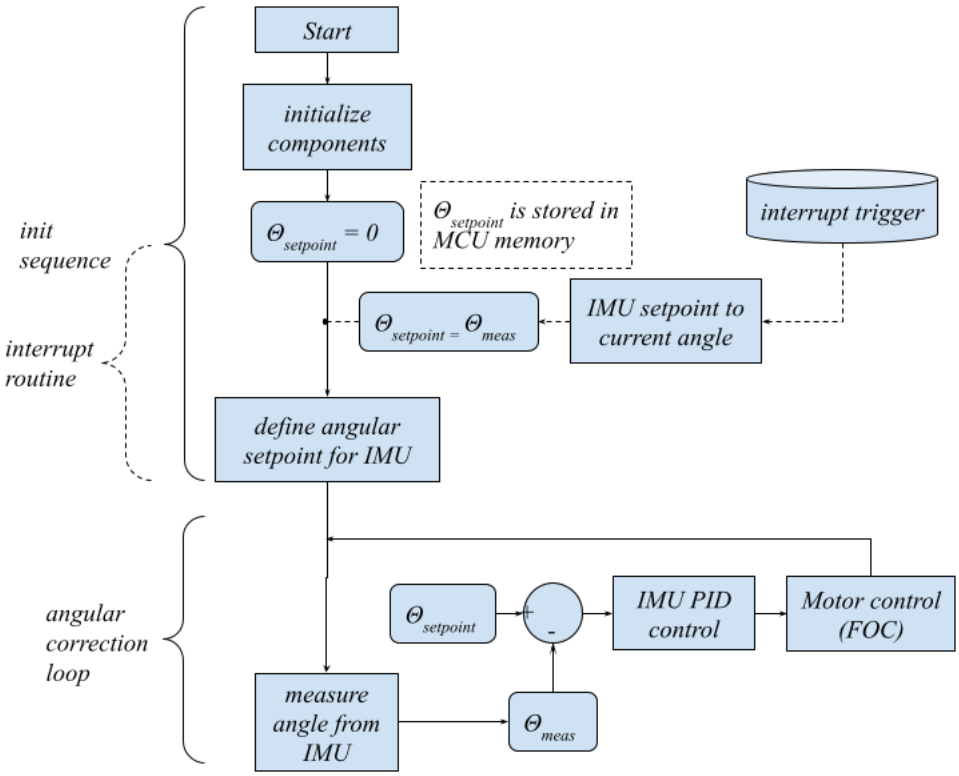

Figure 3.6: Procedure of the gimbal control software.

3.4 Driving the motors using FOC

FOC-control requires measuring the current outputs from each phase of the BLDC. This was accomplished by employing low-side current-sense resistors (0.226 Ω, one for each phase), and measuring the voltage across these resistors. In other words, each phase-current can be represented as *voltage*.

The PGND-pins output voltages ranging between ±500 mV [11, p. 5]. Since STM32 ADC may only read positive voltages, the signal from PGND had to be scaled and adapted to the voltage range of the ADC.

3.5 Sensing and adapting currents for FOC

This section aims to explain how the phase currents from the BLDC is transformed to an ADC-readable voltage, which is used when calculating FOC.

Most microcontrollers have built-in ADCs, which are used to represent analog signals as digital values. Usually, these converters are made to read positive voltages, as is the case with STM32 MCUs whose ADC ranges between 0 – 3.3V. To tackle this problem, the INA2181 current-sense amplifier was used, since it allows the user to set a reference voltage which biases the output signal. When converting bipolar voltage to ADC-readable signals, the reference voltage should be set according to (7) where $V_{sense,ref}$ is the resulting amplifier reference voltage, $V_{in\pm}$ are the bipolar maximum and minimum input voltages, and $V_{ADC,ref}$ is the ADC reference voltage. [Note that this equation takes into account that the bipolar voltage may be biased.]

$$V_{sense,ref} = ((V_{in+}\ mod\ \frac{V_{in+} - V_{in-}}{2})\frac{1}{V_{in+} - V_{in-}} + 1) * \frac{V_{ADC,ref}}{2} \quad (7)$$

Since the signal from the PGND-pins on the BLDC-driver is ±500 mV, thus being *unbiased*, (7) may simply be reduced to (8).

$$V_{sense,ref} = \frac{V_{ADC,ref}}{2} \quad (8)$$

The conversion is illustrated further in Figure 3.7.

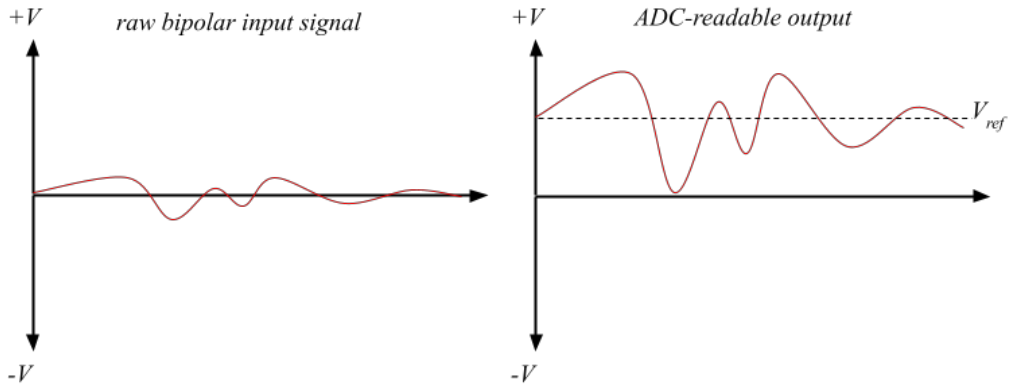

Figure 3.7: Converting bipolar (left) to ADC-readable signal (right).

Since the ADC reference voltage of a STM32 chip is set to 3.3V, $V_{sense,ref}$ = 1.65 V was calculated using equation (8). In the end, the cheapest voltage reference generator that could be found at the time was the MAX6018. This component could only output fixed values, where 1.6 V was deemed acceptable.

The reference voltage, along with the gain of the amplifier, and the current-sense resistance values, can be used as software parameters in the MCU when calculating the *actual* phase-currents of the BLDC.

3.6 Measurement precision

The measurement precision in the IMU and magnetic encoder are determined by their respective bit resolutions. The precisions of the IMU and encoder are given by (9) where Θ is the angular error in degrees per least significant bit (LSB), and b is the bit resolution of the component. The precision of each component is listed in Table 3.1.

$$\theta_b = \frac{360}{2^b} \tag{9}$$

Table 3.1: Resolution and precision for the IMU and magnetic encoder.

Component	Resolution [bits]	Precision [degrees/LSB]
BMI270	16	0.00549
AS5048 (PWM)	14	0.02197
AS5048 (SPI)	12	0.08789

The resolution of the IMU is crucial, since it determines how precisely the camera can be positioned. An IMU with high resolution will produce small positioning errors, whereas low resolution produces larger errors which are further exaggerated when zooming the camera. The total error projected on-screen is given by (10), where h_{cam} is the camera height, $w_{err,proj}$ is the on-screen error (measured physically), and z_{cam} is the camera zoom. The use of the equation is illustrated in Figure 3.8, where it should be noted that the resolution of the encoder affects the positioning error.

$$w_{err,proj} = z_{cam} \cdot h_{cam} \cdot tan\frac{360}{2^b} \tag{10}$$

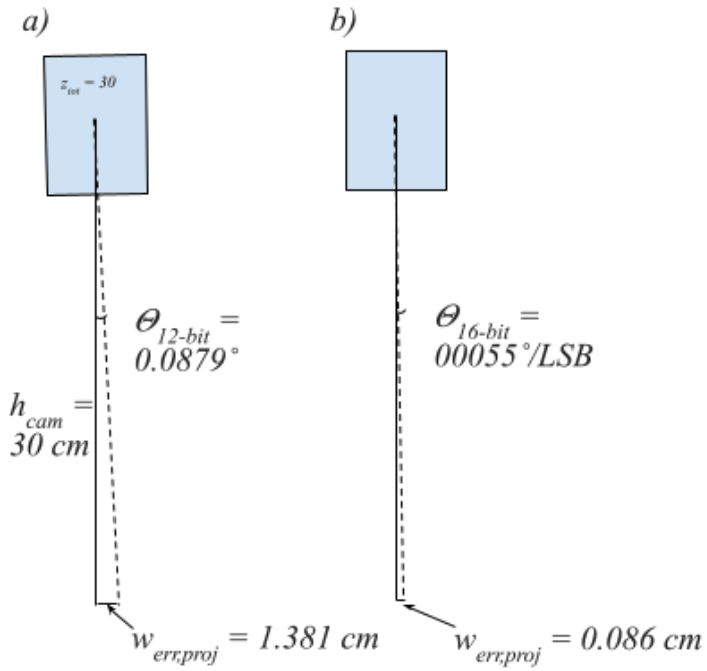

Figure 3.8: IMU resolution effects on the precision of position control.

3.7 Filtering and PID-tuning

As previously mentioned in section 2.2.1, each BLDC motor employs a total of five PID-controllers (including the IMU controller). Each PID-controller is setup according to Figure 3.9, where *r* is the input signal, and *u* is the output signal.

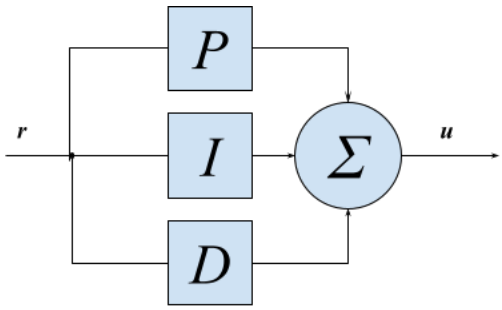

Figure 3.9: PID-controller setup.

Each controller is fed an input signal i.e., quadrature and direct current, motor velocity and angle, and the IMUs rotation around its corresponding motor axis.

The gimbal system is constantly taking measurements at very high sampling rates, which produces high-frequency noise on the measured channel. Noise may affect the performance of the gimbal, causing the camera to jitter as it strives to reach a rotational setpoint. However, such noise can be reduced by utilizing a digital low-pass filter (LPF), which attenuates frequencies higher than a set cutoff frequency. The main reason for using digital filters is that they are simple to configure and

requires no external components. After some testing, the camera jitter that was produced by the noise had been reduced as a result of utilizing the LPFs.

Once the measured signals are satisfactorily filtered, PID-tuning commences. The parameters are set experimentally through trial-and-error to find settings for each of the two motors. The goal was to set the PID-parameters to make the gimbal respond as fast as possible to changes in rotation, in order to counteract angular disturbances on the camera.

The effects of the filtering and PID-tuning is evaluated by examining the stability of the projected image when the camera has maximal zoom, and by monitoring data in the Serial Wire Viewer Data Trace Timeline Graph available in the IDE's debug mode [16].

4 Results

This chapter presents the physical construction as well as the functions of the individual parts of the gimbal prototype. Furthermore, the chapter examines the measurements acquired during evaluation of the prototype.

A key aspect in this chapter is how well stability is maintained when the gimbal has reached *steady-state* on its axes, which is the state where the gimbal has settled (close) to its rotational setpoints, making the camera stationary.

As the block diagram in Figure 3.4 shows, the prototype was constructed by assembling interconnectable discrete components, called *breakout boards*, such that each component could be developed and tested individually. A very useful feature of this technique is that once a component worked as intended, more of the same component may be added to the system if needed. Finally, the parts can be consolidated onto a single-purpose PCB, which contains all functions required for measurement, control, etc.

4.1 Inertial Measurement Unit (IMU)

A breakout board (MPN: MIKROE-4073) containing the *BMI270* IMU was used. Data from the IMU is sent to the microcontroller using SPI.

By placing the IMU on a camera, its rotation could be visualized using a Python-based visualizer [17]. The visualizer was used to show the effects of converting the IMU data to quaternions using Madgwick's algorithm. Figure 4.1 reports a 3D-block mimicking the rotation of the IMU.

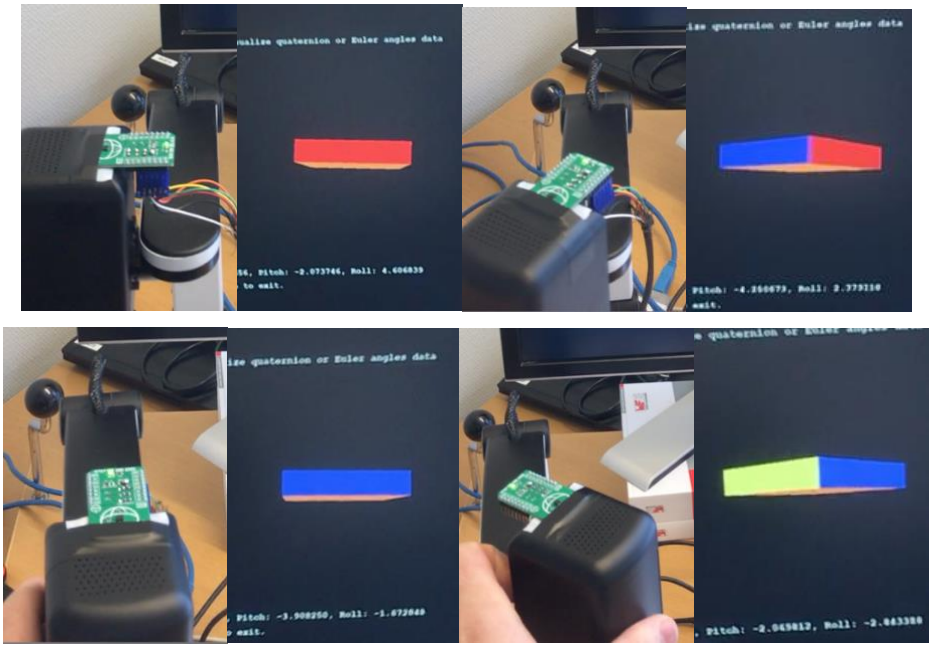

Figure 4.1: Rotating a 3D-model using an IMU.

It is observed, however, that the yaw drifts over time (Figure 4.2). Previous studies have concluded that this drift is a result of gyroscope error [18, 6]. A possible explanation: As gyroscope data is

converted to quaternions, sudden noise produces a constantly growing offset to the yaw rotation. This could become problematic when controlling the yaw of the gimbal, but can be solved by adding a magnetometer either separately to the current 6-DOF IMU, or by replacing the current IMU with a 9-DOF IMU. In this case, the yaw axis shall simply be omitted, since it would further complicate this project.

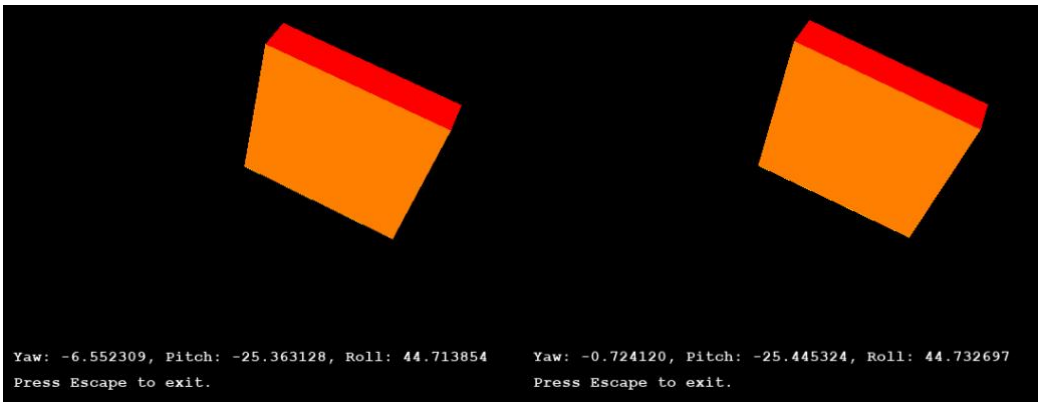

Figure 4.2: Left: Initial yaw is -6.552°. Right: After just a few seconds, the yaw drifts to -0.724°.

4.2 Magnetic encoder

Initially, the magnetic encoder *AS5048A* was pre-attached inside the BLDC out of the box, as seen in Figure 4.3. The chip was factory set to output 12-bit PWM encoder data, but was instead reconfigured by the author to output the encoder data in 14-bit SPI. This allowed for increased angular precision, which is demonstrated further in the following section.

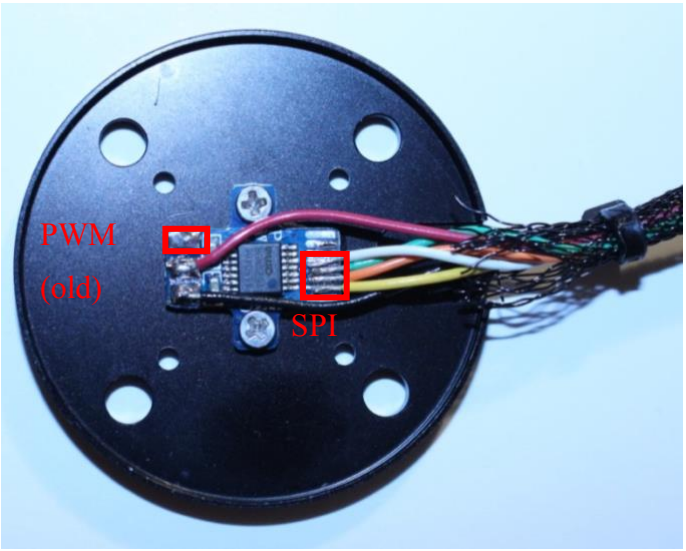

Figure 4.3: An AS5048A magnetic encoder (with third-party PCB) mounted on the backplate of a BLDC.

The encoder PCB proved to be mechanically unstable: the solder pads were small, and therefore the joints were hard to configure and easily broke off or short-circuited with other contacts. Since the encoders are fundamental for controlling the BLDCs, it is essential that the encoders are mechanically stable. Therefore, a new PCB was designed (Figure 4.4). The solder joints (and encoder) were wrapped in shrink tubing to further improve the joints' mechanical resilience.

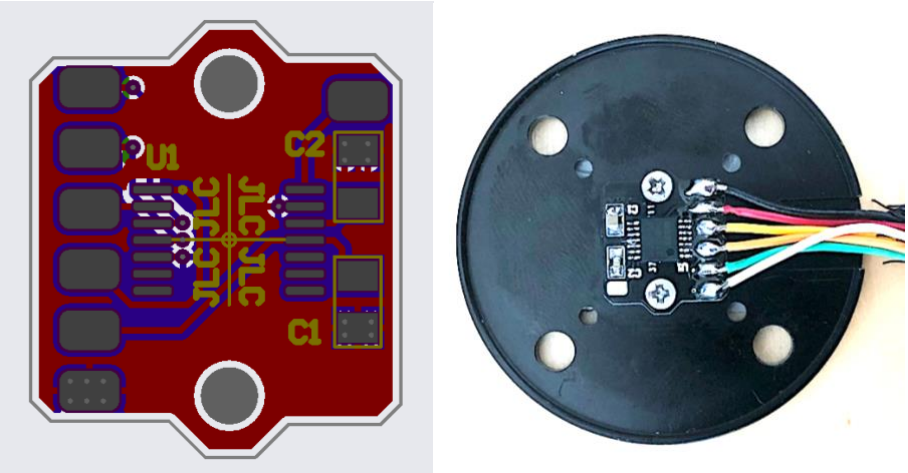

Figure 4.4: New AS5048A PCB with larger solder pads.

When rotating the BLDC, the encoder output data is registered by the microcontroller. Since only one motor is connected to the MCU, a single angle is shown under the "MotorX"-label. The angle is measured with reference to a zero-position, and then printed on-screen through a command prompt, as demonstrated in Figure 4.5.

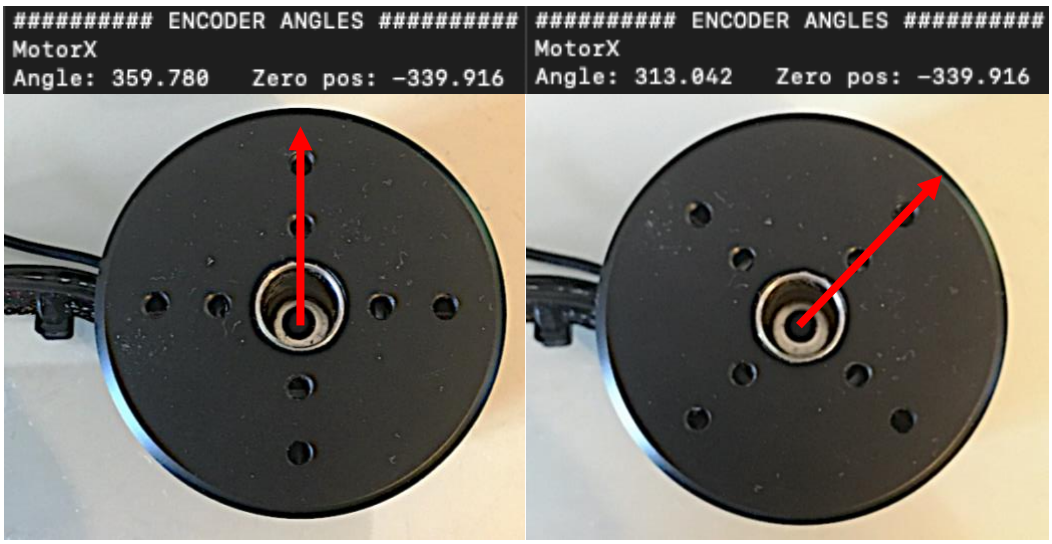

Figure 4.5: Angular output from the magnetic encoder shows the rotation of the BLDC.

4.3 Motor driver

A breakout board for the *DRV8313* motor driver was designed, and is presented in Figure 4.6. See Figure A.2 in the Appendix for the full schematic of this circuit.

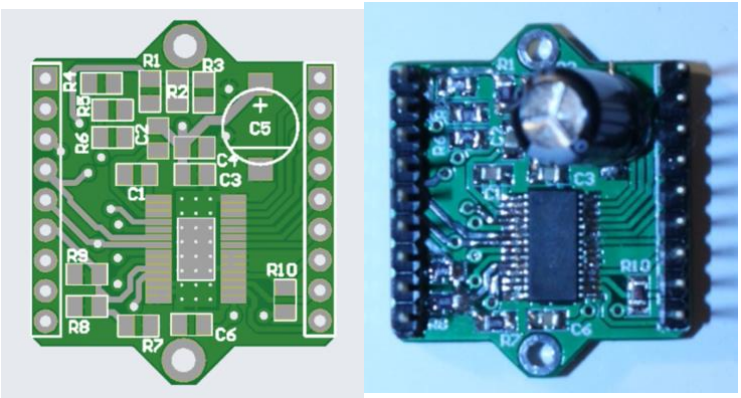

Figure 4.6: DRV8313 breakout board (top-view). Left: PCB layout. Right: Manufactured PCB with soldered components.

The procedure of using the DRV8313 with a connected BLDC is as follows:

- Enable the chip by setting nRESET and nSLEEP inactive HIGH
- Set each ENx-pin to active HIGH, which allows for converting PWM-input signals from logic-level to high-voltage (8-60 V) output.
- Input PWM signals on the INx-pins to run the motor.

Figure 4.7 illustrates the three-phase voltages that are output to a BLDC-motor. Each row in the picture shows the on-time of each phase, which is a result of the calculations that are included in the FOC-algorithm.

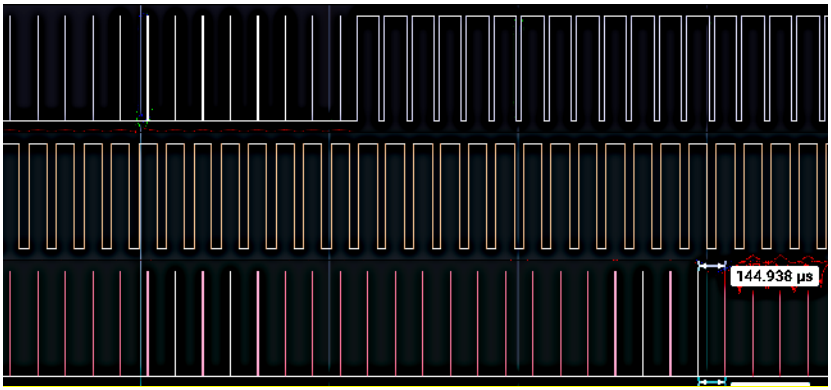

Figure 4.7: Logic-level 3-phase PWM test signals being sent to the DRV8313, measured with generic logic analyzer [19].

4.4 Current amplifier

A breakout board for the current amplifieris designed, and is presented in Figure 4.8. See Figure A.3 in the Appendix for the full schematic of this circuit.

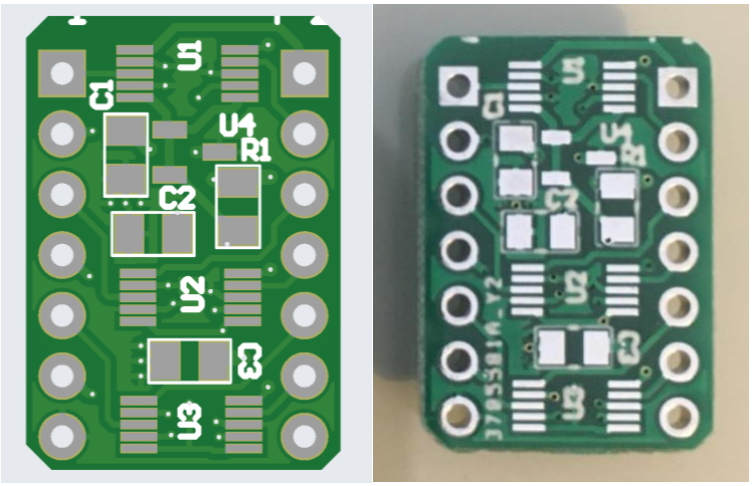

Figure 4.8: INA2181 breakout board (top-view). Left: PCB layout. Right: Manufactured PCB without soldered components.

4.5 Gimbal driver shield

After the features of each breakout board were tested and confirmed to be functioning as intended, they were now ready to be placed on a single board, a so-called *shield*, which is a term for PCBs that can be connected on top of another PCB. The gimbal driver shield is presented in Figure 4.9, and contains motor drivers, current amplifiers, and connections to the IMU and magnetic encoders, and will be connected to the Nucleo board through the connectors CN7 and CN10 shown in the figure. A schematic overview of this circuit is presented in Figure A.1 (Appendix).

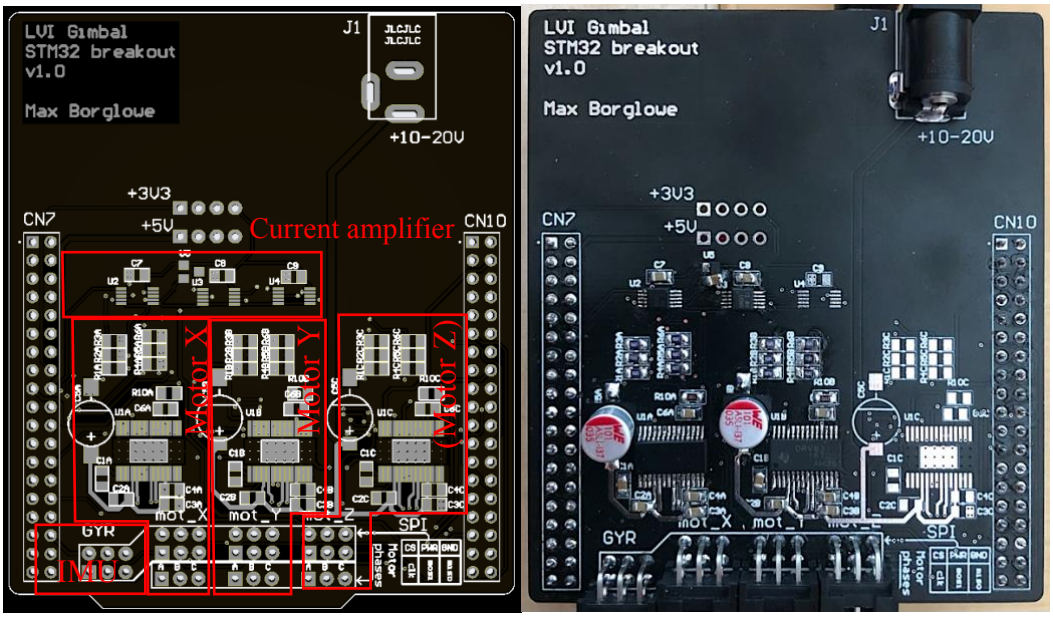

Figure 4.9: Gimbal driver shield with combined breakout board components. PCB layout (top-view). Right: Manufactured PCB with soldered components. The Motor Z components are not in place.

4.6 Prototype setup

Throughout the designing process, it was decided to only use two BLDC motors. The reasons for this are the following:

- Converting IMU data into quaternions accumulates yaw rotation over time when using an IMU without a magnetometer, as stated in section 4.1.
- The control system for one BLDC contains five PID-controllers, as mentioned in section 2.2.1. This makes parameter setup for each controller quite complex, since adjusting one controller may affect the performance of another.

The final assembled gimbal prototype is presented in Figure 4.10. The prototype is subject to evaluation, and the results are presented in the following sections.

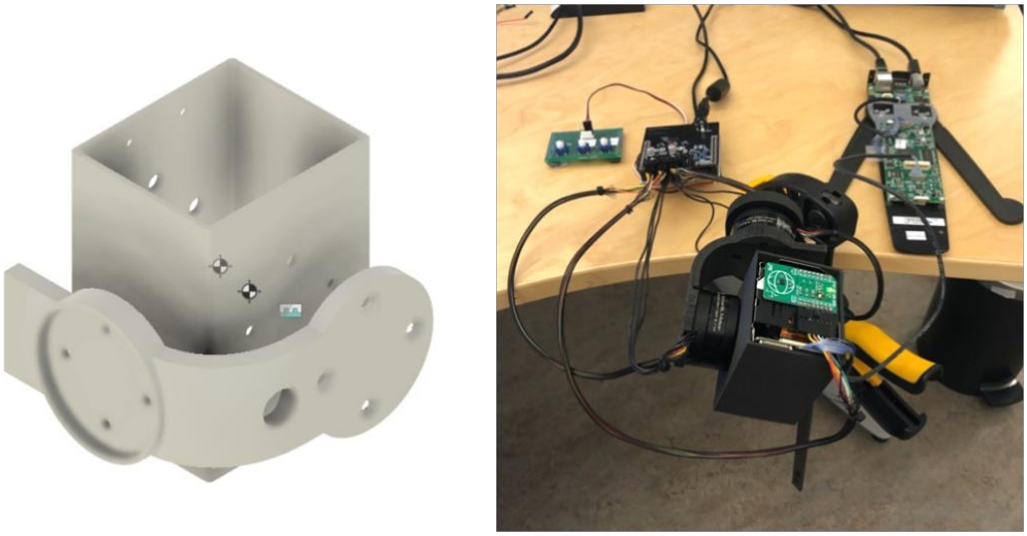

Figure 4.10: Testing assembly using an IMU and two BLDCs. Left: 3D-sketch of the camera housing, and a gimbal bracket. Right: Fully assembled gimbal prototype.

4.7 Filtering and PID-tuning results

Using the methods stated in section 3.7, the parameters in Table 4.1 lead to a significant noise reduction on the measured signals, which consequently improved camera stability during steady-state. The filtering comes with a tradeoff: since high-frequency material is being filtered out, the gimbal is not as responsive.

Table 4.1: Filter cutoff frequencies for each motor controller.

Controller signal to be filtered	Motor roll cutoff frequency [Hz]	Motor pitch cutoff frecuency [Hz]
IMU	2.5	2.5
Motor angle	2.86	2.86
Motor velocity	3.33	3.33
Direct current (i_d)	10	10
Quadrature current(i_q)	10	10

Table 4.2: PID settings for each motor controller.

Controller	Motor roll PID-parameters [K_p; K_i; K_d]	Motor pitch PID-parameters [K_p; K_i; K_d]
IMU	7.89; 6.57; 0.51	6.5; 5.0; 0.63
Motor angle	2.0; 0.0; 0.0	2.0; 0.0; 0.0
Motor velocity	0.75; 0.0; 0.0	0.75; 0.0; 0.0
i_d & i_q	1.25; 0.001; 0.0	1.6; 0.001; 0.0

In addition to filtering, the PID-parameters were then tuned according to Table 4.2. The LPF reduced high-frequency noise, while the PID-controller improved the gimbal's ability to reach its rotational setpoints, as can be seen in Figure 4.12. During these measurements, a step function changes the roll and pitch setpoints from -45˚ to 0˚, where Figure 4.11 reports the physical rotation of the camera.

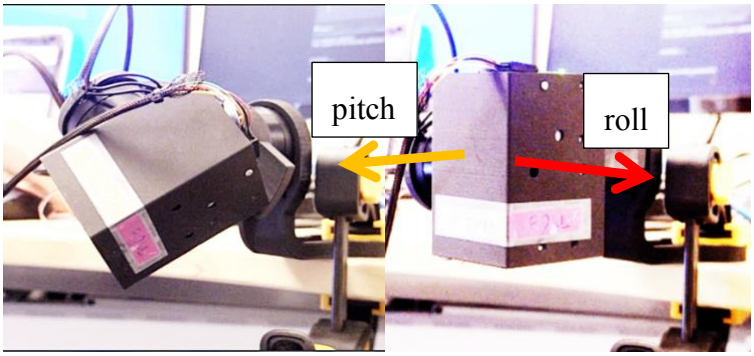

Figure 4.11: The camera being positioned by the gimbal to defined setpoints. Left: Roll and pitch setpoints are -45˚. Right: Roll and pitch setpoints are 0˚. The arrows indicate the rotational axes.

Continuing, Figure 4.12 demonstrates the effects of applying filter- and PID-parameters on different signals, showing the camera roll (red) and pitch (yellow). These graphs show the measured roll and pitch signals where (a) and (b) shows the effects of turning the LPFs and PID-controllers off and on respectively.

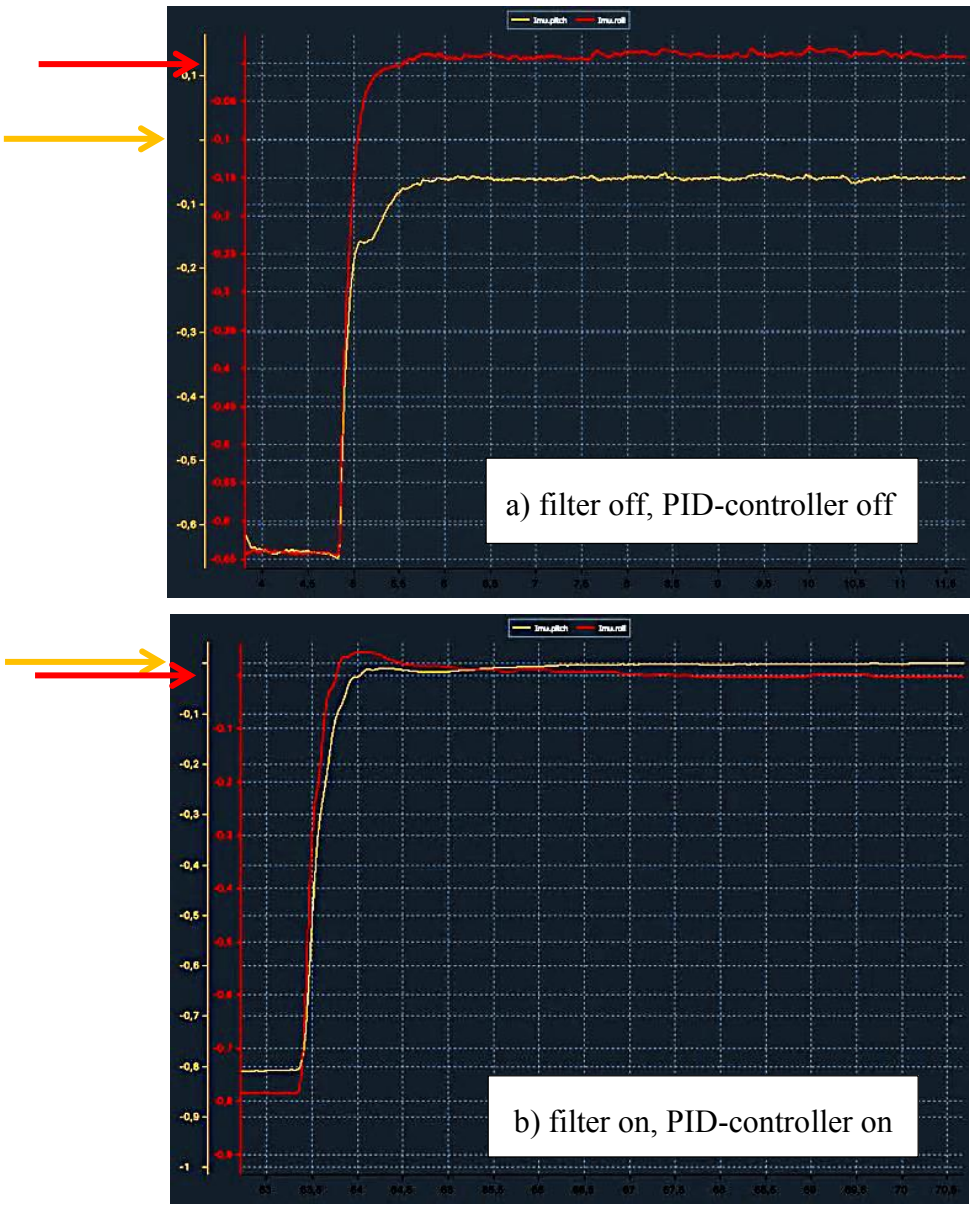

Figure 4.12: Graphs showing the effects turning digital LPFs and PID-controllers on/off. The red and yellow lines represent the measured roll and pitch respectively, while the red and yellow arrows indicate the setpoints for these signals.
Vertical: IMU values in radians. Horizontal: Elapsed time [0.5s/div].

The main observation from Figure 4.12 is that filtering removes noise to some extent, while properly configured PID-regulation positions the camera very close to its rotational setpoints (0° for each axis).

Although the noise in the signal had been attenuated in part b), compared to part a) in Figure 4.12, the projected image appeared unstable during maximum zoom. When shaking the camera, the gimbal did not respond fast enough to prevent angular disturbances from affecting the camera rotation.

4.8 Noise

After filtering and regulation, a minute steady-state noise remained on the IMU measurement signals, which resulted in a consistently shaky image. The steady-state noise cannot be seen in Figure 4.12, as its amplitude is too low to be visualized at the given scale. A close-up of the steady-state noise is shown in Figure 4.13, which in this snapshot ranges from -0.15˚ to 0.23˚ on the roll axis, and -0.1˚ to 0.09˚ on the pitch axis.

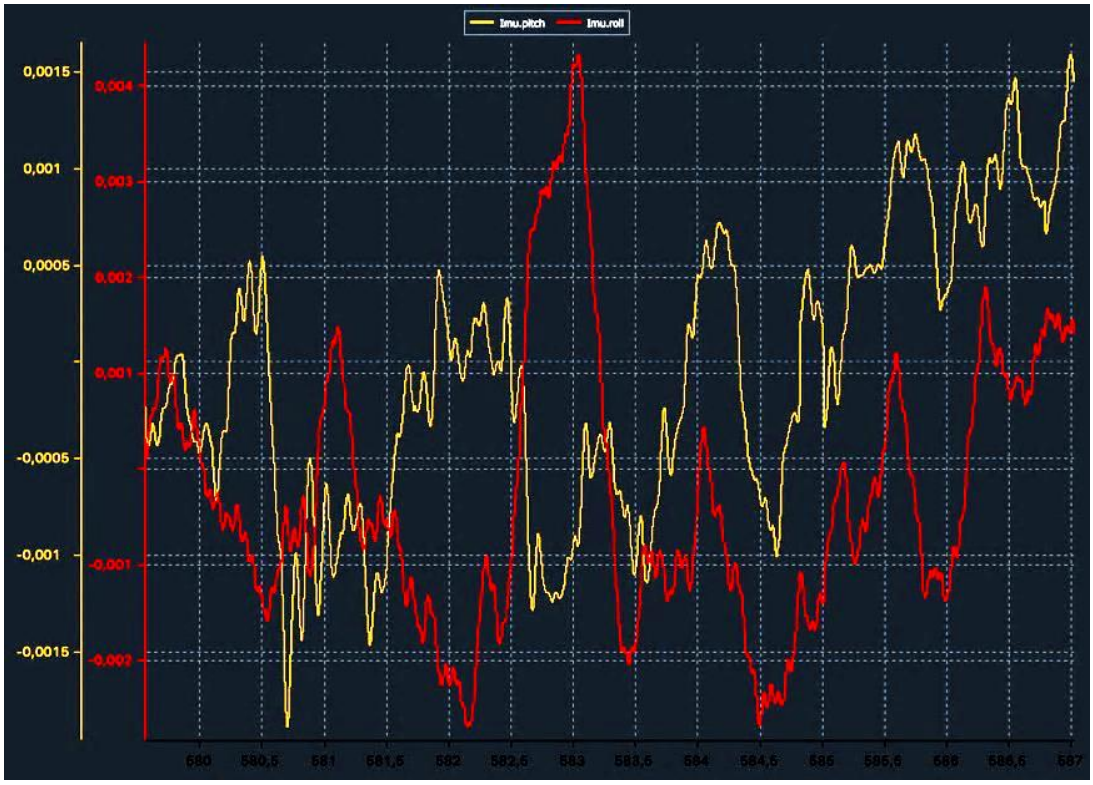

Figure 4.13: Steady-state noise on IMU roll (red) and pitch (yellow). Vertical: IMU values in radians. Horizontal: Elapsed time [0.5s/div].

This steady-state noise may have multiple sources:

- Slow updating of FOC-algorithm may cause the BLDCs to induce mechanical noise in the system.
- The relatively low resolution of the encoders and IMU impedes the gimbal's ability to precisely set the position of the BLDCs and determine the IMU orientation.

Figure 4.14 demonstrates an image sequence of how the steady-state instability appears in the projected video. These images were captured with a camera height of 124.5 cm from the actual image, and 13.4x zoom applied. Using (10), it was found that the figure had a pitch error of 0.028˚, which resulted in a remarkable offset of the original image position shown in the left part of the figure.

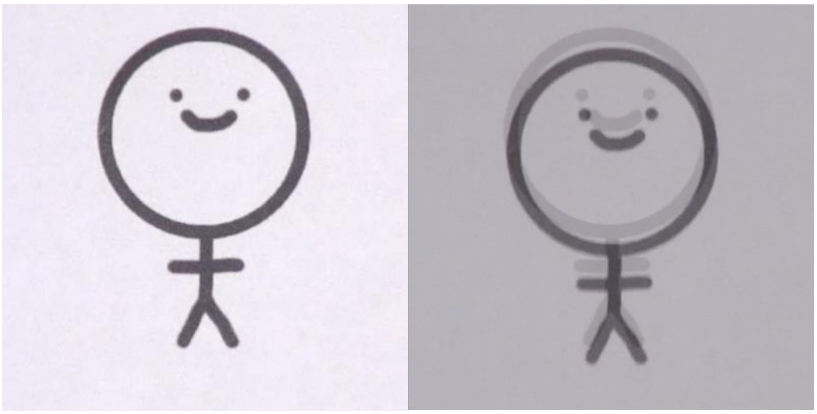

Figure 4.14: Projected video sequence from gimbal mounted camera. Left: Original image position with gimbal off. Right: Peak image offset when gimbal is turned on.

5 Conclusions

- As a result of this project, a fully functional gimbal prototype was developed and evaluated. The main function of the gimbal was to maintain the camera orientation. The gimbal performed well in this task at low speeds, such as tilting the gimbal by hand, but had issues compensating for the fast change in rotation produced by shaking. The main cause for this may be that the BLDC motors' positions are not updated fast enough. This could, in turn, depend on a multitude of other factors, e.g., slow computing, low resolution measurement, camera weight, flawed motors and other mechanics, non-optimal filter- and PID-settings, etc.
- The primary flaw of the prototype is that it does not maintain stability in steady-state, which makes the image *more* unstable when the gimbal function is turned on, compared to when it's turned off. Changing the motor encoders and IMU for alternatives with higher resolutions could decrease the magnitude of measurement errors, which could improve (at least) the steady-state stability. It should be noted that the perceived instability of projected images grows larger the more you zoom in on an image. In the case of Figure 4.14, 13.4x zoom was used, which made the image shifting much more apparent compared to when the camera had no zoom applied.
- Despite the issues that have been observed during the project, LVI is confident that the gimbal solution will be the way to go when it comes to image-stabilization. It should be noted though, that the current state of the prototype is not viable for use in the products of LVI.
- While filtering measured signals reduced noise in the system, it also impedes the gimbal's ability to counteract high-frequency rotation. This means that the cutoff frequency of the applied low-pass filters must be chosen with care, so as not to remove important high-frequency data. There may also be other filtering methods could be applicable for the purposes of this study.

5.1 Future work

- Should further research be conducted on the subject, it would be interesting to see if replacing the current electronics with better alternatives will improve the stability. Switching the MCU for an even faster one (there are user-friendly STM32 alternatives that have clock-speeds up to about 200 MHz) would make the FOC-algorithm more efficient and precise, which could eliminate noise and improve the response time of the gimbal. Another possible improvement is optimizing the FOC-code (and other code as well), which could also provide faster to the response time.
- There are mechanical improvements that could be made to the prototype, such as driving each gimbal axis with geared motors, instead of driving them directly as was done in this project.
- The omission of the yaw-axis was done on the basis of the proposed yaw drift in IMUs without magnetometers, and reducing the system complexity. This may or may not be beneficial to the system stability, but should nonetheless be investigated.
- Although the gimbal prototype was not sufficiently stable, it could act as a base for future studies around the subject of using gimbal stabilizers for LVI products. Another future use for the prototype is to extract some of its functions in the development of a Pan-Tilt-Zoom (PTZ) camera, which is a simple two-axis camera controller that allows for manual rotation

and zoom. Some testing was done using the prototype as a PTZ system, by bypassing the IMU and its functions. This resulted in much more stable images, since IMU noise was no longer a factor – so this usage may be possible with the components used currently.

6 Bibliography

[1] S. Raut, K. Shimasaki, S. Singh, T. Takaki and I. Ishii, "Real-time high-resolution video stabilization using high-frame-rate jitter sensing," Springer Nature, 2019.

[2] BaseCam, “BaseCam SimpleBGC 32-bit Tiny Rev. C,” BaseCam, 6 June 2022. [Online]. Available: https://www.basecamelectronics.com/sbgc32tiny_rev_c/. [Accessed 6 June 2022].

[3] S. Särkkä, "Notes on Quaternions," 2007. [Online]. Available: https://users.aalto.fi/~ssarkka/pub/quat.pdf. [Accessed 7 June 2022].

[4] S. Madgwick, "An efficient orientation filter for inertial and inertial/magnetic sensor arrays," 2010.

[5] M. Hein, "Demystifying BLDC motor commutation: Trap, Sine, & FOC," 2020. [Online]. Available: https://www.ti.com/lit/ml/slyp711/slyp711.pdf?ts=1647877504170&ref_url=https%253A%252F%252Fwww.google.se%252F.

[6] O. Uçak and S. Yilmaz, "Implementation of Space Vector PWM In FOC of a Servo Motor Drive," Eleco, 2015.

[7] OpenCV team, "OpenCV," 2021. [Online]. Available: https://opencv.org/.

[8] M. Borglowe, "OpenCV Velocity Estimator," 2021. [Online]. Available: https://github.com/maxborglowe/opencv_rotational_velocity_estimator. [Accessed 23 March 2022].

[9] Low Vision International, "MagniLink MacViewer," October 2021. [Online]. Available: https://lvi.se/catalog/products/magnilink-macviewer. [Accessed April 2022].

[10] STMicroelectronics, "STM32F401xD STM32F401xE," 22 January 2015. [Online]. Available: https://www.st.com/resource/en/datasheet/stm32f401re.pdf. [Accessed June 2022].

[11] Texas Instruments, *DRV8313 2.5-A Triple 1/2-H Bridge Driver,* 2016.

[12] Texas Instruments, *Low-Cost Bidirectional Current Sensing Using INA181,* 2017.

[13] Maxim Integrated Products, Inc., "Precision, Micropower, 1.8V Supply, Low-Dropout, SOT23 Voltage Reference," 2015. [Online]. Available: https://datasheets.maximintegrated.com/en/ds/MAX6018-MAX6018B.pdf. [Accessed 7 June 2022].

[14] Bosch Sensortec, "BMI270 6-axis, smart, low-power Inertial Measurement Unit for high-performance applications," November 2020. [Online]. Available: https://www.bosch-sensortec.com/media/boschsensortec/downloads/datasheets/bst-bmi270-ds000.pdf. [Accessed 7 June 2022].

[15] AMS, "AS5048A/AS5048B Magnetic Rotary Encoder (14-Bit Angular Position Sensor)," 29 Jan 2018. [Online]. Available: https://ams.com/documents/20143/36005/AS5048_DS000298_4-00.pdf/910aef1f-6cd3-cbda-9d09-41f152104832. [Accessed 7 June 2022].

[16] STMicroelectronics, "Integrated Development Environment for STM32," November 2021. [Online]. Available: https://www.st.com/en/development-tools/stm32cubeide.html. [Accessed 7 June 2022].

[17] N. Elkunchwar and K. Apostolopoulos, "PyTeapot Quaternion Euler cube rotation," 2021. [Online]. Available: https://github.com/thecountoftuscany/PyTeapot-Quaternion-Euler-cube-rotation. [Accessed 2021].

[18] G. Shi, X. Li and Z. Jiang, "An Improved Yaw Estimation Algorithm for Land Vehicles Using MARG Sensors," MDPI, Basel, 2018.

[19] Saleae, "Logic 2 Software," 2021. [Online]. Available: https://support.saleae.com/logic-software/sw-download. [Accessed 7 June 2022].

[20] M. Gustavi and L. Andersson, "Implementation of control algorithm for mechanical image stabilization," Lund University, Lund, 2017.

[21] Axis Communications, "Image stabilization - improving camera usability," 2014.

[22] N. Bevara and A. Jagtap, *Low-Cost Bidirectional Current Sensing Using INA181,* Dallas, Texas: Texas Instruments, 2017.

[23] G. Günastı, "Quaternions Algebra, Their Applications in Rotations and Beyond Quaternions," 2016.

[24] D. Hudgins, *Low-Side Current Sense Circuit Integration,* Dallas, Texas: Texas Instruments, 2017.

Appendix

The schematic below is an overview of the full circuit, i.e., the gimbal shield that is used to drive the BLDC motors. The green areas link to other schematics, namely the Nucleo (microcontroller) pins, motor driver (DRV8313), and the current amplifier (INA2181), which are too detailed to involve in this overview.

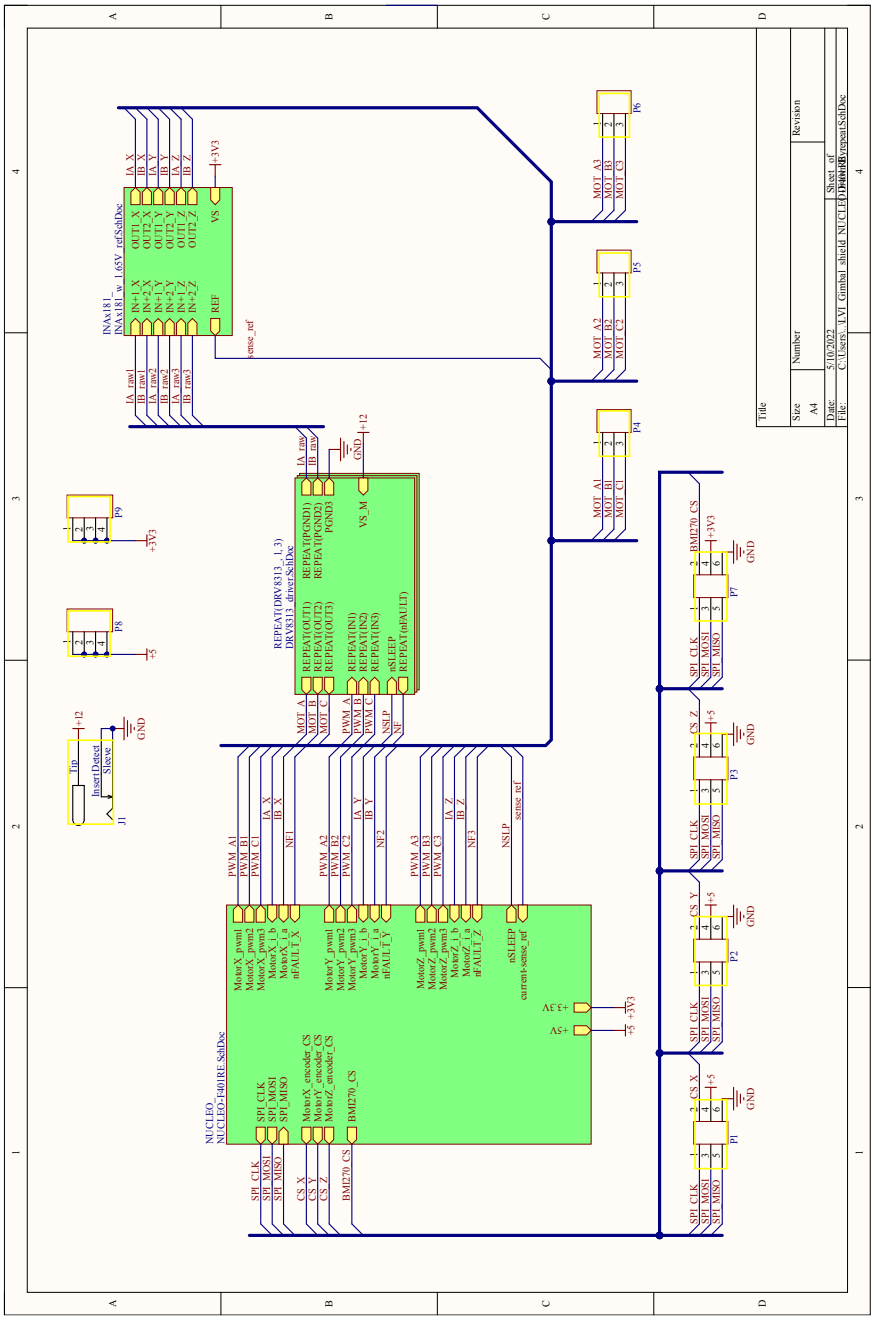

Figure A.1: Schematic of gimbal driver shield, containing motor drivers, current amplifiers, and pinouts to the microcontroller.

The schematic below contains the motor driver DRV8313, which converts logic-level signals to a higher voltage, which is used to drive the BLDC motors.

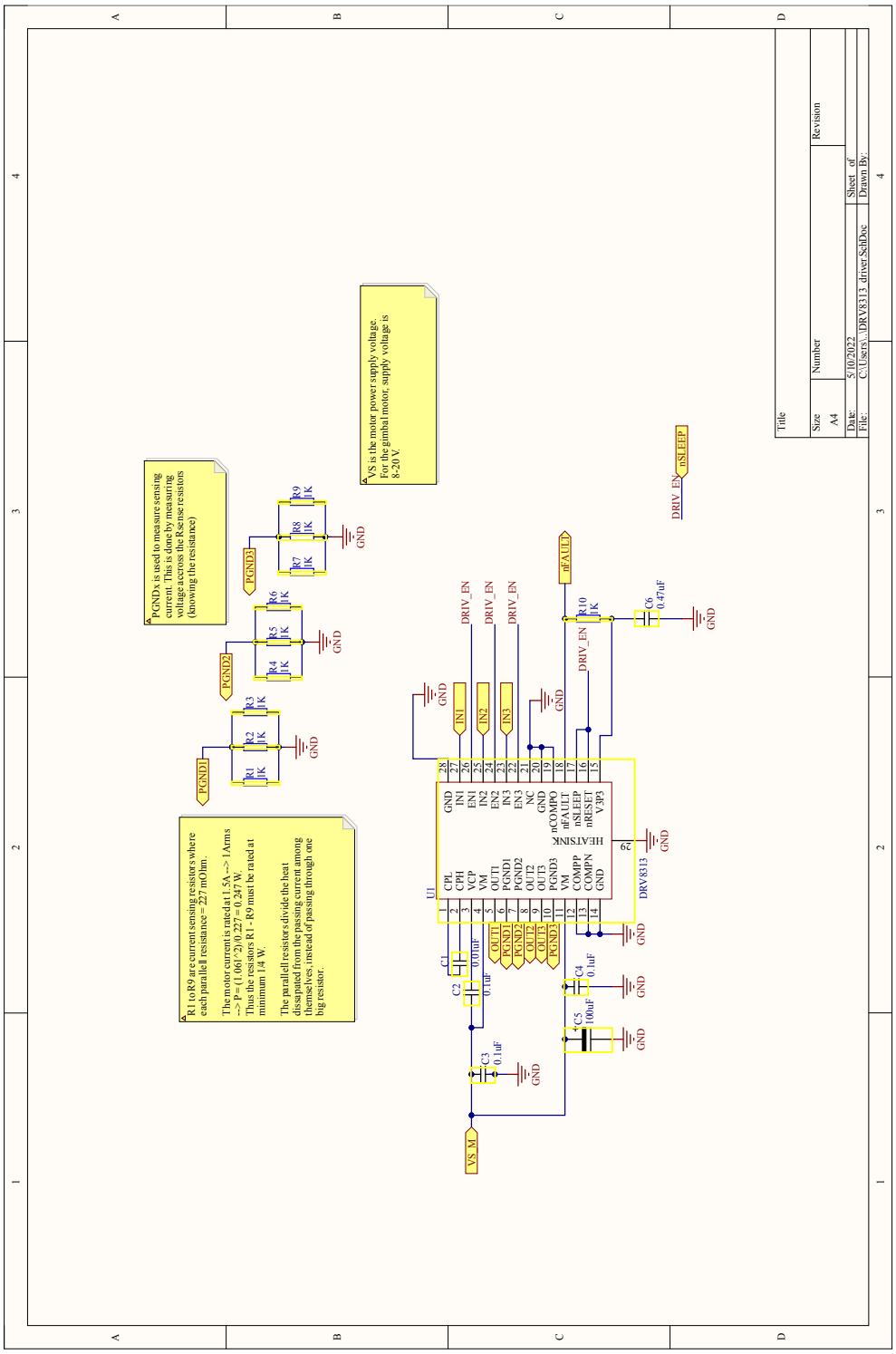

Figure A.2: Schematic of the motor driver circuit (one per BLDC motor).

The schematic below contains the current amplifier INA2181, which is used to convert the phase currents through the BLDC motors to an ADC-readable voltage.

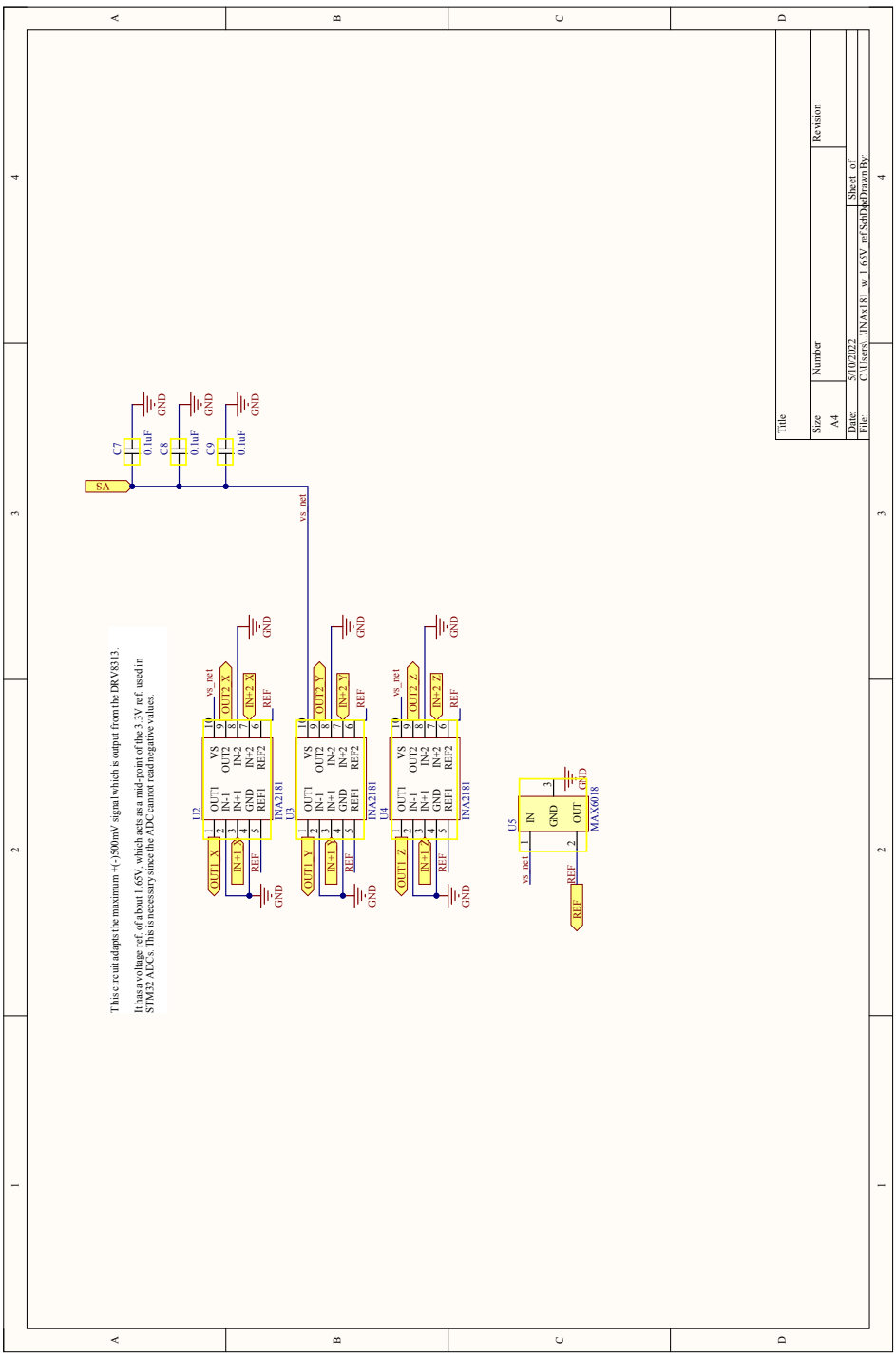

Figure A.3: Schematic of the current amplifier circuit.

The schematic below shows the pins that will be connected directly to the Nucleo board. This circuit is used to connect the microcontroller to the motor driver, current amplifier, etc.

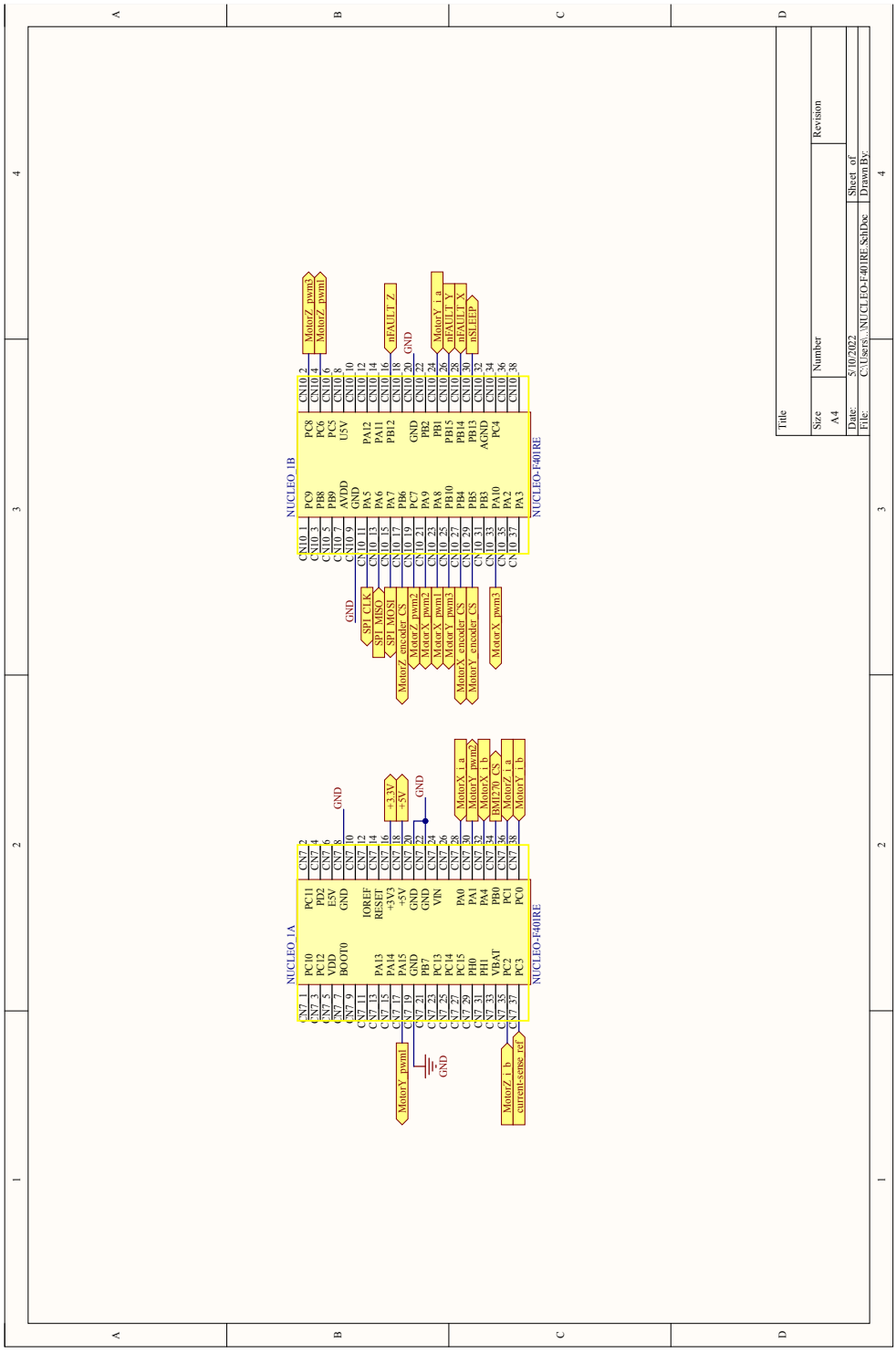

Figure A.4: Schematic of the connections between the Nucleo board and the gimbal driver shield.