CZECH TECHNICAL UNIVERSITY IN PRAGUE

FACULTY OF MECHANICAL ENGINEERING

BACHELOR'S THESIS

ROTOR BALANCING MACHINE

2024

MATOUŠ HRUBÝ

BACHELOR'S THESIS ASSIGNMENT

I. Personal and study details

Student's name: **Hrubý Matouš**   Personal ID number: **494660**

Faculty / Institute: **Faculty of Mechanical Engineering**

Department / Institute: **Department of Instrumentation and Control Engineering**

Study program: **Theoretical Fundamentals of Mechanical Engineering**

Branch of study: **No Special Fields of Study**

II. Bachelor's thesis details

Bachelor's thesis title in English:

Rotor balancing machine

Bachelor's thesis title in Czech:

Konstrukce vyvažovačky rotorů

Guidelines:

The aim is to design a small machine for rotor balancing. It should be used for balancing of rotors with permanent magnets.
Approximate dimensions of the rotor: length 150 mm, diameter 20 mm. The project includes these steps:
1) mechanical design (rotor holders, drive motor holder, belt drive, etc.). The parts can be 3D printed.
2) sensors - probably equipped with strain gauges on the rotor carriers
3) electronics - selection of suitable sensors
4) programming - e.g. with LabView or on PC + Arduino - create user interface and data acquisition software

Bibliography / sources:

[1] Schneider, H.: Rotor Balancing: Fundamentals for Systematic Processes 1st ed. 2023 Edition, Springer Vieweg; ISBN-13
: 978-3662660485

Name and workplace of bachelor's thesis supervisor:

doc. Ing. Martin Novák, Ph.D.   Division of electrotechnics FME

Name and workplace of second bachelor's thesis supervisor or consultant:

Date of bachelor's thesis assignment: **26.04.2024**   Deadline for bachelor thesis submission: **31.05.2024**

Assignment valid until: _____

|		
doc. Ing. Martin Novák, Ph.D.	prof. Ing. Tomáš Vyhlídal, Ph.D.	doc. Ing. Miroslav Španiel, CSc.
Supervisor's signature	Head of department's signature	Dean's signature

III. Assignment receipt

The student acknowledges that the bachelor's thesis is an individual work. The student must produce his thesis without the assistance of others,
with the exception of provided consultations. Within the bachelor's thesis, the author must state the names of consultants and include a list of references.

_____
Date of assignment receipt

24.4. 2024

_____
Student's signature

Bachelor Project

Czech Technical University in Prague

F2 Faculty of Mechanical Engineering
Department of Instrumentation and Control Engineering

Rotor Balancing Machine

The Development and Production of a Model Machine for Educational Purposes

Matouš Hrubý

Supervisor: doc. Ing. Martin Novák, Ph.D.
Field of study: Theoretical Fundamentals of Mechanical Engineering
May 2024

Acknowledgements

I would like to thank doc. Ing. Martin Novák, Ph.D. for his patient guidance and advice during the process of writing this thesis. I would also like to thank my family for their support during the preparation of this thesis and during my studies.

Declaration

I declare that I have prepared the submitted work independently and that I have listed all sources of information used.

Prohlašuji, že jsem tuto práci vypracoval samostatně s využitím zdrojů, které jsou všechny v práci uvedeny.

V Praze, 30.5.2024

Abstract

Vibrations induced by spinning rotors are a common problem that cause damage and improper functioning of machines. In order to decrease these vibrations, balancing machines are used to establish the unbalance of the rotors. Balancing machines are often quite elaborate and expensive; therefore, one of the aims of this thesis is to build a balancing machine on a budget using 3D printing and basic components. This machine is meant to be used as a lecture tool to demonstrate the basic use of sensors when balancing. The mechanical design of the machine corresponds to the classical concept of a balancing machine. The sensors are two accelerometers and a reflective optical sensor, they are controlled by Arduino Mega 2560 board. Sensor calibration, precise timer-based sampling, and value calculation are done by a code loaded into the microcontroller. The machine is suitable for educational purposes; the data from the sensors are without significant interference and can be seen live. For complex balancing, it has proved to be difficult to get enough precise data.

Keywords: Balancing, 3D-printing, sensors, microcontrollers

Supervisor: doc. Ing. Martin Novák, Ph.D.
Ústav přístrojové a řídící techniky,
Technická 4,
Praha 6

Abstrakt

Vibrace vyvolané otáčejícími se rotory jsou běžným problémem, který způsobuje poškození a nesprávnou funkci strojů. Ke snížení těchto vibrací se používají vyvažovací stroje, které zjišťují nevyváženost rotorů. Vyvažovací stroje jsou často poměrně složité a drahé, proto je jedním z cílů této diplomové práce sestrojit vyvažovací stroj bez vysokých nákladů - pomocí 3D tisku a základních komponent. Tento stroj má sloužit jako pomůcka při přednáškách k demonstraci základního použití senzorů při vyvažování. Strukturou odpovídá stroj klasickému konceptu vyvažovačky. Senzory jsou dva akcelerometry a reflexní optický senzor; jsou řízeny deskou Arduino Mega 2560. Kalibraci snímačů, přesné vzorkování na základě časovače a výpočet hodnot provádí kód nahraný do mikrokontroléru. Zařízení je vhodné pro výukové účely; data ze senzorů jsou bez významného rušení a lze je sledovat živě během přednášky. Pro komplexní vyvažování se ukázalo, že je obtížné získat dostatečně přesná data.

Klíčová slova: Vyvažování, 3D-tisk, senzory, mikrokontroléry

Překlad názvu: Vyvažovačka — Vývoj a výroba modelového přístroje pro edukativní účely

Contents

Figures

Chapter 1

Introduction

In mechanical engineering, vibrations are recognized as a significant problem. Although there are various types of vibrations, in this thesis, I shall be exclusively concerned with the vibrations generated by the imbalance of rotating masses. Nowadays, when the trend in machines is to decrease in size and increase in rotational speed, the need to tackle this issue is greater than ever. The problems caused by this kind of vibrations include increased cyclic stress leading to possible fatigue failure, unforeseen contact between the rotor and its housing, sealed bearing, etc. Rotor vibration may also cause other parts of the machine to vibrate, which may be more sensitive and prone to failure. This leads to a decrease in the life expectancy of parts or whole machines. [1] [2]

To be more illustrative, spinning shafts and parts are used almost everywhere in industry. From immense steam turbines in powerplants, to small impellers in a household pump, From an electric motor in a car to an electric motor in a blender. In every aforementioned application, vibration may be causing difficulties. Difficulties, which can be solved by proper balancing.

In this thesis I am focused on balancing of rotors of a smaller scale (approx. 150 mm length and 20 mm diameter). The aim of this thesis is to construct a balancing machine for small rotors and evaluate the precision of the results. Based on this, I shall conclude how well this machine could be used in a workshop or a laboratory. The machine will be used as an example of an application of sensors (accelerometers) during lectures at the Department of Instrumentation and Control Engineering. Therefore, it should be compact and portable and the measurement data should be easily displayed. The other specific thing about this machine is that it will be built with basic components and with the use of 3D printing. So, it may be reproduced

1

by anyone with a 3D printer, soldering iron, and other common tools.

In Chapter 2 of this thesis, a general balancing machine is described and several possible concepts of the balancing machine of the field are introduced. Chapter 3 provides the basic physical background for the balancing and the design used. In chapter 4, the process of designing 3D printed components is described, followed by information about the choice of sensors and necessary electronics. Chapter 4 is closely related to Chapter 5, where the machine is tested after assembly and further enhancements or component exchange is performed. Chapter 6 puts the machine to real use with calibrated sensors. And finally in Chapter 7, the main points of this thesis are reviewed and the results obtained are summarized.

Chapter 2

Balancing Machine: Brief Description and Application

In this chapter, I provide a brief overview of the general function and components of a balancing machine. Moreover, several types of balancing machines are listed. This chapter sets a direction for the later design process.

A balancing machine is used to determine the imbalance of a rotating part. It generally consists of these components: [2]

- Rotor **supports** on which lies the balanced part.
- A **bed** which holds firmly the supports and which enables to vary the space between the supports.
- An **electric drive** setting the rotating part into motion.
- **Sensors** rested in/on the rotor supports which are measuring the vibrations induced by the rotor, also an RPM sensor.
- A **computer software and hardware** which is interpreting and calculating the obtained data into an understandable form.

The usuall balancing process is following: [3] [4]

1. The supports and software are set according to the dimensions of the rotor.

2. The rotor is mounted to the balancing machine.

3. By turning on the electric drive, the rotor gets to balancing RPM. The RPM does not have to be the operational RPM of the part, since the amount of unbalance is not dependent on the RPM (during *rigid* rotor balancing). [5]

4. The data from the sensors is sent to the computer, the software determines the amount of imbalance based on the data.

5. The first weight adjustment is made according to the obtained results. The software gives the information about where and how much mass should be added/taken away.

6. Since the results are not entirely precise, in the next test the rotor might not be balanced yet. This leads to several iterations of measurement and mass change until the rotor meets the specified criteria.

2.1   Types of Balancing Machines

Here is a rough description of the types of balancing machines according to the manufacturers. [6] [7]

2.1.1   Hard-bearing and Soft-bearing Balancing Machines

This differentiation is based on the rigidity of the supports. **Soft-bearing** balancing machines are not firm. When the centrifugal and couple forces start to push in the places of contact between the rotor and the support (see chapter 3), the "bearing" part of the support starts to move accordingly to the induced vibration from the rotor. In this application, accelerometers are predominantly used as vibration sensors.

Hard-bearing balancing machines have an entirely rigid support. The forces created by the unbalanced rotor push on the supports. Ideally no movement of the support occurs, besides the needed deformation of the sensor. Often used sensors for this solution are piezo-electric force sensors. [9]
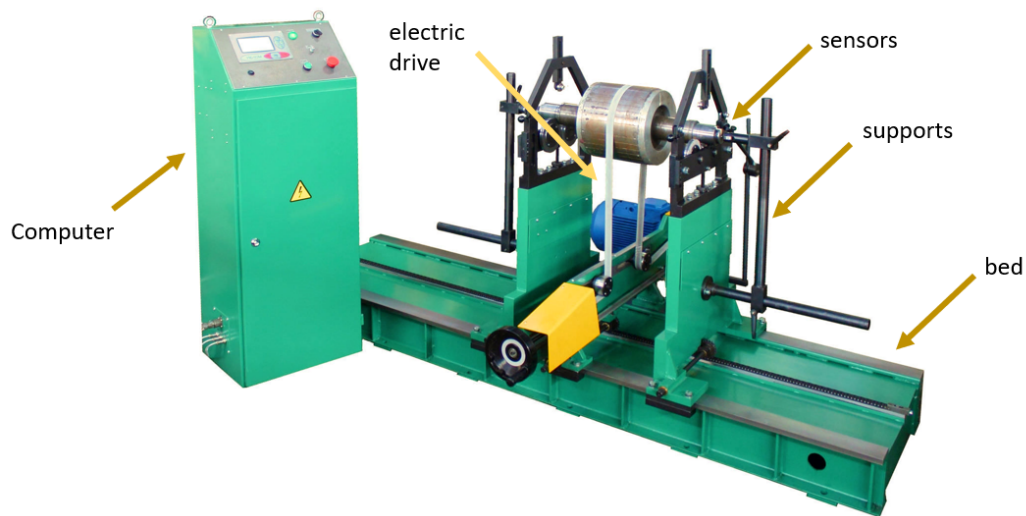
Figure 2.1: Horizontal balancing machine with belt drive, co-created [8]

2.1.2   Vertical or Horizontal

The position of the rotor in the supports can be vertical or horizontal. The vertical position is usually used for parts with a large diameter/length ratio. An example would be fans, pump impellers. The products are mounted only on the bottom side, which makes the exchange of parts much faster and the balancing process more productive. Sometimes, the balancing machines used for disc-like components are referred to as axial. The position of the rotating part is horizontal; however, the part is still mounted from one side. [7] [10]

Horizontal balancing machines have supports on both sides; further differentiation is to hard- and soft-bearing.
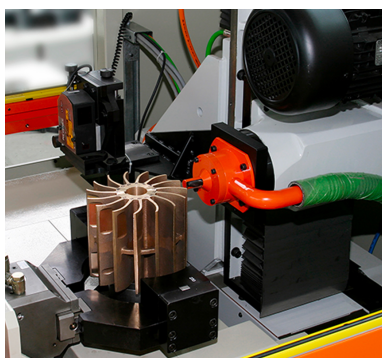
(a) : Part mounted to a vertical balancing machine [11]

(b) : Axial balancing machine [12]

Figure 2.2: Balancing machines for fans, impellers or disc like components

5

2.1.3 Belt or End Drive

In a horizontal machine, the part can be set into motion either via a belt or a universal joint coupling. A belt drive balancing machine is used for parts where there is space on the rotor for mounting the belt. Also, the belt is relatively light, so it does not influence the vibration of the part. It is used in applications where high precision is demanded and the torque needed for balancing the rotor is small to medium.

Figure 2.3: End drive balancing machine with universal shaft coupling [13]

End drive balancing machines give the torque from the electrical drive to the part through universal joint coupling (see Fig. 2.3). Unfortunately, this solution may bring impreciseness to the obtained vibration data, so it may be necessary to filter it later on in signal processing. Nonetheless, the end drive gives comparatively higher torque than the belt drive and does not need a surface on the rotating part to attach a belt. [14]

2.1.4 Special Balancing Machines and Techniques

There is a large scope of special balancing machines used precisely for the needed purpose (balancing machines for car industry, pump industry, paper industry, etc. each having their own specifications. Usually some of the aforementioned principles are incorporated into these machines with slight adjustments. Some balancing machines can balance rotors (or whole assemblies) in their own bearings or even driven by their own source of power. Also, it is possible to only take the balancing instrumentation (meaning the sensors, the software, and a computer) and make the balancing on site. [6] [7]

Chapter 3

Physical Description of Balancing

This chapter offers a background for understanding important physical phenomena when dealing with an unbalanced rotor. It touches on the topic of static, couple, and dynamic unbalance. In addition, it also covers the behavior of a pendulum and its specifics, which could influence the inner workings of the machine.

3.1 Unbalance of Rigid Rotors

The unbalance of rigid rotors can be divided into static unbalance and couple unbalance. The combination of which is called dynamic unbalance.

3.1.1 Static Unbalance

Static unbalance occurs when the center of gravity of a rotor is not lying on the axis, around which the rotor is rotating; see Fig. 3.1. That creates a centrifugal force which is linearly proportional to the mass of the rotor and the distance from the center of gravity to the axis. [1]

$$F_c = m \cdot a_c = m \cdot r \cdot \omega^2 = U \cdot \omega^2 \tag{3.1}$$
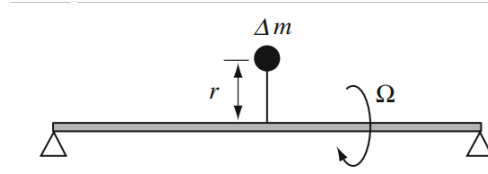
Figure 3.1: Static unbalance [1]

Where $\mathbf{F_c}$ is the centrifugal force; $\mathbf{m}$ is mass of the rotor; $\mathbf{r}$ is distance of the center of gravity from the rotation axis; ω is the angular velocity.

The quantity $U = m \cdot r$ is called unbalance. This term is widely used in the field of balancing, since it combines the radius and the mass of the object. In order to remove this unbalance U, it is necessary to remove or add weight on the same or opposite side with the exact amount of unbalance. So, to balance a rotor in that way, you can choose either mass m or radius r as a parameter and calculate the other. [15]

In theory, by placing the unbalanced rotor on perfect bearings that create no friction, we can determine the orientation of the unbalance. The gravity force would pull on the center of mass of the rotor. Then the centre of gravity would find itself directly under the rotation axis. However, we would still need to use other methods to determine the amount of unbalance. Hence, we use the rotation of the rotor to obtain that information.

If we write down the static equations for the forces and bending torque of the rotor, we can determine the axial position and magnitude of the centrifugal force F. The equations look as follows (based on fig. 3.2):

$$F = F_l + F_r \tag{3.2}$$

$$F \cdot a = F_r \cdot (a + b) \tag{3.3}$$

When balancing, the reaction forces $\mathbf{F_l}$ and $\mathbf{F_r}$ are the input based on which the centrifugal force $\mathbf{F}$ is calculated.

By modifying the equations, we obtain the following relation:
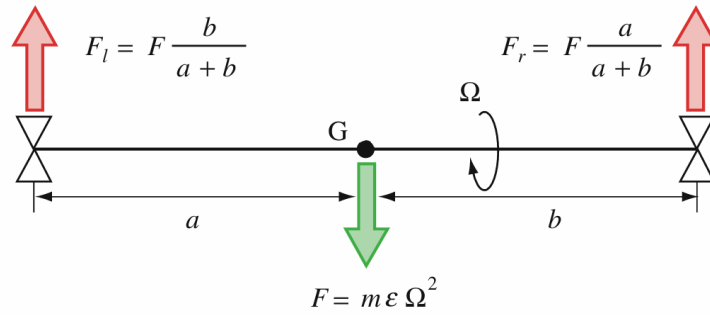
Figure 3.2: Distribution of reaction forces in static balancing [1]

$$\frac{F_l}{F_r} = \frac{b}{a} \tag{3.4}$$

That is the ratio giving us the relative axial position of the center of gravity of the object. Knowing the actual length of the rotor allows me to determine its absolute position. If the rotor is only statically unbalanced, this place determines where to remove or add material to balance it. If I remove material in any other place of the rotor, it results in an incline of the axis of rotation and thus creates a couple unbalance.

As a sensor I will be using an accelerometer. That means that I will get the acceleration of each side of the rotor instead of the force. With static unbalance, it means that knowing the mass of the rotor m, the peak horizontal acceleration (equivalent to a_c) of each side and the RPM or ω I can calculate the amount of unbalance.

3.1.2 Couple Unbalance

When a couple imbalance is present on a rotor, the center of gravity lies on the axis of rotation. However, the natural axis of rotation of the rotor is at an angle to the actual axis of rotation. This misalignment causes a spinning moment of the rotor. The situation is illustrated in Fig. 3.3. The masses in the picture are misaligning the axis of natural rotation. When the rotor spins, the two masses each create a centrifugal force - $\mathbf{F_1}$ and $\mathbf{F_2}$. Since they are apart at a distance a, together they create a moment $\mathbf{M}$ - a couple, which forces the rotor to spin. [16]

9
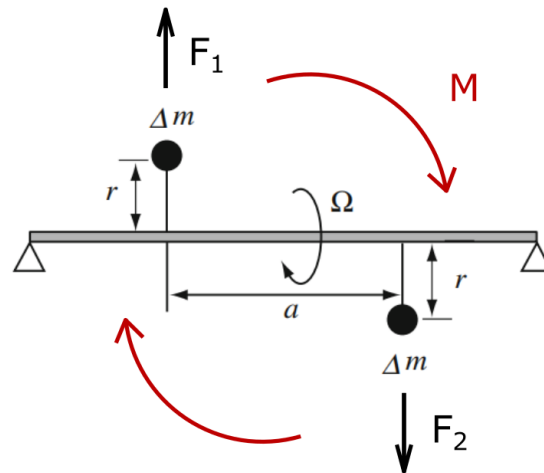
Figure 3.3: Couple unbalance, co-created [1]
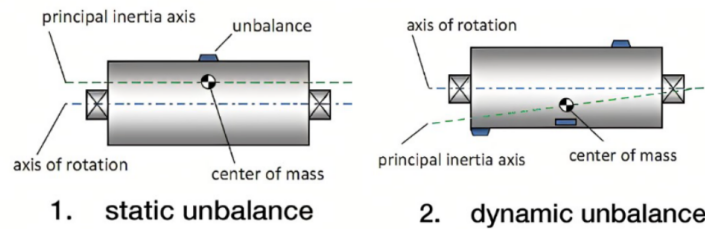
Figure 3.4: Comparison of static and dynamic unbalance [17]

■ 3.1.3 Dynamic Unbalance

The dynamic unbalance is the combination of the aforementioned types. The natural axis of rotation is: *1)* at a distance to the actual axis of rotation *2)* misaligned with the actual axis of rotation. The difference between static and dynamic unbalance is illustrated in Fig. 3.4.

■ 3.2 Basic Oscillation of a Pendulum

This section is mainly important for the design choice of my machine. I opted for a soft-bearing balancing machine, heavily inspired by [18]. This concept uses the principle of a pendulum as a "seat" for the sides of the rotors.

I chose this solution because it is relatively simple to construct. The other concept
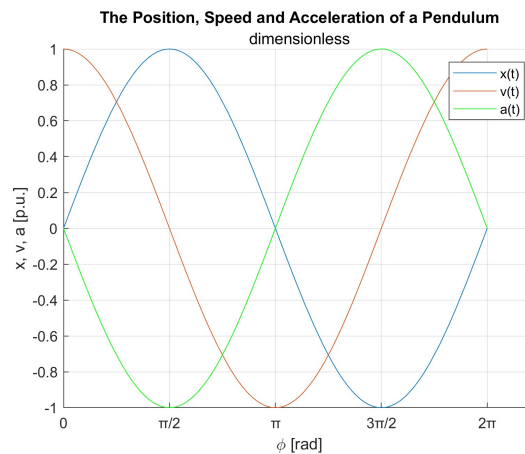
Figure 3.5: The position, velocity and acceleration of a general pendulum

on the table would be some kind of linear guide rail. There I was not sure if I could reduce friction enough. Also, I would have to deal with the effect of the springs putting the rotor end into its original position. Overall, the pendulum seemed like the best solution.

In the aforementioned source, no data processing is used. The balancing machine works solely on the principle of determining the position (angle) of the unbalance based on the amplitude of the accelerometers. The balancing process is then made by trial and error until the amplitude is reduced to sufficent level. However, in this work, I would like to get a result based on the obtained data. First of all, I need to get some physical background and estimate the behaviour of the pendulum.

The rotor will be placed on each side on the supports, which should ideally move freely without friction and only horizontally. If I am able to minimize these effects, the accelerometer will provide me with more precise data about the part. In this section I am providing basic calculation in order verify that this construction choice is correct. I am also estimating how the physical quantites, like angular speed of the rotor, length of the pendulum etc. influence the resulting data. This estimation will then influence the dimensions (predominantly length) of the pendulum used.

3.2.1 Deflection Estimation

The maximum acceleration in which the pendulum is found is in its maximum deflection (see Fig. 3.5).

The situation is illustrated in Fig. 3.6. Based on the geometry and Newton's

11

second law of motion, we can write the situation in this equation:

$$I\ddot{\phi} = -F_g \cdot l \cdot sin(\phi) + F_c \cdot l \cdot sin(\omega t) \tag{3.5}$$

Where $\mathbf{I}$ is the moment of inertia of the pendulum; ϕ is the angle of the pendulum; $\mathbf{F_g}$ is the force of gravity; $\mathbf{l}$ is the length of the pendulum; $\mathbf{F_c}$ is the centrifugal force; ω is the angular velocity; $\mathbf{t}$ is time.

By inserting into the equation for the known variables and superseding the $sin(\phi)$ by only ϕ (This operation can be made if the deflection of the pendulum does not go over 5°), we get the following equation and its modification:

$$m \cdot l^2 \cdot \ddot{\phi} = -m \cdot g \cdot l \cdot \phi + m \cdot r \cdot \omega^2 \cdot l \cdot sin(\omega t) \tag{3.6}$$

$$\ddot{\phi} + \frac{g}{l} \cdot \phi = \frac{m \cdot r \cdot \omega^2}{m \cdot l} \cdot sin(\omega t) \tag{3.7}$$

Where $\mathbf{m}$ is the mass of the pendulum; $\mathbf{g}$ is the gravitational constant; $\mathbf{r}$ is the distance of the center of gravity from the rotation axis.

In its own right, the mass that creates the centrifugal force F_c is actually only the mass of the rotor without the mass of the rest of the pendulum. However, since the pendulum is plastic PLA with two small bearings and the rotors are usually metal and quite heavy, I decided to neglect this difference in this calculation, especially when it is only a rough estimation.

Now we can obtain the mathematical relationship between all the parameters and the maximum deflection d:

$$d = \frac{r \cdot \omega^2}{l \cdot (\Omega_0^2 - \omega^2)} \tag{3.8}$$

Where $\mathbf{\Omega_0}$ is the natural frequency of the pendulum; also $\Omega_0 = \sqrt{\frac{g}{l}}$.

Based on these calculations, I estimate possible physical properties of the system with some case values. Let the length $\mathbf{l}$ of the pendulum be 40 mm and the eccentricity $\mathbf{r}$ be 1 mm. Now, a further neglection can be made, since for these values the natural
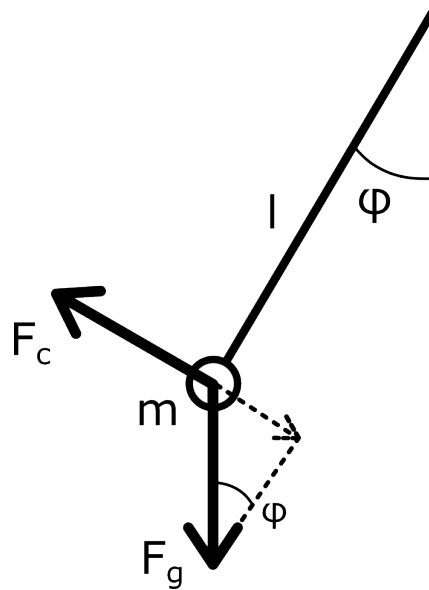
Figure 3.6: The forces effecting the pendulum

frequency $\Omega_0 \doteq 2,5 \; Hz$. Since presumably $\omega^2 >> \Omega_0^2$, we can simplify the maximum deflection equation 3.8 to this relation:

$$d = \frac{r}{l} = \frac{1}{40} = 0,025 \; rad = 1.43° \tag{3.9}$$

Based on that, we can see that 1) the maximum deflection is a function of the eccentricity and length of the pendulum; 2) the deflection based on the estimated values is much lower than 5°. From my point of view, this result looks quite promising. Then I have some physical basis to show that this solution is a viable choice. Moreover, I have a rough estimate of how long the pendulum should be.

3.2.2 Negative Influence of Gravity

When the pendulum is deflected, the vector of gravitational acceleration is divided into its tangential and normal components. The sizes of these components depend on the angle ϕ. From the tangential acceleration we can take its horizontal component a_{hg}. For an illustration, see Fig. 3.7. In the equation, the horizontal acceleration is:

$$a_{hg} = g \cdot sin(\phi) \cdot cos(\phi) \tag{3.10}$$

13
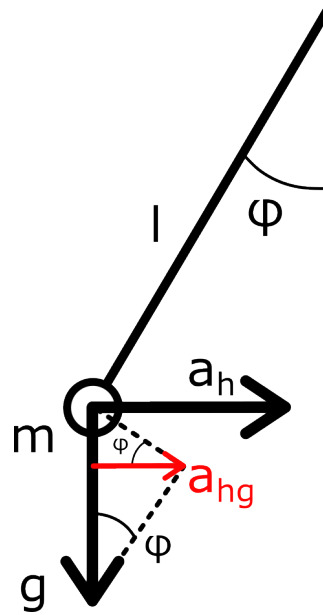
Figure 3.7: The circular and gravitational horizontal acceleration

Where **g** *is the gravitational constant,* ϕ *is the deflection angle of the pendulum.*

This "by-product" horizontal acceleration can influence the results. However, based on Fig. 3.8, when ϕ is low, it can be negligible. It depends on the size of the obtained acceleration a_h. If the ratio a_h/a_{hg} is very large, a_{hg} can be neglected. Another point is that the balancing is done in iteration. So, even if the first measurement is imprecise, in the other iterations the deflection is gradually decreasing, and also the influence of a_{hg}.

The magnitude of acceleration a_{hg} based on deflection can be seen in Fig. 3.8. Taking into account that it depends only on the deflection and nothing else, the ratio a_h/a_{hg} can be increased either by lengthening the pendulum (thereby lowering the maximum deflection according to equation 3.9) or by increasing a_h, which depends on the RPM of the rotor.
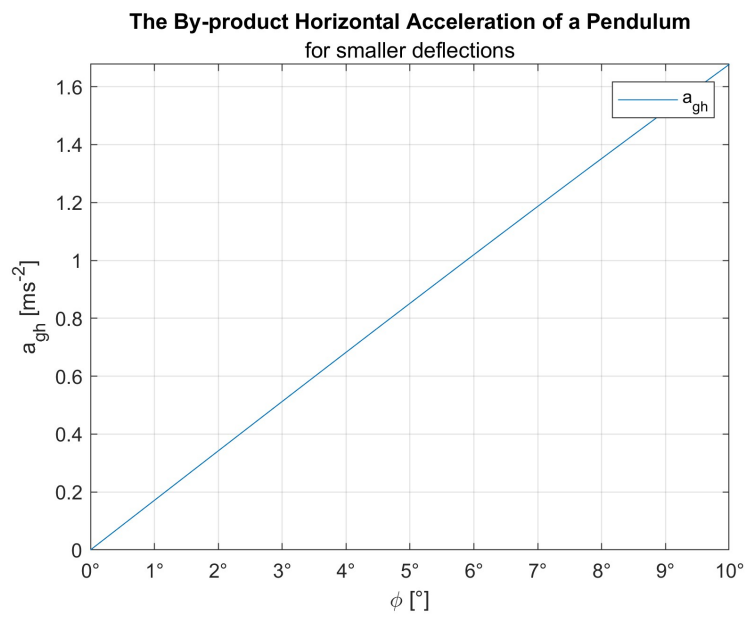
Figure 3.8: Acceleration a_{hg} versus deflection ϕ

Chapter 4

Building Process

In this chapter, I describe the machine building process while also listing all the components and materials. I am putting a heavy emphasis on the main sensor - accelerometer, and I also mentioned some alternatives to it, since the machine will be used as an example of application of accelerometers and other sensors during lectures of the Department of Instrumentation and Control Engineering.

During the construction of this machine, my main source and inspiration was a very nice instructable [18]. Some of the basic principles influenced me to a great extent, e.g. the pendulum, the Z-axis compensation, and so on. However, in many ways and details, I tried a different approach. All the 3D printed parts are modelled by me, I incorporated the stock parts available to me from the laboratory and tried to make few more practical adjustments.

Based on the information provided in Chapter 2, I should specify which concept of balancing machine I use for this project. I build a **horizontal soft-bearing balancing machine**. The balancing machine is composed of many parts; however, in general there are 5 main components, as specified in Chapter 2. Those are the supports, the bed, the computer hardware and software, the sensors, and the electric drive. It is illustrated in Fig. 2.1 and can be compared to the balancing machine in Fig. 4.3. In this chapter, the development of these components is described (however, they may not be clearly named as in the aforementioned example).

4.1   3D Printing

One of the aims of this work is to build a prototype that could be built in the best-case scenario by almost anyone with access to basic components and tools. That is why I am using 3D printing. It gives the opportunity to design and make necessary parts for not too large of a cost. Then I can test those components and, if necessary, make changes to the CAD model and print them out again. So, every time I find a fault in my design, I can start again the next day or sooner.

The material that I am using for 3D printing is **PLA** - polyactic acid. It is the most common material for 3D printing, it has a low price and sufficient mechanical properties for my project. There are also other advantages, like that it is made from renewable sources and it is compostable. Different uses for PLA are also food packaging, medical implants, etc.[19]
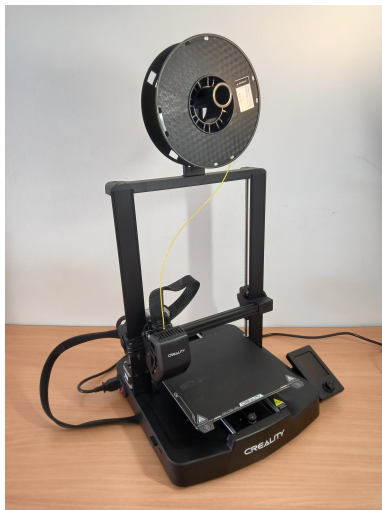
Figure 4.1: Creality Ender-3 V3 SE

Being a thermoplast, it melts beyond a certain temperature and can be shaped into various forms. This is used when extruding filament from a 3D printer head, injection molding, thermoforming, etc. Unlike other cathegories of widely used plastics - thermosetting plastics - it can be melted and shaped repeatedly without significant changes in the material, which makes it recyclable. It does have a few disadvantages due to its nature. For example, it might become soft when the temperature is substantially increased (glass transition temperature is 60 - 65 °C). But that should not affect my project in any way.[19]

For the purposes of this thesis, I have bought my own 3D printer. The value of 3D printers is nowadays quite considerate, according to my view as a student. However, mainly having a 3D printer available at all times makes prototyping process faster and easier. My Creality **Ender-3 V3 SE** (Fig. 4.1) is then the backbone of my project.

Regarding 3D printing, it has a lot of specifics for the CAD designed parts as well as for the setup of the 3D printer. I learned on the go, and sometimes I was more successful, sometimes less successful.

Here is a few of 3D printing hot takes that I learned:

1. When making a fit, you have to accommodate for the width of the filament - holes should be designed slightly larger to fit the desired diameter.

2. The printing area must be thoroughly cleaned to ensure that the first layer adheres to the surface. Compensating this with Z-axis offset is possible, but be careful not to overdo it. You will damage the print surface.

3. The printer is quite loud. Trying to fall asleep in the same room while waiting for an 8 hour print is a bad idea.

4. It is recommendable to print the outer layers at a lower speed to obtain a smoother surface.

5. Prusaslicer [20] is more user-friendly than Cura [21]. However, Cura has more pre-installed settings of 3D printers, including mine (as of 3/25/2024). The printer parameters in both programs can also be set manually.

This is my 3D printing designing process:

1. First the part is designed in a CAD software. I used Autodesk Inventor [22].

2. Then the Inventor file .ipt is exported as .stl file. Stl file is a file that represents the surface of the modeled object. It breaks it into simple geometrical shapes, most often triangles. In this step, it is important to set a high resolution, or everything round might end up like polygons. [23]

3. The part is then uploaded to a slicer. That is a special program for setting up the 3D printer and the 3D printing process. The position and orientation of the printed object is determined. All the necessary characteristics are determined for the internal structure, the layer height, the printing speed, the printing nozzle temperature, the print bed temperature, and scores of other characteristics to get the best print.

4. The slicer, in my case Ultimaker Cura [21], generates a g-code and uploads it to an SD card. G-code is a special machine code which contains all the information necessary for the printer setup and extrusion head and printboard movements.

5. Then I clean the printboard with isopropyl alcohol and stick in the removeable SD card. The printing takes place. Usually, it begins with a calibration.

6. After the printing is completed, I carefully remove the part(s) from the printboard and remove all printing supports, when printing overhanged or bridge-like structures. Then I make sure that every important dimension has the correct
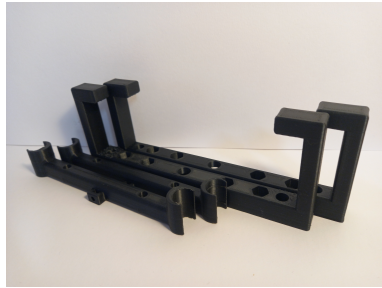
(a) : Some of the unusable printouts   **(b) :** Design fault which led to a rupture
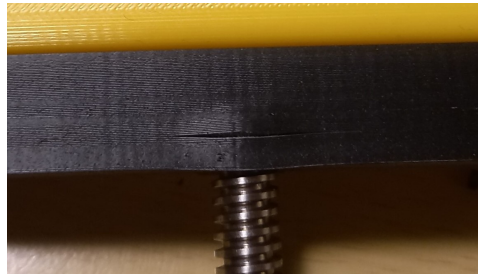
Figure 4.2: Errors in the creating process

size and other necessary properties of the part I need. If I identify a problem, I change the original CAD file to eradicate it. Usually I try to solve the problem without repeating printing (drilling a bigger hole, using sandpaper, adding glue, etc.). If it cannot be solved, I have to print a new part → back to step 2.

In Fig. 4.2 you can see examples of bad prints. Up until this point I did not manage to create a disastrous example of a failed print. So, the following examples look normal to the naked eye. However, they always had a design flaw which made them useless for further use. PLA is recyclable, but ways of recycling PLA at home are unfortunately still very expensive.

◼ 4.2 Parts

In this section, I will describe the designed 3D printed parts in the order they were created and the components with which they are connected. I started my design of the parts from the supports in contact with the rotor to the frame holding everything in place.

In order to understand better the description and workings of the machine, see Fig. 4.3, where the whole machine is in the state as of 4/14/2024, with a **coordinate system**.

Since sometimes I have to update some of the components while working on this thesis, at the end of each part's description, I will add a paragraph including a brief summary of the adjustment I made.
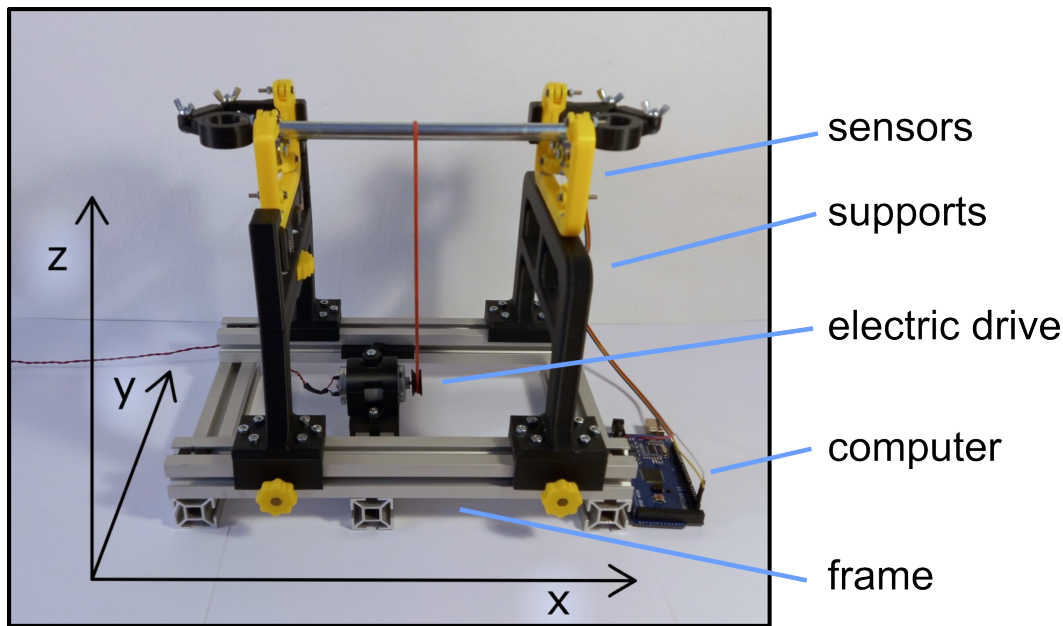
Figure 4.3: Rotor balancing machine to date 4/14/2024, with coordinate system

4.2.1 Pendulum

As already mentioned in section 3.2, the main part of the machine is the pendulum supporting the balanced rotor. The ideal support would let the end of the rotor move completely freely in the horizontal direction. And in the vertical direction it would completely constrain the rotor. This is simulated by the pendulum, which has a low deflection/pendulum length ratio. So, the vertical movement (z-axis) of the rotor should be negligible. The horizontal movement (y-axis) then depends on the properties of the rotating rotor. Unfortunately, here also apply quite a few of passive resistances.

The main one is friction in the joints of the pendulum. This is avoided by placing small radial bearings in the joints, two in each. After connecting two parts together, it is necessary to prevent any other contact between the two connecting parts of the pendulum. For that, I used a combination of spacer rings, which I printed out from PLA. PLA is by no means suitable for this job, unfortunately I had no other material at hand. Most importantly, it works.

The two bearings on the pendulum panel provide good conditions for spinning of the rotor without too much resistance or unexpected behaviour. On the outside of the pendulum panel is mounted the MPU-6050 accelerometer. You can see the pendulum with the accelerometer in Fig. 4.4.
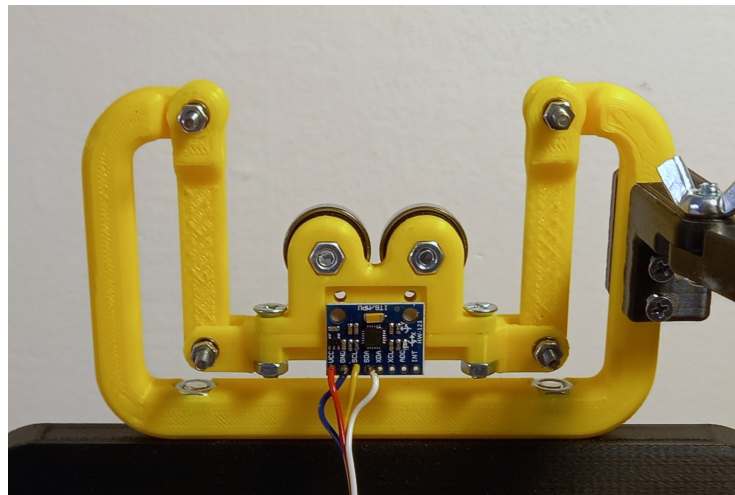
Figure 4.4: The pendulum with accelerometer, as of 4/15/2024

V 2.0: I had to remove the bearings as a point of contact, because they were creating noise in the signal. After going through several variants, the best solution was a piece of teflon as point of contact. See more in Section 5.

4.2.2 Supports

The supports overall have two main functions. Firstly, they provide enough space between the frame and the pendulum for rotors with larger diameters (e.g. fans or other disclike objects). Secondly, they enable the compensation in x-, y- and z-axis in order to adjust the machine for the correct size of the rotor.

The y-axis compensation is the simplest. It consists only of two slots in the simple support for the mounting screws (Fig. 4.5b). This enables the whole pendulum to move to the sides when loosening and tightening the screws holding it in place. It is only there to compensate for the imprecision of the 3D prints and the overall assembly. It is to be set once during assembly of the balancing machine. The aim of this is for the pendulums to be directly opposite each other, so that the rotor does not sit oblique.

The x-axis compensation is for adjusting the machine for rotors of different lengths. The whole supports are mounted on a 3030 profile with a T-slot. They can be moved along the profile and then locked in position by tightening the yellow wheel from the side of the profile.

The z-axis compensation is for lowering one pendulum in case of a rotor with
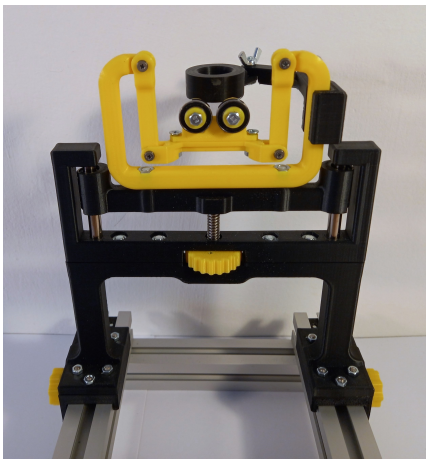
(a) : Support with the compensation in the z-axis
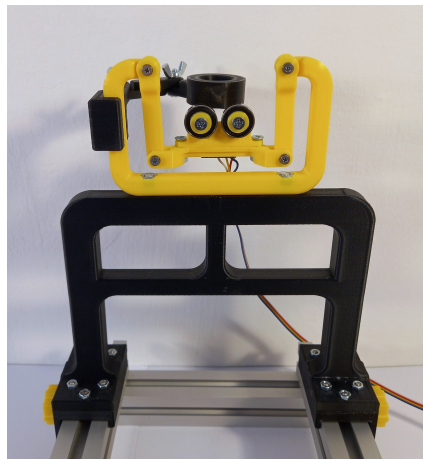
(b) : Support with the compensation in the y-axis

Figure 4.5: Supports

different sizes of diameters on each end. Without compensation, there would naturally be an incline of the rotor, which could influence the measurements and the stability of the rotor. You can see it in Fig. 4.5a. It consists of two 8mm rods with linear bearings, which act as linear rails, and an 8mm trapezoidal rod. The lead screw is nested in the moving part of the support, where the pendulum is fixed. Then the lead screw goes through the yellow adjusting wheel, which has a 8mm trapezoidal nut in it. By turning the adjusting wheel, the pendulum is elevated or lowered.

After the assembly of the machine, I realized that I made the bottom of the supports unnecessarily wide. I wanted the connection of the supports to the frame to be quite durable, so that the machine does not desintagrate so easily with further use. Based on that, I added 6 M4 bolts, but 4 would be enough. In that way I could shrink the minimum distance between the pendulum bearing by cca 18 mm, which would enable the machine to balance smaller electric engine motors, similar to the one I am using for the powering of my machine.

■ 4.2.3 Frame

The frame is assembled from 4 3030 profiles with T-slots. The profiles are quite cheap in comparison to other methods for attaching the supports. With the practical T-slots, they enable me to easily slide in any necessary part of the machine, like the supports, the block with the engine, box for arduino or power source. Moreover, there were enough leftovers of the profile available in the lab of my Faculty Department.
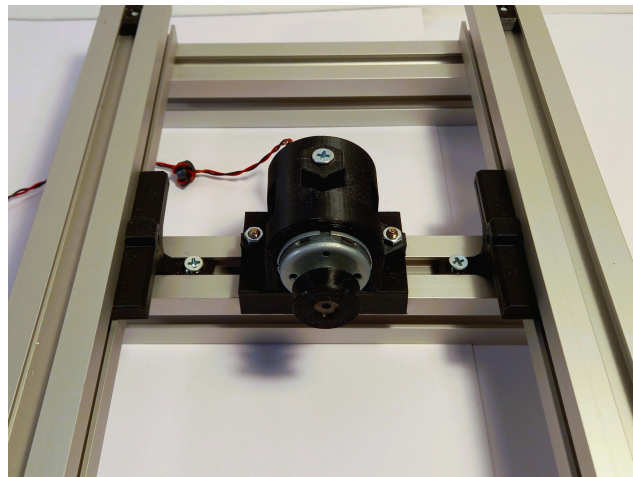
Figure 4.6: DC motor positioned in the frame, with V1 3D-printed pulley

The profiles have a drilled hole on each end, and then they are connected via a 4mm bolt and a nut nested in a special inset. This kind of assembly is not rock solid, but should be rigid enough. On the plus side, it is easy to take apart, which is handy when making a prototype. You can see the frame in Fig. 4.3.

◼ 4.2.4 DC motor

The DC motor was obtained from the supervisor of my thesis. According to its datasheet, it operates at 12-42 V and the current at maximum efficiency/stall should be 0.38/1.85 A. The speed at maximum efficiency can achieve up to 5380 rotations/min. The relatively high speed of the motor may become a difficulty in the future. According to first tests, the sensors are not able to catch up with the turning speed of the rotor. Part of the solution may be to feed lower voltage to the motor (currently 19 V), make a smaller pulley on the rotor, or to get a sensor with higher sampling rate. I might even add a circuit to drive the speed of the motor manually. [24]

The motor is placed below the pendulums at the level of the frame. It is attached to a piece of 3030 profile, which is connected to the rest of the frame via special slide-ins. That enables the motor to be moved along the x-axis making the machine all the more adjustable for different kinds of rotors (see Fig. 4.6).

The motor has its own pulley, which is suitable for a timing belt. However, these small machines are usually fitted with a rubber band to transfer the power from the motor. As I see it, there are a few advantages: 1) the rubber band has a enough friction, so it does not slide on the surface of the rotor and 2) it stretches easily,

so I don't have to make some other tensioning rigs for the belt. 3) Its stretching properties help to overcome the fast acceleration of the DC motor. That is why I also chose the rubber band. For it, I designed and 3D printed a few prototypes of pulleys, which I then embedded on the old pulley. By adjusting the size of the pulley, I can change the drive transmission ratio and speed up the rotor. Currently, the power transmission is causing me difficulties because the rubber is pushing the testing rotor away to the side along the x-axis.

V 2.0: Fitting the PLA pulley on the old pulley was not a good idea, because it had a tendency to misalign and then add to the vibrations of the machine. Since the old pulley was tightly fitted to the shaft of the DC motor, I had a difficulty to remove it without damaging the motor (that is why I did not remove it in the beginning). However, after removing it and fitting the printed pulley directly on the axis, the connection was much sturdier.

V 3.0: I changed the attachment of the motor to the frame. It is now much more firm and also suitable for an exchange of the motor if necessary, since it can now accomodate motors of various sizes.

4.2.5 Axial arms

The axial arms prevent the rotor from moving along the x-axis when balancing. However, since the contact of the rotor with the axial arms creates frictional resistance, it should be avoided as much as possible. There should be at most a gentle push against the arm.

The axial arms are attached to the stationary yellow frame of the pendulum and consist of 3 moveable parts. The arms are adjustable; they are locked into position by tightening two wingnuts in each joint of the arm. The point of contact is, as of now, a PLA ring. Few other possibilities with lower resistance, such as teflon or a simple radial bearing are considered. You can see the axial holder in Fig. 4.7.

V 2.0: I exchanged the last segment with a segment having an ordinary skateboard 8mm bearing at the end. During the rotation of the rotor the friction seem to be smaller, the part is now much more durable. The PLA version was showing strong signs of wear on its surface after short use. This added a strong induced (fake) vibration to the rotor, when it made contact with the axial arm holder.

V 3.0: Eventually I ended up not using the holders. For the basic purposes of the machine, I fixed any movement along the x-axis by slightly misaligning
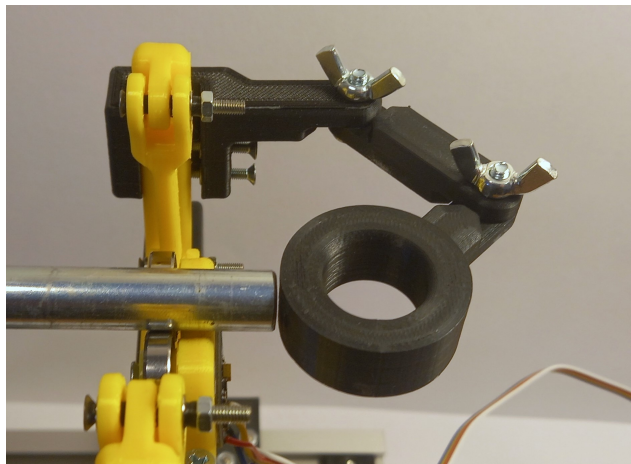
Figure 4.7: Axial holder V1

z-compensation on one of the supports.

■ 4.2.6 General Information about Construction

The frame is approximately 300mm long and 200mm wide. The total width of the machine with the yellow tightening wheels is 240 mm. The total height of the machine is 285 mm high. The maximum inner distance between the pendulum bearings is 260 mm. The minimum inner distance between the pendulum bearings is 58 mm.

The machine uses predominantly M4 nuts and bolts, with a cross- or hex head. The only exceptions are M3 screws in the pendulum joints, determined from the inner size of the joint bearing. And the other exception is M6 socket bolts, which had to be tightened in less available places, therefore, tightening with hex key was the wiser choice. **For the assembly** and disassembly of the machine, **you need:** A crosshead screwdriver of "average to smaller" size, a 5mm hex key, and a 7mm wrench.

■ 4.3 Sensors

Sensors are one of the crucial parts of this machine. For this machine, only two kinds of sensors are used, a vibration sensor to measure the size of the unbalance and

an RPM sensor, which, together with easier RPM measurements, gives the angular position of the unbalance.

■ 4.3.1 Vibration sensors

There are various vibration sensors using a wide range of principles; here, I provide a description of several of the most common ones, including the one used for the balancing machine.

■ Accelerometers

Accelerometers are usually used for soft-bearing types of balancing machines [9]. There are numerous types of accelerometer that use various physical principles to measure acceleration.

Accelerometers are sometimes referred to as inertial linear sensors. The name is derived from the fact that the main part of these sensors is an object with large enough inertia of motion. This inertia creates a force when the measured object accelerates. This force is then measured, and given that the mass of the object is known, according to Newton's second law, it is possible to determine the acceleration (eq. 4.1). [25] [26]

$$a = \frac{F}{m} \tag{4.1}$$

$$F = k \cdot x \tag{4.2}$$

The important parts of most types of accelerometers are springs and displacement sensors (Fig. 4.9). Spring can be understood as any object connecting the proof mass to the non-deforming part of the sensor. The inertia force F then deforms the spring according to eq. 4.2. k is the stiffness of the spring. If we measure the displacement x, combined with the eq. 4.1 we get the desired acceleration. [25]

Since a spring is used, it is necessary to prevent unwanted oscillations or slow response of the spring. Therefore, the system should have correct damping (ideally
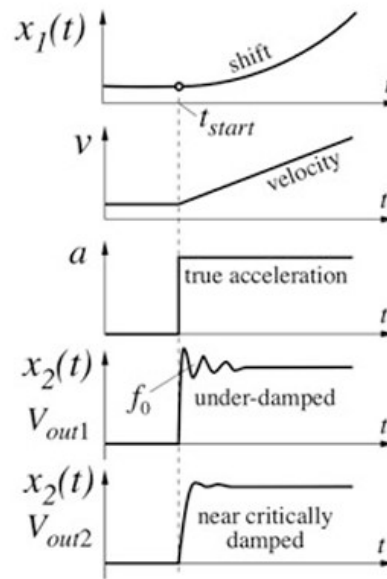
Figure 4.8: Damping of an accelerometer [25]

critical damping, Fig. 4.8). The accelerometer should not measure vibrations close to its natural frequency to prevent resonance and its effects on sensor precision. [25]

■ Capacitative Accelerometer

In this sensor, the changes in capacity between conductors are measured to obtain acceleration. Again, a proof mass is suspended on spring supports. Usually, three conductive layers are used to measure the change in capacitance. One is on the suspended mass, the other ones are on either side of the support. When the measured object accelerates, the mass moves either right or left, changing the capacity between the mass and both the left-side and right-side conductive layers. The difference in resulting capacitance between the two capacitors is through a capacitance-to-voltage circuit converted to changes in voltage, which enables it to be evaluated. Theoretically, only one capacitor (mass and one conductive layer) can be used, but when two opposing capacitors are used, by subtracting their capacities, interferences and drifts can be compensated. [25] [26]

Capacitative accelerometers have many different designs. Usually, there are more couples of capacitors in the MEMS (Micro Electrical-Mechanical Structure). In this work, the MPU-6050 capacitative accelerometer will be used. Its picture and working principle can be seen in Fig. 4.10. There is a number of sensing fingers which change capacitance between each other with the movement of the proof mass along the measured axis. [27]
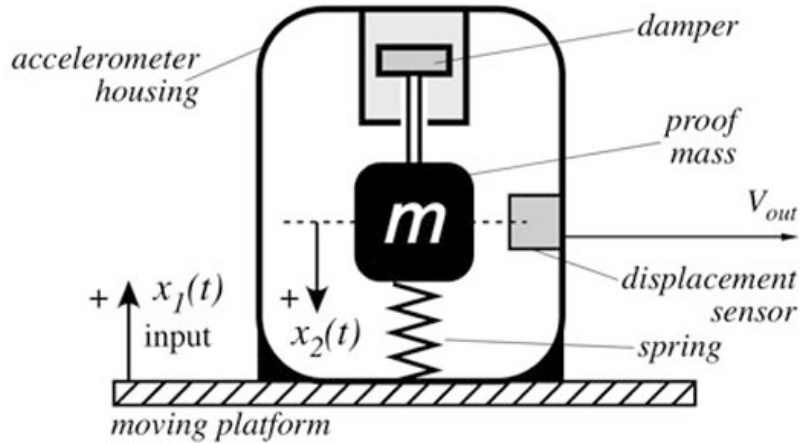
Figure 4.9: Basic parts of accelerometer [25]

The MPU-6050 has 3 accelerometers and 3 gyroscopes. Since I will be measuring vibration in one axis, only one of the accelerometers and none of the gyros will be used. The reason why this accelerometer was chosen is simple. It is cheap and available from the faculty laboratory. More importantly, it is precise and easily incorporatable to the balancing machine. It has a selectable measurement range between $\pm 2g$, $\pm 4g$, $\pm 8g$ and $\pm 16g$. From analog to digital, it is converted through 16-bit AD converter giving it 65536 steps to cover the mentioned ranges, greater than enough precision for the scope of this work. [28]

The structure of the sensor and the detail of the y-axis accelerometer are shown in Fig. 4.11.

In a different architecture, the mass is suspended not only on the springs but also on its sensing fingers. The mass moves up and down, making its sensing fingers bend. This creates a smaller overlap area between the stationary and bending fingers (Fig. 4.12). [25]

Below is a list of alternative types of accelerometers, which could theoretically be used to measure vibrations [25]. I have not considered them for practical use, solely because they were not available and could be an unnecessary expense for a project of this scale.

1. Piezoresistive accelerometer - strain gauges are used to measure the strain in the mass supporting springs. The inertia force deforms the springs, which

(a) : Actual accelerometer used in the balancing machine

(b) : Principle of the sensor [27]
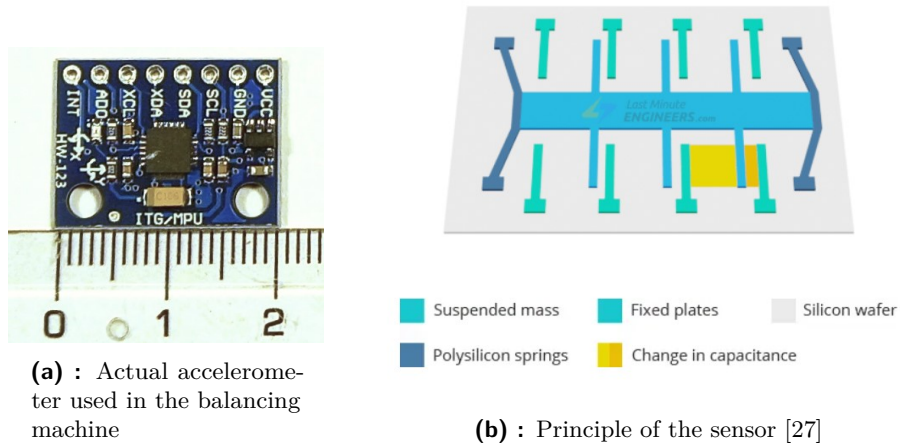
Figure 4.10: Accelerometer MPU-6050 and its working principle
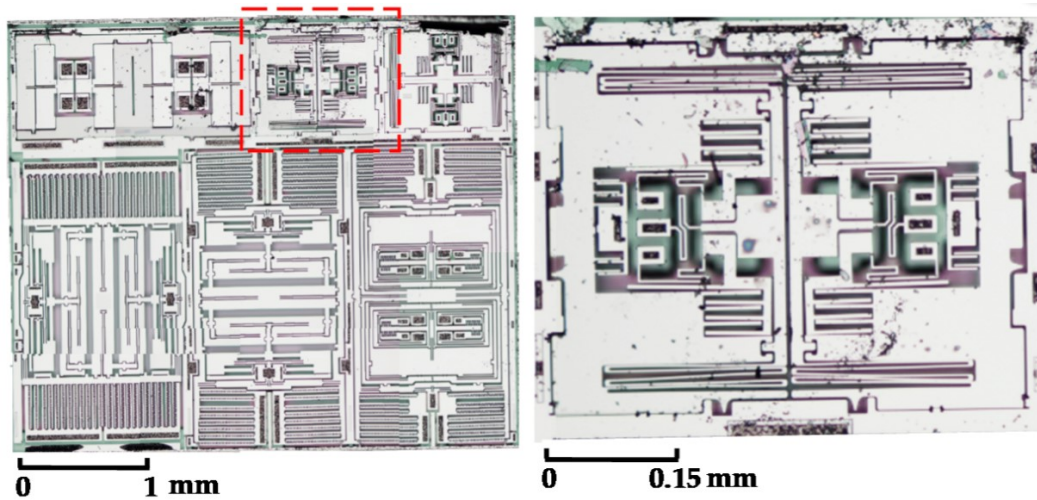
Figure 4.11: MEMS structure of the MPU-6050 [29]

changes the resistance of the strain gauges mounted in/on them. This change of resistance changes the voltage output of the sensor, which can be then recalculated to acceleration.

2. Piezoeletric accelerometer - uses the piezoelectric effect. The spring is in the form of a piezoelectric crystal, which when pressed creates positive and negative charges on the opposing sides, depending on the amount of pressure. This charge difference is then transferred to voltage through the charge-to-voltage converter (Fig. 4.13).

3. Heated plate accelerometer - similar principle to the capacitative accelerometer. Plate heated via a heating resistor is supported on springs. Above and under the heated plate are two heat sinks separated by a thermal conductive gas. The temperature of these two heat sinks is measured and based on that the displacement of the heated plate. And with the displacement we can measure

(a) : Photo with description of MEMS accelerometer

(b) : Detail of the sensing fingers
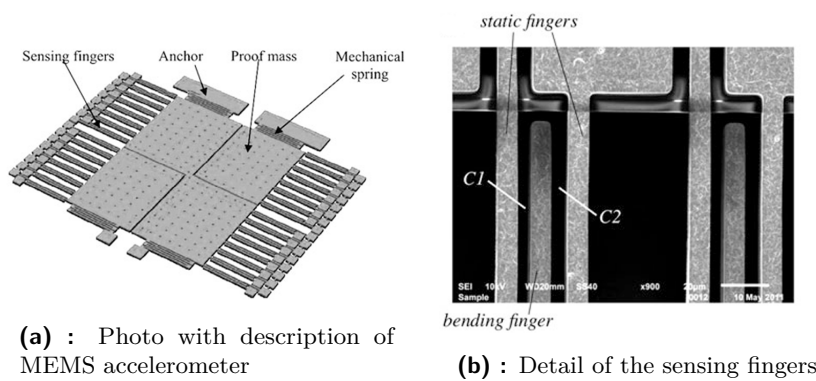
Figure 4.12: Example of accelerometer with overlapping sensing fingers [25]
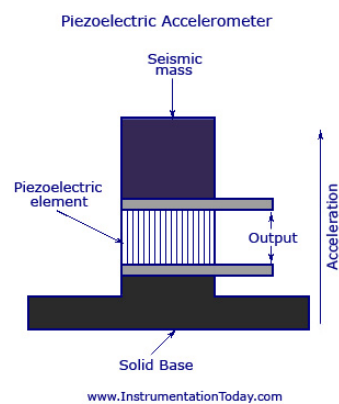
Figure 4.13: Principal of piezoelectric accelerometer [30]

the acceleration. This accelerometer might seem a little bizzare, but actually thanks to its different physical principle it is less prone to be influenced by an electro-magnetic and electro-static noise and ambient temperature.

■ **Force sensors**

The force and acceleration sensors are very similar. Since in accelerometers first a force has to be measured, and then with eq. 4.1 recalculated to acceleration, in a way, the force sensors are one step simpler. Generally, force sensors include a spring-like element and a means to measure its displacement. The simplest possible solution is a regular spring with a displacement sensor measuring its compression, an example is shown in Fig. 4.14 where for the displacement sensor a Linear Variable Differential Transformer is used.

According to a manufacturer [9] a piezoelectric force sensor is used in a hard-
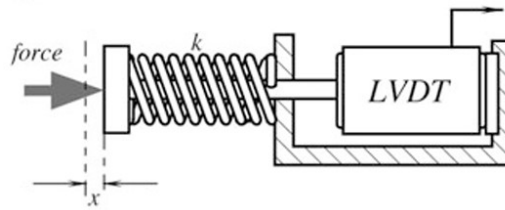
Figure 4.14: Basic force sensor [25]

bearing balancing machine. So, for the alternative design of my machine, it could be considered. This sensor uses the property of quartz crystals that under stress generate charges (as described earlier in this section). In this case, the crystal is excited into resonation by an external circuit. By applying a force on the crystal, its stiffness increases, which changes the frequency of the resonation. The applied force is determined on the basis of the frequency change. The main difference from Fig. 4.13 is the lack of the excitation signal in the picture. Also, the crystal used is cut and oriented in such an angle as to be sensitive on external load as much as possible. [25]

The most common type of force sensor is a strain gauge (Fig. 4.15). It usually consists of a thin wire on a backing glued to the measured object in the direction we want to measure the force. Since the measured object undergoes deformation when a force is applied, the wire changes its resistance due to a piezoresistive effect. Typical materials are constantan, platinum and its alloys, silicon, etc. This sensor is strongly influenced by temperature, so it is necessary to compensate for it to achieve reliable results.
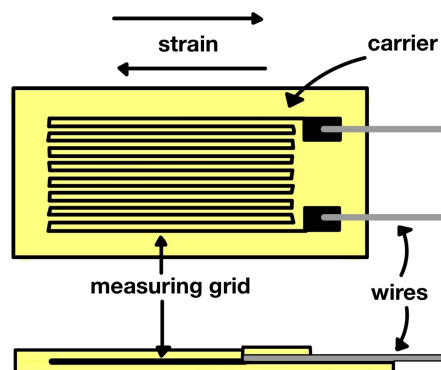
Figure 4.15: Strain gauge [31]
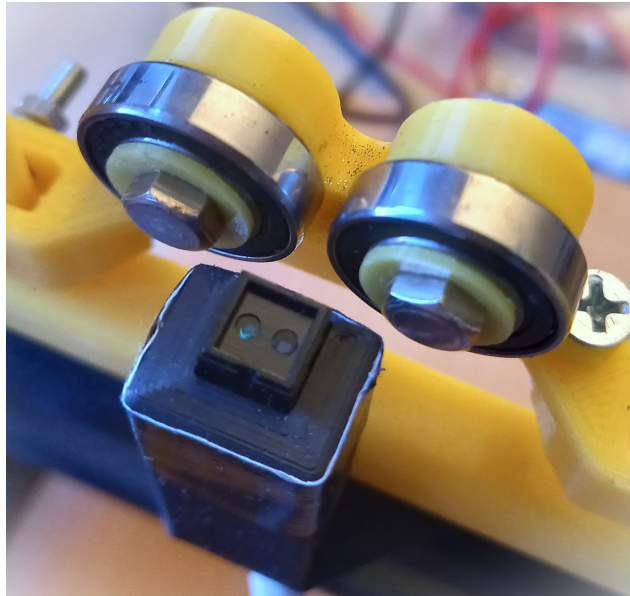
Figure 4.16: CNY70 attached to the balancing machine
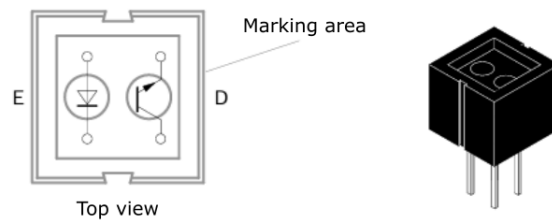
Figure 4.17: CNY70 schematic [32]

4.3.2 RPM Sensor

For the RPM sensor I will be using the CNY70 reflective optical sensor with transistor output. It consists of an infrared diode, which emits light at 950 nm, and a phototransistor. When there is an object near enough to reflect the LED light on the phototransistor, the resistance of the semiconductor junction decreases, and the voltage drop on the transistor is smaller. Therefore, a higher voltage can be measured on the output of the phototransistor. You can see the schematic of the CNY70 in Fig. 4.17. [32]

When connecting the CNY70 to 5V power source, there has to be a resistor in order to drop the voltage on the LED diode (approx. 200 Ohms). There also has to be a pull-down resistor on the output of the fototransistor, because otherwise there is an undefined state if the output is connected on an analog input of the microcontroller. I inspired myself by [33], you can see the circuit (without connection
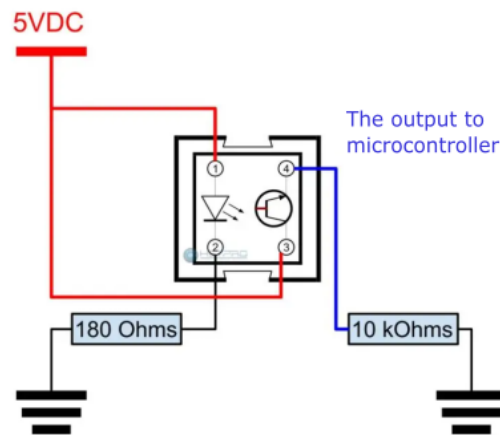
Figure 4.18: The circuit wiring of CNY70 [33]

to the arduino board) in Fig. 4.18.

4.4 Electronics

4.4.1 Arduino Mega 2560

The backbone hardware of my project is the Arduino Mega 2560 board. It is one of the most popular microcontrollers, and it is more friendly to beginners thanks to the official Arduino IDE. And, as always, it was available to me in the faculty laboratory. Similar choice for running this project would be the Arduino Uno; however, the Mega has a better microcontroller (ATmega 2560) than Uno has (ATmega 328P). The main difference is the size of the SRAM which is by Mega 8 kB and by Uno 2 kB, making the Mega available to run more complicated programs with more variables. Also, the Mega's EEPROM is much larger, 256 kB compared to 32kB, so I can upload longer code. One of the other main features of Arduino Mega is that it has many more pins, which allows it to be suitable for more complicated robotics projects. However, for this project, I do not need that many inputs and outputs, the main advantage for me is the better computing power. You can see the Arduino Mega 2560 on fig. 4.19. [34] [35]

4.4.2 Power Supply

For powering the balancing machine, I am using a 12V/2A power supply. It should be enough for the current drawn from the DC motor, which is at stall (maximum value) around 1.85 A and the Arduino, which should not take more than 100 mA even with the sensors. If it is still not sufficient and there is not enough power for the Arduino, so it suffers from brown outs, I have a 19V/3A source as a back up. However, a 12V power supply means that I do not have to use a step-down circuit and simply plug it into the Arduino.

4.4.3 DC Motor Drive Circuit
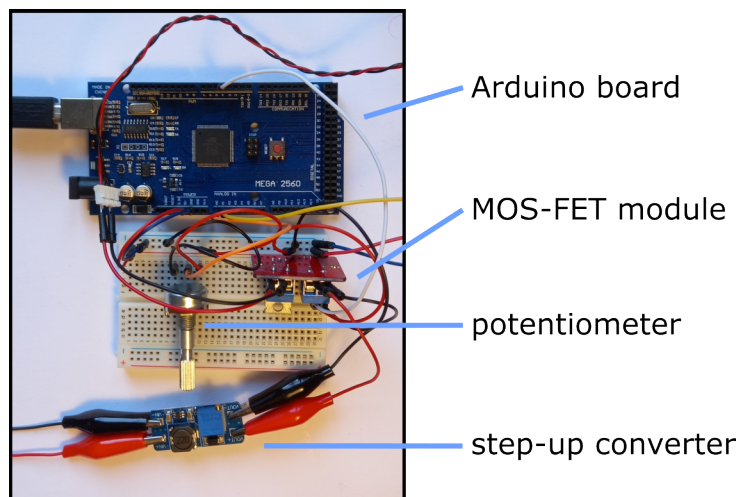
Figure 4.19: DC motor drive circuit when prototyping on a breadboard

Every different rotor on the machine suffers from different amounts of friction and other passive resistances. In order to set the correct speed for the balancing of the rotor, I decided to add a MOS-FET module. Based on an input PWM signal from the Arduino, the MOS-FET changes the average voltage on the input of the DC motor. Change in the input voltage means a change of power, which increases or decreases the RPM of the motor and the rotor.

Since 12 volts from my power supply is barely enough to start the DC motor, I decided to add a step-up module with MT3608 chip capable of increasing the input voltage to the MOS-FET module to 28 volts. [36]

In order to manually set the speed of the rotor, I decided to add a potentiometer. The potentiometer is attached to an analog input of the Arduino, which reads the

position of the potentimeter, and in the code it maps it to the PWM output. The inspiration for the whole wiring can be found in [37]. You can see the breadboard version of this circuit in Fig. 4.19.

Chapter 5

Troubleshooting and Data Acquisition Enhancement

After 3D printing or acquiring all the necessary components in Chapter 4, the machine can be assembled. In this section, I take the first steps in order to obtain stable and reasonable data. In order to do so, I need to mitigate several issues, which are based on both the mechanic and the electronic side.

5.1 Resolving Unwanted Rotor Motion

During the first tests of the machine, I came across a first of several unpleasant phenomena. For calibration of the machine, I use a metal rod, which is precisely manufactured, has a smooth surface, and therefore should be perfectly balanced. But the pendulums were visibly shaking even with the naked eye. Moreover, the rotor was pushed to one side seemingly for no reason. I tried to tackle this issue by eliminating the following suspects:

1. I changed the pulley on the DC motor to make a better fit, also making it smaller. After the change there was a slight improvement. The system (which was not supposed to vibrate at all) was less vibrating.

2. The motor mount (see Fig. 4.6) was not firm enough, so when pulled by the rubber band, its axis of rotation inclined upward. I recreated the whole attachment of the motor in order to be tightly fixed to its 3030 profile. This

gives my machine a more quality look by being sturdier and possibly more durable against unwanted vibrations. However, it still did not solve the issue of the rotor being pulled to the side along the x-axis.

3. The point of contact between the rotor and the support consists of two metal bearings. The rotor does not sit on the bearings too precisely, so I suspected that there might be a "screw" effect at play here. My suspicion got stronger when, by rotating the rotor to the other side, it was also moving along the x-axis to the other side. I tried to verify this hypothesis by replacing the bearings with a simple friction contact. For the contact material I tried PLA, steel wire and material plastic tubes are made out of. Unfortunately, the effect did not diminish, so this hypothesis was invalid. Yes, it was pushed to the side to a lesser extent, but because of friction, it was spinning much slower, so I believe this was the only reason.

4. Since the supports are moveable on the frame, are 3D printed and therefore they are neither too straight nor firm, I tried to manipulate them - straightening them - while balancing to see if there is any difference. Unfortunately, there was no difference.

Since I was unable to solve this problem, I tried to at least cure the symptoms. By misaligning the rotor in the z-axis, I can achieve a position of the rotor where it does not shift to the side. However, this will interfere with the measured data, and later I should investigate if such interference is negligible or not. Then I improved the axial holder 4.7 by replacing the last PLA segment with a metal bearing. I also considered teflon, but it is quite hard to come by (and relatively expensive). Eventually after the change of an accelerometer, I used a piece of teflon plate because another reason for its use appeared (see the end of Section 5.2.2). Quite surprisingly, this almost mitigated the whole problem.

▉ 5.2 Combining Sensors and Motor Drive

In this section, I describe the software and hardware difficulties that disabled the microcontroller. These were caused by combining the data sampling and the PWM modulation drive of the DC motor.

Figure 5.1: Nonsense data from the first measurements

5.2.1 MPU-6050

After the first assembly of the machine as a whole, that is, connecting all the necessary parts and adding an accelerometer to one pendulum - I undertook a first measurement of data. By connecting the accelerometer via the I^2C serial bus to an Arduino Mega, I could see the data directly in the arduino IDE [38]. For that purpose, I used a simple sketch from [27] to plot the real-time data on my screen in the serial plotter.

The first data measurements were also quite unsatisfactory. After attaching one of the pendulums to the accelerometer, I punched it in order to determine if the accelerometer works. I obtained a good-looking diminishing sine wave. However, when starting the actual balancing procedure, I gained very unclear data (Fig. 5.1). I tried changing the filter bandwidth of the accelerometer and sampling delay, but to no avail. After consulting with my thesis supervisor, I needed to look at several aspects that could negatively influence my results. However, first I needed to determine the RPMs of the machine in order to better troubleshoot. For that, I decided that I needed to add the RPM sensor first. Also, in order to have the speed of the DC motor under control, I added the MOSFET module to manually change the RPMs.

The DC motor was then driven by a PWM signal from the Arduino. Unfortunately, the frequency of the PWM was in an audible range, which caused loud high-pitched noises. The PWM signal is generated by Timer0 on the arduino's ATmega 2560 chip. By changing the parameters of this timer, I am able to increase the frequency of the PWM beyond the audible range. Unfortunately, this disrupted the millis() function, which is an ordinary function to use for timing events (for example, sample rate). The function then incorrectly measured the time (roughly 127.5x faster) because it uses the same Timer0. The simplest solution for me was to write another function, millis2(), where I divided the value by 127.5. One has to be careful when medling in time measuring. But after a brief consideration, I concluded that the deviation from the real time for me is less than $\tilde{0}.01\%$, which is definitely lower than that it could

influence my later calculations in any way. [35]

The greatest issue I came across was when I combined the two sensors and the motor drive, trying to get data out of it. I need to sample the accelerometer and phototransistor at at least 100 Hz (everything can be output this fast with some margin), which I tried to implement simply using the millis() function, which returns the microcontroller run time in milliseconds. This works, however, only for a while, then the microcontroller stops doing anything and "freezes" - it stops outputting values over the serial line, the control LED flashes, and you cannot even change the motor speed.

In fact, I never found exactly what the problem was, even after several consultations. But when diagnosing the situation, I found out this:

- As long as I was just reading analog and writing PWM, the setup ran arbitrarily long.

- If I plugged in and read only the MPU-6050, everything ran arbitrarily long.

- After loading the code, the program memory and the space for global variables were always only a few percent full.

However, I still could not find out what was the root cause of the problem, so I tried to implement it all in another way.

First I tried using different library for getting the data from MPU-6050. It behaved exactly the same. Then I tried using a different arduino board that I had - Arduino Uno - which has a different microcontroller. It also behaved exactly the same. Then I delved into the ATmega2560 microcontroller datasheet and set a timer with an interrupt to read the values. This also did not work. However, I believe that it added more certainty and "cleanliness" to the sampling process. You can see the setup of the Timer5 for sampling in Fig. 5.2.

▮ 5.2.2   ADXL335

Finally, I tried to approach the problem from a different angle. Since reading the analog values is relatively fast, I bought an accelerometr with analog output - ADXL335. Now I do not use any I2C serial bus and only read analog values, which makes it much easier for the microcontroller. However, the ADXL has to

```
44
45  ∨ void setup() {
46        // Setting the speed of the serial communication
47        Serial.begin(1000000);
48
49        // Deleting autoset values from the microcontroller for Timer5, it will be used for interrupt
50        TCCR5A = 0;
51
52        // Setting the prescaler of the 16-bit Timer5
53        TCCR5B |= B00000100;
54        TCCR5B &= ~B00000010;
55        TCCR5B &= ~B00000001;
56
57        // Resetting the Timer5; setting compare value for Timer5
58        TCNT5 = t5_load;
59        OCR5A = t5_comp;
60
61        // Enabling the compare register A for TIMER5
62        TIMSK5 = _BV(OCIE5A);
63
64        delay(50);
65
66        // Enabling global interrupts
67        sei();
68
69  }
```

Figure 5.2: Code example of the Arduino IDE - setup function for the Mega 2560

be calibrated first to give the correct values. The other issue is that the analog value coming from the ADXL is always changing by cca 1 ms^{-2}, every few seconds without the accelerometer actually moving. This is a serious drawback, but with the ADXL, the whole setup runs without freezing.

Now I was able to obtain some data from a working setup, so I proceeded to lower the noise of the signal to get some cleaner data for later data processing and disbalance evaluation. First of all, I had to change the point of contact between the pendulum and the rotor. The bearings (see Fig. 4.4) on the pendulum created such a noise that any data obtained from the ADXL were heavily distorted. Although the pendulum seemed to swing quite nicely by the look of a mere eye.

I changed the bearings with a point of contact made of metal wire. The data obtained were much cleaner, but it was not able to run at a slower RPM because the friction was too high. Although it was spinning, it had a tendency to jam. Since my sample rate is not very high, I needed to make the rotation as smooth as possible.

Then I tried to make the contact point from the available plastic material, which was a piece of PVC pipe. It seemed to have lower friction with the metal rod. The results were similar to those for the metal wire, so it was not a possibility. At this point, the testing was already quite time-consuming. All the more for the fact that with every aforementiod option the rotor was always pushed strongly to one side at the measuring RPM.

Figure 5.3: The data obtained from ADXL, point of contact out of teflon

The last possible solution I had in mind was the use of teflon as a point of contact. I ordered a teflon plate on the internet, although it seemed to be quite similar to the PVC pipe. However, when I tried it out the noise was lower than before, and the "push-to-the-side" problem also quite diminished. The data obtained from the serial plotter resembled distantly a sine wave (see Fig. 5.3).

The signal in Fig. 5.3 is still distorted. According to the data sheet [39], for sampling at higher frequencies, the analog output should be fitted with the corresponding noise canceling capacitor. The GY-61 module with the accelerometer has already soldered some capacitors to the board. I could not find the official data sheet for this board, but according to some forums ([40]), it already has $0.1\mu F$ capacitors soldered to it. According to the data sheet for the ADXL, this value is ment for bandwidth of 50 Hz, which is too low for my use. Instead, I exchanged it for a 100 nF capacitor suitable for a bandwidth of 500 Hz.

Another suitable improvement would be to set the reference value of the analog input externally to 3.6 volts, which could increase the precision of the measured data. That is because the maximum value of the analog output of the ADXL accelerometer is around 3.6 volts.

Unfortunately, the PWM used to drive the motor causes so much noise on the board that it heavily influences the results of the analog input. Therefore, the ADXL 335 results undergo significant changes up and down every few seconds even if the sensor is at rest, almost up to 2.5 ms^{-2}. Without the PWM, the value at rest remains at its value, which only oscillates with an amplitude of 0.15 ms^{-2}. I also implemented a simple autocalibration of the machine, so the acceleration value in an idle state is always zero. However, for proper auto-calibration the accelerometers cannot be moving when turning on the machine (cca for 0.3 seconds).

I solved this problem by buying a whole module for PWM control, so the circuit

with the microcontroller is completely separated from the DC motor drive. This also enables the microcontroller to run faster because it has fewer tasks.

Chapter 6

Example of Use

When the machine is equipped with the correct components, it can be used for data output. The data output can be seen live, or, based on the data, unbalance can be determined.

6.1 Viewing Live Data

The first aim of this thesis is to use the balancing machine as a sensor application example during lectures. This can be done quite easily. Through the Arduino IDE, it can be seen in the Serial Plotter. Everything there is quite straightforward; see Fig. 6.1. These data were transferred live to the serial plotter, giving the students the opportunity to see the machine working and to see the data coming out of it directly; this situation is illustrated in Fig. 6.2.

6.2 Estimating Unbalance of a Test Rotor

To test the basic function of the model machine as a balancer, I use it in a simple case of measuring static unbalance. The model object is a metal rod that rests on the supports with several magnets in the middle, representing the simplest case of static unbalance (see Fig. 6.3 - I added a few magnets more later on, for clearer balancing data). I will estimate the unbalance based on the data from the machine
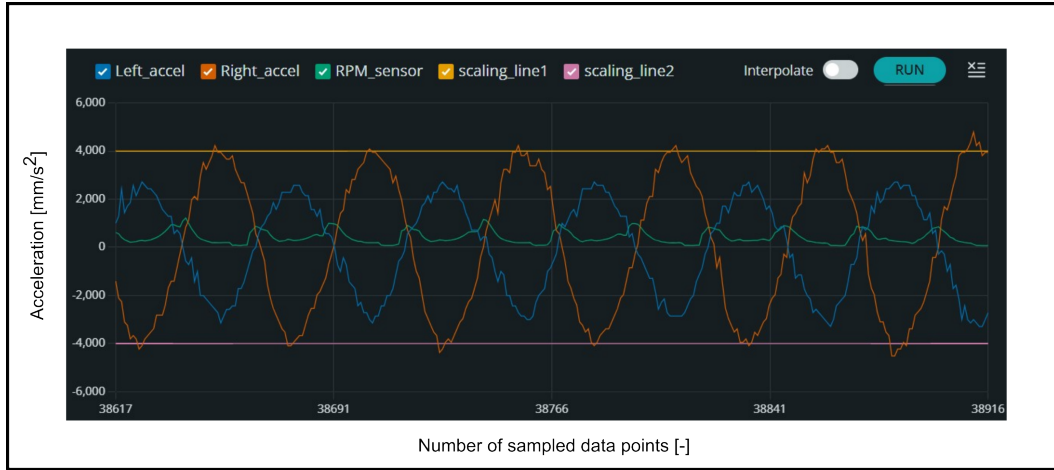
Figure 6.1: Live data from the Serial plotter

and compare it with the unbalance based on the known geometry and mass of the rotor and magnets.

6.2.1 Unbalance Based on Measured Data

By reading the amplitude of the two accelerometers (which should be the same in theory for this model scenario), we obtain the circular acceleration a_c. The circular acceleration is dependent on the distance r

$$a_c = r \cdot \omega^2 \;\; \rightarrow \;\; r = \frac{a_c}{\omega^2} \tag{6.1}$$

Where $\mathbf{a_c}$ is the circular acceleration, $\mathbf{r}$ is the distance from center of gravity to the rotation axis, ω is the angular speed.

Knowing the distance r and weighing the rotor, we can count the amount of unbalance:

$$U = m_r \cdot r = m_m \cdot r_m \tag{6.2}$$

Where $\mathbf{U}$ is the unbalance, $\mathbf{m_r}$ is weight of the rotor, ω is the angular speed, $\mathbf{m_m}$ is weight of the magnets, $\mathbf{r_m}$ is the distance of center of gravity of the magnets to
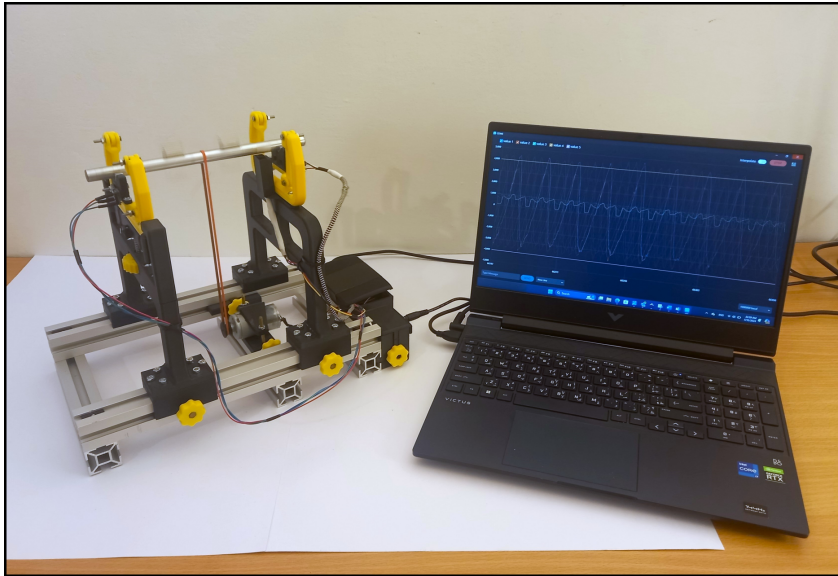
Figure 6.2: Example of use of the balancing machine with the Arduino Serial Plotter

the axis of rotation.

Regarding equation 6.2, the unbalance should be equal to the weight of the magnets multiplied by the distance of their center of gravity from the rotation axis. That is then the value with which I will compare it, further details in Subsection 6.2.2.

During testing, I found that the signal from the 100 nF input is considerably less noisy than the signal from the 10 nF capacitor. This was expected, but it seemed to be worth the trade-off, because lowering capacitance increases the bandwidth. Then the amplitude of the signal does not decrease significantly [39]. However, the noise from the 10 nF capacitor signal makes the data too incoherent for any serious further processing. See the comparison in Fig. 6.4, the sampling rate and the angular speed of the rotor are the same or very similar. The 10 nF capacitor data have signal in opposite phases (because of the orientation of the sensor), it has no influence on the amount of noise.

That is why I will be using the 100 nF data to calculate the unbalance. 100nF supposedly gives me ([39]) the bandwidth of 50 Hz, whereas my sampling rate is at this point 250 Hz. Therefore, I am curious about the results.

It is of great importance to measure the acceleration when the RPM of the rotor is different by a safe margin from the natural frequency of the pendulum. We can obtain the formula for the natural frequency from equation 3.7, for the balancing
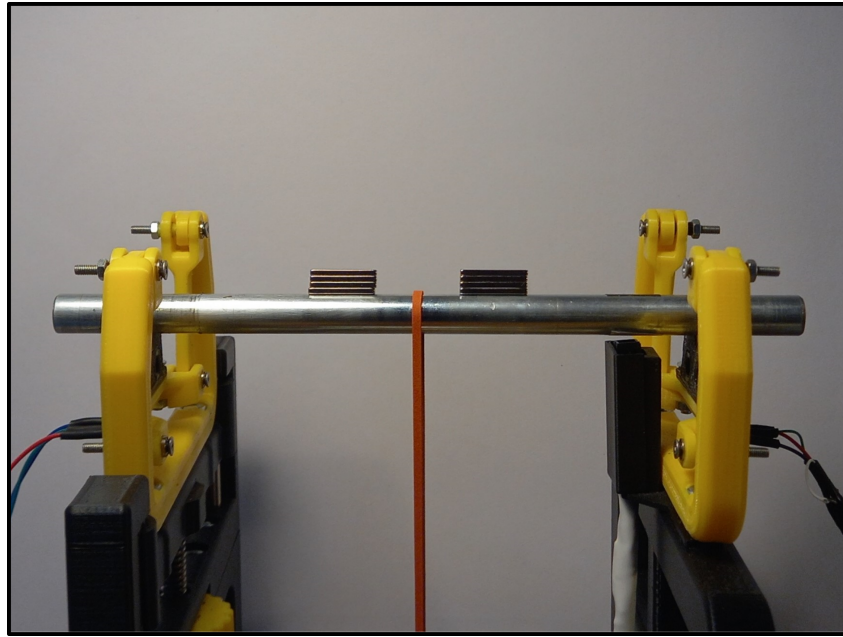
47

Figure 6.3: Testing rod with magnets causing unbalance

machine it is:

$$\Omega_0 = \sqrt{\frac{g}{l}} = \sqrt{\frac{9.81}{0.05}} = 14 \ rad/s = 2.23 \ Hz \tag{6.3}$$

In Fig. 6.5 are the data that I used to estimate the imbalance. I calculated an average value of the peaks, which gave me an acceleration of $\mathbf{a_c = 4.25/ms^{-2}}$. The RPM based on the data is $f = 8.5 Hz$, which should be far enough from the natural frequency of the pendulum, so it should not be influenced by any resonance. Writing the result into an equation gives me the desired distance r:

$$r = \frac{a_c}{\omega^2} = \frac{a_c}{(2\pi f)^2} = \frac{4.25}{(2\pi \cdot 8.5)^2} = 1.5 \ mm \tag{6.4}$$

The weight of the rotor is $m_r = 217 \ g$, therefore, the unbalance is:

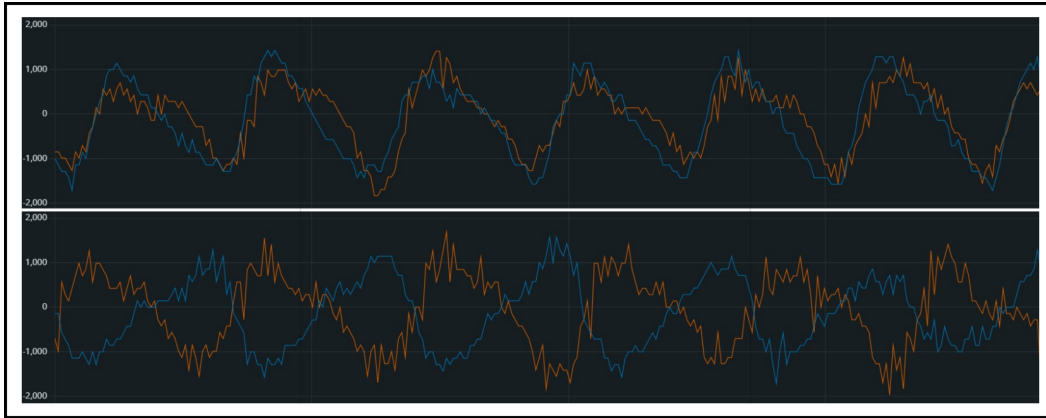$$U = m_r \cdot r = 217 \cdot 1.5 = 325.5 \ g \cdot mm \tag{6.5}$$

Figure 6.4: Noise comparison of data output, 100 nF (top) and 10 nF (bottom) capacitors
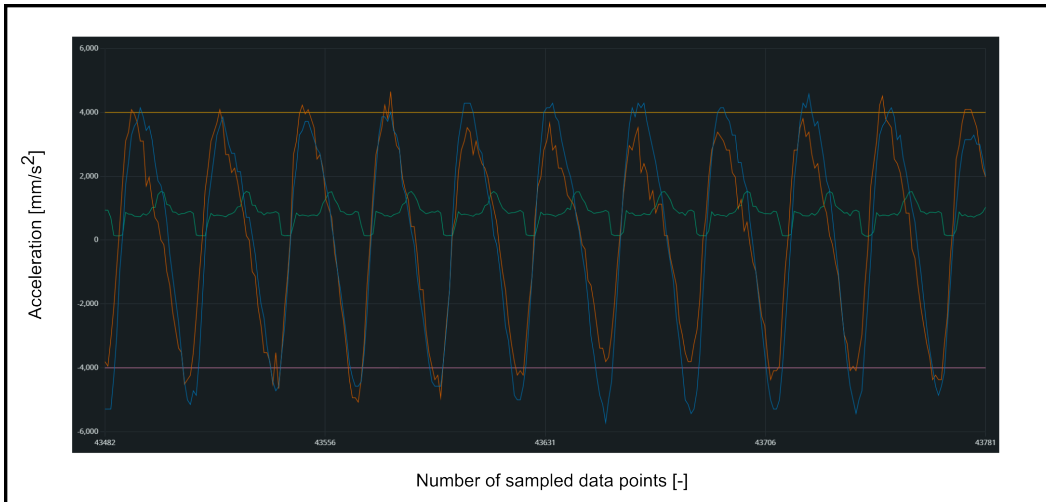

Figure 6.5: Data used for the unbalance calculation

◼ 6.2.2 Unbalance Based on Geometry

By measuring the distance of the center of gravity of the magnets from the rotation axis and weighing them, I obtained these values: $m_m = 21 \ g$ and $r_m = 13.25 \ mm$. Therefore, the real unbalance of the whole rotor is:

$$U_{real} = m_m \cdot r_m = 21 \cdot 13.25 = 283.5 \ g \cdot mm \tag{6.6}$$

That means that the balancing machine gave **14.81%** greater value of unbalance than it is in reality. Taking into account that the larger noise-reducing capacitor is used, the unbalance from the balancing machine data was expected to be smaller
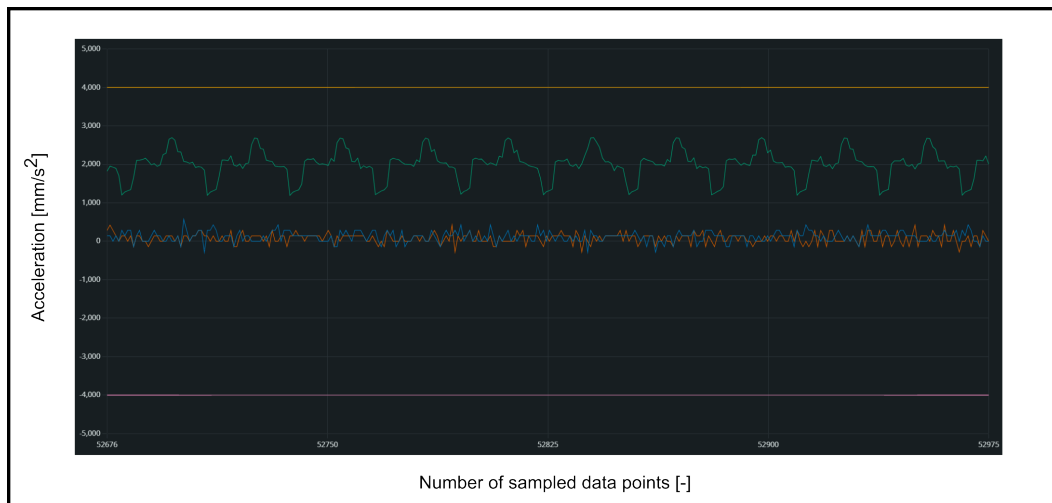
Figure 6.6: Data of a balanced rotor, i.e. without magnets, same frequency as in Fig. 6.5

than the real value. It is probably a matter of further calibration. However, this result shows that the balancing machine works in the same ballpark as the real values.

6.3 Balancing Dynamic Unbalance

In order to balance dynamic unbalance of the rotors, more complex mechanics come into play, which is unfortunately beyond the scope of this work. The proposed way to determine the dynamic unbalance would be to calculate the position of the natural axis of rotation based on the different amplitudes and phases of acceleration on both sides of the machine. In order to achieve this, I believe that a higher level of precision of components of the machine should be achieved, which is difficult to obtain by using simple PLA 3D printing.

Chapter 7

Conclusion

The aim of this thesis is to construct a balancing machine. The machine has two uses; the main one a is as an example of sensor application during lectures, the other is to determine the unbalance of the rotors.

In the first chapters, theoretical background on balancing machines and balancing is set, in order to give a correct direction for the design. In the following chapters, the construction process of the balancing machine and component selection are described in detail, with an emphasis on 3D printing. The updates and modification of the components are described, and their possible positive effect on the machine is explained. More complicated issues during the assembly and initial run of the machine are also outlined. Several solutions for these complications are proposed, eventually settling for the one that provides the best data. Finally, the application of the machine is shown, and the results are evaluated.

The machine is able to perform as an example of sensor application during lectures. Thanks to the serial plotter in the Arduino IDE it shows the signal live quite easily. The students can see an example of use of accelerometers and possible solution of the RPM sensor. The live chart from the machine can be seen in Fig. 6.1. In addition, the machine is lightweight and portable, which makes it suitable for traveling to lectures and back. Mainly, it has the correct size to measure the rotors specified by the assignment.

The machine is also tested to determine how well it could be used in a workshop or laboratory. In a simple case of static balancing, the balancing machine showed a **14.81%** deviation of unbalance from the real unbalance value. That signals that the balancing machine gives out meaningful data, but would still need more calibration
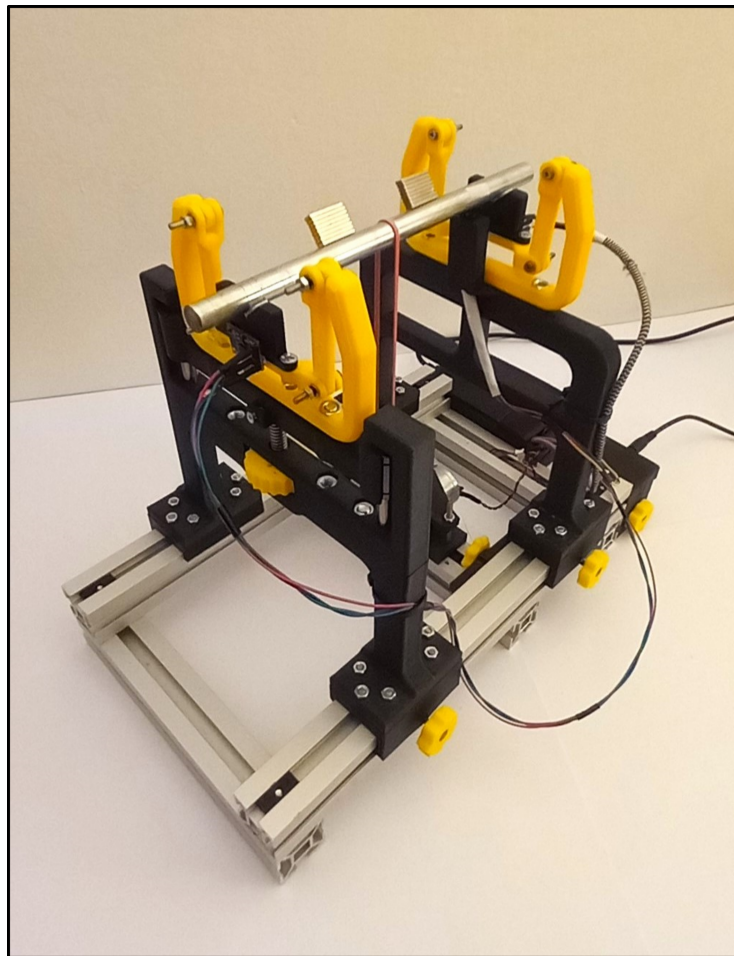
Figure 7.1: The balancing machine in its final form

or possible alteration in order to work more precisely. In that case, the machine could be able to balance even dynamic unbalance, but that is beyond the scope of this work. The latest version of the balancing machine before submitting this thesis can be seen in Figs. 7.1, C.1, C.2, and C.3.

Appendix A

Bibliography

[1] Osami Matsushita et al. *Vibrations of Rotating Machinery: Volume 1. Basic Rotordynamics: Introduction to Practical Vibration Analysis*. English. 1st ed. Vol. 16. Tokyo: Springer Japan, 2017. ISBN: 2198-350X.

[2] PRECIBALANCE. *Basics of balancing and balancing machines [online]*. [cit. 11/17/2023]. URL: https://precibalance.com/basics-of-balancing-and-balancing-machines/.

[3] Balancing Machines JP. *belt-drive balancing machine: operation steps YouTube [online]*. 1/12/2021 [cit. 11/19/2023]. URL: https://www.precibalance.com/gfaq.pdf.

[4] MPE 66. *Electric Motor Rotor Dynamic Balancing in 15 Minutes YouTube [online]*. 3/15/2023 [cit. 11/19/2023]. URL: https://www.precibalance.com/gfaq.pdf.

[5] PRECIBALANCE. *Frequently Asked Questions on Dynamic Balancing Machines and Answers [online]*. [cit. 11/19/2023]. URL: https://www.precibalance.com/gfaq.pdf.

[6] PRECIBALANCE. *Balancing Machines and Balancing Instrumentation [online]*. [cit. 11/18/2023]. URL: https://precibalance.com/.

[7] JP-BALANCER. *Products [online]*. [cit. 11/19/2023]. URL: https://www.jp-balancer.com/product/.

[8] DIRECT INDUSTRY. *Horizontal balancing machine 9K716 [online]*. [cit. 11/19/2023]. URL: https://www.directindustry.com/prod/vneshtorg-micron-ltd/product-63357-1575868.html.

[9] PRECIBALANCE. *Balancing Instrumentation and Balancing Machine Upgrades [online]*. [cit. 11/18/2023]. URL: https://precibalance.com/balancing-instrumentation-and-balancing-machine-upgrades/.

[10] PRECIBALANCE. *Vertical Balancing Machine [online]*. [cit. 11/19/2023]. URL: `https://precibalance.com/vertical-balancing-machine/`.

[11] EBS. *VR50-2P-AM [online]*. [cit. 11/19/2023]. URL: `https://www.ebs-balancing.com/machine-type/vr50-2p-am-vertical-balancing-machine/`.

[12] JP-BALANCER. *Axial Flow Fan Balancer [online]*. [cit. 11/19/2023]. URL: `https://www.jp-balancer.com/product/axial-flow-fan-balancer/pha5.html`.

[13] PRECIBALANCE. *Pump Impeller Balancing Machine [online]*. [cit. 11/19/2023]. URL: `https://precibalance.com/pump-impeller-balancing-machine/`.

[14] PRECIBALANCE. *Horizontal Balancing Machine [online]*. [cit. 11/19/2023]. URL: `https://precibalance.com/horizontal-balancing-machines/`.

[15] Vladimír Stejskal, Stanislav Stejskal, and Jaromír Brousil. *Mechanika III*. Czech. Vyd. 2. Praha: ČVUT, Strojní fakulta, 2001. ISBN: 9788001023884;8001023885;

[16] Derek Norfield. *Practical Balancing of Rotating Machinery*. English. 1st ed. San Diego: Elsevier, 2006;2011; ISBN: 9781856174657;1856174654;0080459382;9780080459387;

[17] JOHN MAHER RACING. *Project 2110 – Part 4: Dynamic Balancing*. 1/13/2013 [cit. 3/24/2024]. URL: `https://johnmaherracing.com/2013/01/project-2110-part-4-dynamic-balancing/`.

[18] IgorM92. *Simple Rotor Balancing Machine [online]*. [cit. 3/24/2024]. URL: `https://www.ic-components.com/files/bb/MPU-6050.pdf`.

[19] Joseph Flynt. *Polylactic Acid (PLA): The Environment-friendly Plastic [online]*. 11/9/2017 [cit. 3/25/2024]. URL: `https://3dinsider.com/what-is-pla/`.

[20] Prusa Research. *Prusaslicer*. Version 2.7.2. 2024. URL: `https://ultimaker.com/software/ultimaker-cura/`.

[21] Ultimaker. *Ultimaker Cura*. Version 5.6.0. 2024. URL: `https://ultimaker.com/software/ultimaker-cura/`.

[22] Autodesk. *Autodesk Inventor Professional 2024*. 2024. URL: `https://www.autodesk.com/products/inventor/overview?term=1-YEAR&tab=subscription`.

[23] SCULPTEO. *What is an STL file? - Everything About STL files*. [cit. 3/25/2024]. URL: `https://www.sculpteo.com/en/3d-learning-hub/create-3d-file/what-is-an-stl-file/`.

[24] MABUCHI MOTORS. *RS-455PA*. [cit. 4/15/2024]. URL: `https://datasheetspdf.com/pdf-file/1106967/MabuchiMotor/RS-455PA/1`.

[25] Jacob Fraden and PROQUEST EBOOK CENTRAL. *Handbook of modern sensors: physics, designs, and applications*. English. Fifth;5;5th 2016;5th 2016; Cham: Springer, 2016;2015; ISBN: 3319193031;9783319193038;9783319193021;3319193023;

[26] Martin Novák. *Introduction to sensors for electrical and mechanical engineers*. English. First. London;New York;Boca Raton; CRC Press, Taylor Francis Group, 2020. ISBN: 9780367518219;036751821X;

[27]   LAST MINUTE ENGINEERS. *Interface MPU6050 Accelerometer and Gyroscope Sensor with Arduino [online]*. 5/16/2022 [cit. 5/12/2023]. URL: `https://lastminuteengineers.com/mpu6050-accel-gyro-arduino-tutorial/`.

[28]   INVENSENSE INC. *MPU-6000 and MPU-6050 Product Specification Revision 3.4 [online]*. 8/19/2013 [cit. 9/11/2023]. URL: `https://www.ic-components.com/files/bb/MPU-6050.pdf`.

[29]   Yunfan Zhang et al. "Investigation of Acoustic Injection on the MPU6050 Accelerometer". In: *Sensors* 19.14 (2019). ISSN: 1424-8220. DOI: `10.3390/s19143083`. URL: `https://www.mdpi.com/1424-8220/19/14/3083`.

[30]   INSTRUMENTATION TODAY. *Piezoelectric accelerometer [online]*. 8/2011 [cit. 12/5/2023]. URL: `https://www.instrumentationtoday.com/wp-content/uploads/2011/08/Piezoelectric-Accelerometer.jpg`.

[31]   DEWETRON. *Strain Gauges [online]*. 4/1/2022 [cit. 7/12/2023]. URL: `https://www.dewetron.com/2022/04/strain-gauges/`.

[32]   VISHAY. *Reflective Optical Sensor with Transistor Output*. 4/5/2000 [cit. 4/30/2024]. URL: `https://www.alldatasheet.com/datasheet-pdf/pdf/26332/VISHAY/CNY70.html`.

[33]   HETPRO. *CNY70 Reflection optical sensor with Arduino*. [cit. 4/30/2024]. URL: `https://hetpro-store.com/TUTORIALES/cny70-sensor-optico/`.

[34]   Tomisin Olujinmi. *Arduino Mega vs. Uno: Which One Should You Use?* 6/11/2023 [cit. 5/1/2024]. URL: `https://www.makeuseof.com/arduino-mega-vs-uno-which-one-is-best/#:~:text=The%20Arduino%20Uno%20is%20a%20better%20option%20for,hobbyists%20who%20have%20already%20gotten%20their%20feet%20wet.`.

[35]   MICROCHIP. *ATmega640/V-1280/V-1281/V-2560/V-2561/V*. 2/2014 [cit. 5/1/2024]. URL: `https://ww1.microchip.com/downloads/aemDocuments/documents/OTH/ProductDocuments/DataSheets/ATmega640-1280-1281-2560-2561-Datasheet-DS40002211A.pdf`.

[36]   GME. *Step-UP měnič nastavitelný 2 - 28V / 2A s MT3608*. [cit. 5/1/2024]. URL: `https://img.gme.cz/files/eshop_data/eshop_data/10/751-903/dsh.751-903.1.pdf`.

[37]   Amir Mohammad Shojaei. *Interfacing IRF520 MOSFET Driver Module (HC-MODU0083) with Arduino*. [cit. 5/1/2024]. URL: `https://electropeak.com/learn/interfacing-irf520-mosfet-driver-module-switch-button-hcmodu0083-with-arduino/f`.

[38]   Arduino. *Arduino IDE*. Version 2.3.2. 2024. URL: `https://www.arduino.cc/en/software`.

[39]   ANALOG DEVICES. *Small, Low Power, 3-Axis ±3 g Accelerometer ADXL335*. 2008 [cit. 5/24/2024]. URL: `https://pdf1.alldatasheet.com/datasheet-pdf/download/250056/AD/ADXL335.html`.

[40]   HOBBY COMPONENTS. *GY-61 ADXL335 Triple Axis Accelerometer Module (HCMODU0070)*. 9/20/2014 [cit. 5/30/2024]. URL: `https://forum.hobbycomponents.com/viewtopic.php?t=1736`.

Appendix B

Symbols and Abbreviations

Symbol	Unit	Meaning
F	$[N]$	Force
F_c	$[N]$	Centrifugal force
F_l	$[N]$	Reaction force in the left support
F_r	$[N]$	Reaction force in the right support
F_g	$[N]$	Gravitational force
m	$[kg]$	Mass
m_r	$[kg]$	Mass of the rotor
m_m	$[kg]$	Mass of the magnets
I	$[kg \cdot m^2]$	Moment of inertia
a	$[ms^{-2}]$	acceleration
a_c	$[ms^{-2}]$	Centrifugal acceleration
a_h	$[ms^{-2}]$	Horizontal acceleration
a_{hg}	$[ms^{-2}]$	Horizontal acceleration caused by gravity
g	$[ms^{-2}]$	Gravitational acceleration
U	$[kg \cdot m]$	Unbalance
r	$[m]$	Distance of COG from axis of rotation
r_m	$[m]$	Distance of COG of magnets from axis of rotation
$a; b$	$[m]$	Distances from supports
ω	$[rads^{-1}]$	Angular acceleration
Ω_0	$[rads^{-1}]$	Natural frequency
RPM	$[min^{-1}]$	Rotations per minute
ϕ	$[rad]$	Angle of deflection of a pendulum
t	$[s]$	Time
l	$[m]$	Length of the pendulum
d	$[rad]$	Deflection of the pendulum
k	$[N/mm]$	Stiffness of a spring
x	$[mm]$	Displacement of a spring
PLA		Polyactic acid; filament material
CAD		Computer-aided design
MPU-6050		Accelerometer with I2C serial bus
ADXL335		Accelerometer with analog output
MOS-FET		Type of a transistor
MT3608		Chip on a voltage step-up module
PWM		Pulse width modulation
DC		Direct current
LED		Light emitting diode
PVC		Polyvinyl chloride
GY-61		Module board with ADXL accelerometer
IDE		Integrated development environment

Appendix C

Additional Photos

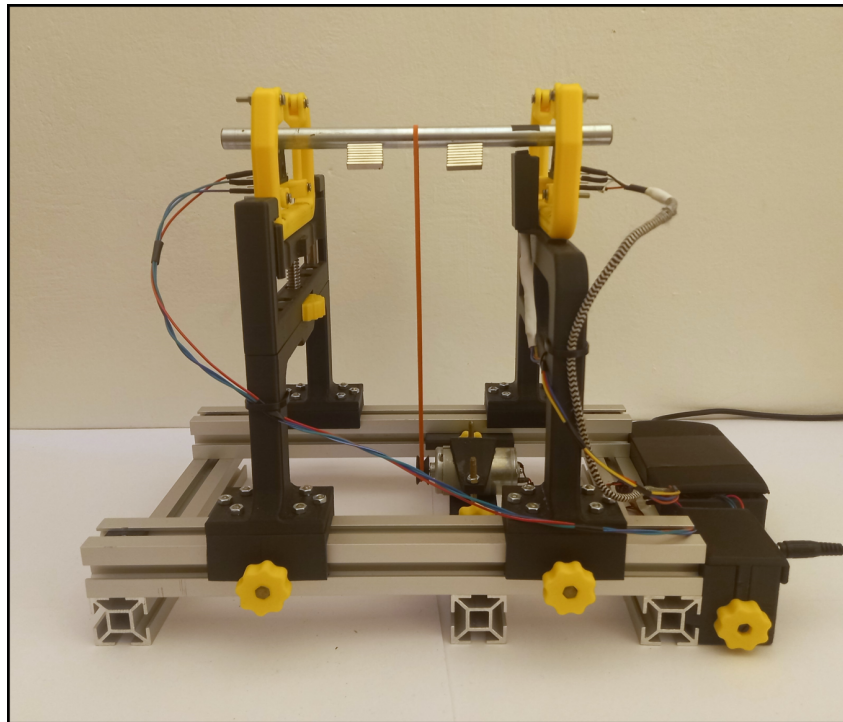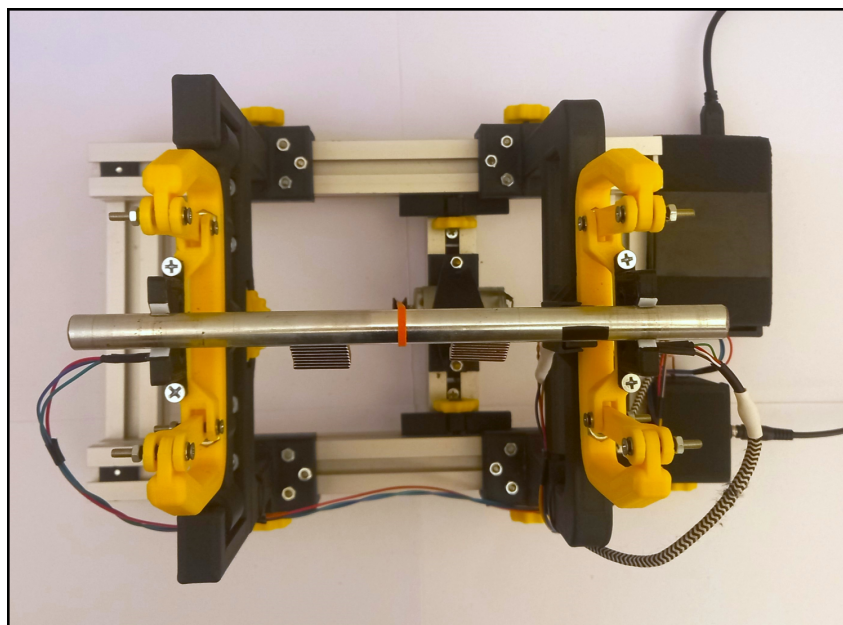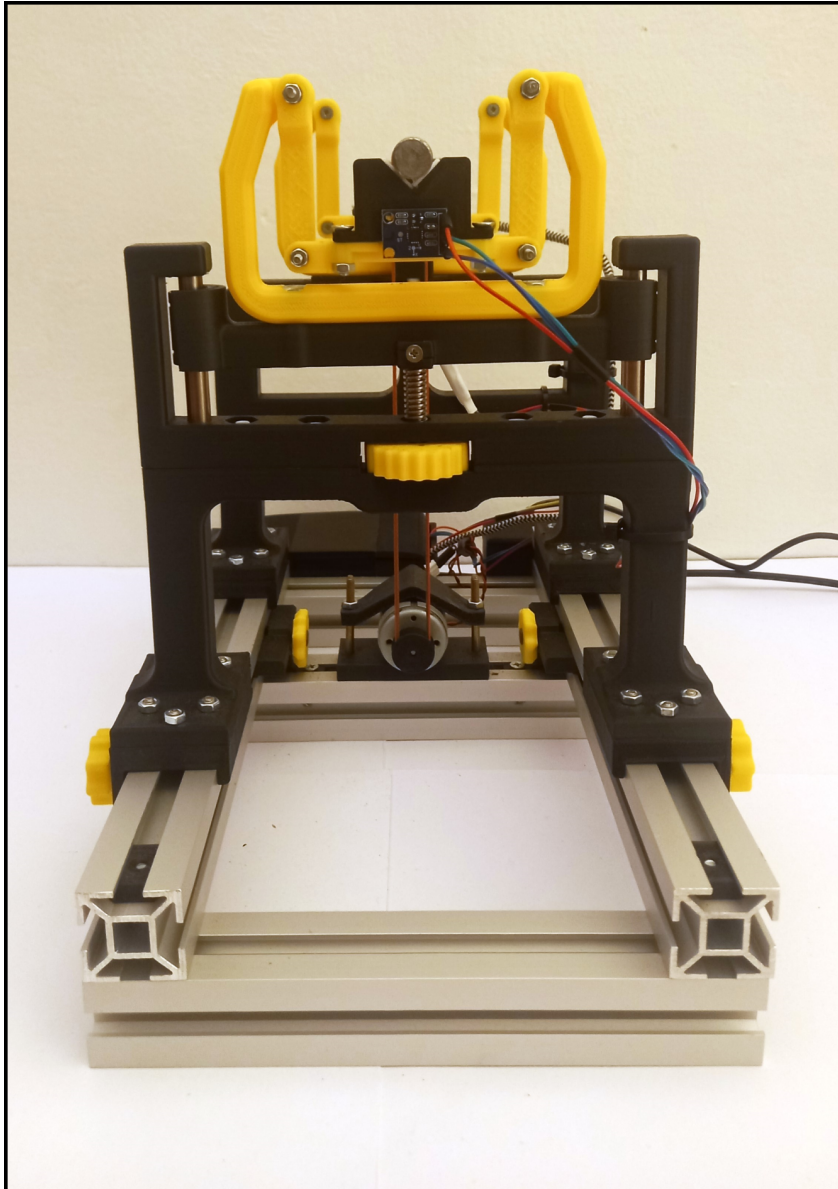**Figure C.1:** The balancing machine - side view

Figure C.2: The balancing machine - top view

60

Figure C.3: The balancing machine - front view