Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

# Stabilizing Flight

Group project

Samuel Gyger, Andreas Steger, Philippe Wenk

gygers@student.ethz.ch, stegeran@ethz.ch, wenkph@student.ethz.ch

Distributed Computing Group
Computer Engineering and Networks Laboratory
ETH Zürich

**Supervisors:**
Pascal Bissig, Jara Uitto
Prof. Dr. Roger Wattenhofer

June 28, 2013

# Abstract

Recently, a new way of taking self-portraits has come up. Throwing a phone camera up into the air, and have it take a picture from above can lead to nice photos. Unfortunately, there are two major problems. The first is that the picture blurs because of the phone rotation which is naturally generated when throwing it up by hand. The second is that the camera does not always face the ground even if the rotation was stopped. The goal of this project is to eliminate these problems. Our approach to these problems is to compensate the rotation with three fixed flywheels. To find out which control algorithms works well on the chosen design a simulation was needed. The second step was to find a controller for three dimensions. The PI Controller works for a limited set of throwing parameters. The result in real life shows this limitation is enough for everyday use. It is realistic to improve this new way of taking photos by stabilizing the flight with flywheels. The next step is to adapt the controller to control the orientation of the camera.

# Contents

Chapter 1

# Introduction

If objects are flying in the air there is not an orientation that is more likely to be assumed then the others.[1] This means you can't build a static object that always reaches a certain orientation after some time.

The change of the orientation is described by angular velocity. Stabilizing the angular velocity of objects is a well studied field in control theory. One example are satellites. They need to be accurately oriented to transmit data or take pictures of certain geographic regions. An object in space can be stabilized by using thrusters but the fuel storage is limited. Therefore the orientation is often controlled by using the conservation of angular momentum through gyroscopes or turning wheels.

A similar situation to an object in space is throwing something into the air. Mike Larson throws cameras into the air to create group photos from above.[2] On Flickr a whole group is established that publishes photos taken by throwing the camera into the air.[3]

With so many and cheap smartphones coming to the market it is safer to use them for taking such pictures. Small phone cameras need much more exposure time. The aperture is open longer so images blur more easily. Hence it takes several attempts to get a sharp picture.

Using the techniques from astronautics the object thrown into the air should stop its rotation as fast and effectively as possible. While satellites are multi million dollar projects the goal is to produce a smaller and much cheaper prototype too. (see the result Figure 1.1).

[1] ignoring air resistance
[2] http://www.youtube.com/v/F35Sn5gAhDg
[3] group "cameratoss" [http://www.flickr.com/groups/cameratoss/pool/]

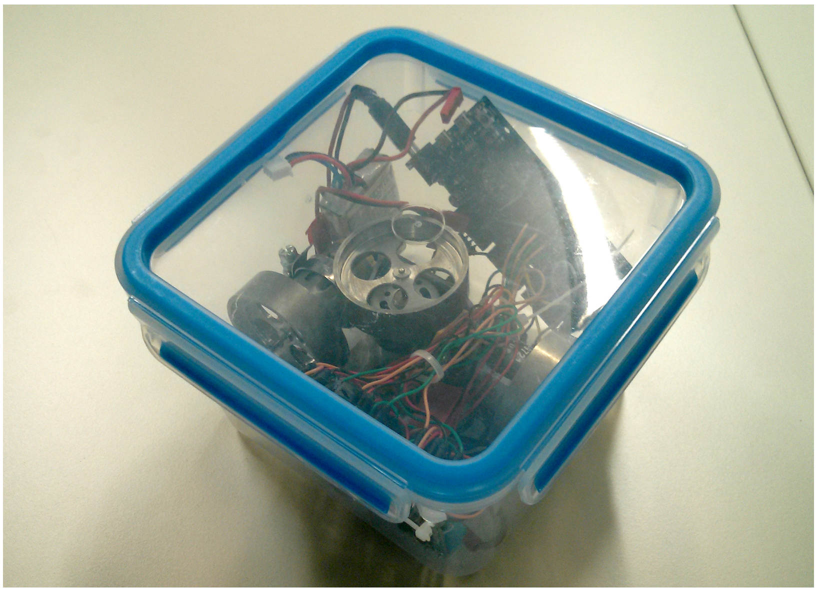


Figure 1.1: The build of the stabilizing box

Chapter 2

# Analysis of the Problem

## 2.1 Characteristics of a flying Object

A non deformable object can be characterized as a masspoint with different characteristica. The basic values are the position in space and the weight. This is enough to know the basic interaction between the object and its surrounding (ignoring Aerodynamics). The basic system is then described by the well-known formula

$$F = m \cdot \ddot{\vec{x}} \tag{2.1}$$

For this project the position is not interesting but only how fast the object is spinning. Therefore the rotating speed $\omega$ and the inertia tensor $I^{(cog)}$ are added to the masspoint's properties. $(cog)$ denotes that it is calculated in the center of gravity. The angular momentum $L$ is given by the formula

$$\vec{L} = I\vec{\omega} \tag{2.2}$$

$L$ is preserved in the inertial reference frame and follows the equation

$$\dot{\vec{L}} = \vec{M} \tag{2.3}$$

where $M$ is the external Torque applied on the object. It is useful to transform Equation (2.3) into a body fixed reference frame by using Equation (2.4) [1, see Eq.(1.81)]

$$\frac{d}{dt}\vec{f} = \frac{d}{dt}^{(cog)}\vec{f} + \vec{\omega} \times \vec{f} \tag{2.4}$$

where $\frac{d}{dt}^{(cog)}\vec{f}$ denotes the change of $f$ as seen in the reference frame. Applying Equation (2.4) to Equation (2.3) we get Equation (2.5), a differential equation.

$$\dot{\vec{L}} = \dot{\vec{L}}^{(cog)} + \vec{\omega} \times \vec{L} = \vec{M} \tag{2.5}$$

$\vec{L}^{(cog)}$ denotes $L$ in a body fixed coordinate system at the center of gravity.

## 2.2 Control Moment Gyros

Angular momentum can be controlled by applying external torque (i.e. by using thrusters) on the object (see section 2.1). However the goal of this project is to develop a stabilizing system without external torques, there are other approaches needed. If the external torque $\vec{M}_{ex} = 0$ then the total angular momentum $\vec{L}_{tot}$ inside the whole system stays constant. This principle stays valid for several bodies combined with joints. So the visible torque $\vec{L}_{box}$ can be reduced by redistributing $\vec{L}_{tot}$ on controlling constructs (i.e. wheels or gyros). Therefore the target is that, the sum of the angular momentums of the controlling constructs are equal to $\vec{L}_{tot}$. Like this, the absence of external momentums would imply that $\vec{L}_{box} = 0$. This idea has been used to stabilize satellites for a long time. [2]

The *Control Moment Gyro* (CMG) is the first concept discussed in this report. We see applications of this principle in well-known experiments like the turning office chair or precession of a turning bicycle wheel.[1]

To understand the principle of a CMG, it is very helpful to look at a spinning flywheel. Due to angular momentum preservation, the orientation of rotating wheels cannot be changed like the one of a non-spinning object. Assuming the rotational speed stays constant, by changing the direction of the rotational axis one changes the angular momentum of the system (see Figure 2.1). Due to Equation (2.3), there must be a torque responsible for that change.

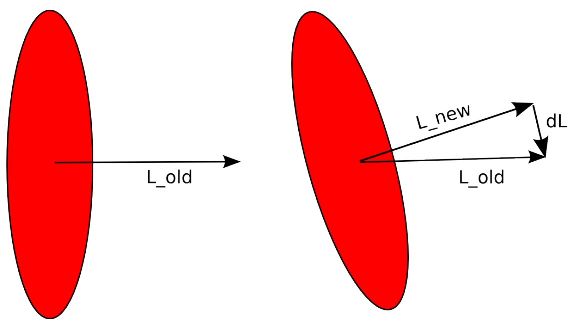



Figure 2.1: Change of Angular momentum

This momentum can be used to conquer the angular momentum of a system connected to the gyro.

[1] Walter Lewi, MIT open courseware [http://www.youtube.com/watch?v=NeXIV-wMVUk]

Attitude stabilization of satellites with CMGs is a very popular theme in control theory. Although they are quite accurate, there are several difficulties. The momentum that can be produced changing the direction of a spinning wheel very much depends on the current position. So the next action is not only determined by the angular speed of the system, but also by the current position and speed of the gyro wheels. [3] [4] Due to the rather complicated suspension and motorization, building a CMG is rather difficult. Therefor, they are mostly very heavy and costly. So reaction wheels are often preferred, although one could save energy and mass using gyros.[5]

## 2.3 Fixed Flywheels

Fixed Flywheels are used in satellites to steer the orientation. Similar ideas can be used for objects on earth.

The ideas from [6] that apply for satellites can also be used for objects on earth. In the system the wheels are installed parallel to the axes of the Body fixed coordinate system. The angular momentum of each wheel $L_{w_i}$ is given by $L_{w_i} = I_{w_i}(\vec{\omega_{w,i}} + \vec{\omega_o})$, where $I_{w_i}$ is the Moment of Inertia tensor of the wheel and $\omega_{w,i}$ the rotation speed of the wheel and $\omega_o$ is the rotation speed of the object. It is important to note that $L_{w_i}$ depends on the location. To move $L$ to a different position we use eq. (2.6), where the integral is evaluated with a coordinate system at the new point.

$$\vec{L}' = \vec{L} + (\vec{x}' - \vec{x}) \times \vec{p} \tag{2.6}$$

The angular momentum $L_o^{(cog)}$ of the device without the wheels in the center of gravity is described by Equation (2.7).

$$L_o^{(cog)} = I_o^{(cog)} \vec{\omega_o} \tag{2.7}$$

The angular momentum of each wheel in the center of gravity of the objecct is described by Equation (2.8). $m_{w_i}$ denotes the mass of the wheel and $x_{w_i}$ is the vector to the center of gravity of the wheel in body fixed coordinates. $I_{w_i}$ stands for the inertia tensor of the wheel and is diagonalized when the wheels are installed in direction of an axis of the body fixed coordinates.

$$L_{w_i}^{(cog)} = I_{w_i}(\vec{\omega_i} + \vec{\omega_o}) - m_{w_i} \cdot (x_{w_i} \times \omega_o \times x_{w_i}) \tag{2.8}$$

Defining $I_{tot}$ in Equation (2.9), where $I_{wheels}^{(cog)}$ is the momentum tensor containing just the wheels modelled as masspoints and $I_{wo_i}$ the inertia tensor of the wheel where the column in direction of the rotation axis is set to 0.

$$I_{tot}^{(cog)} = I_o^{(cog)} + \sum_i I_{wo_i} + I_{wheels}^{(cog)} \tag{2.9}$$

Then the total angular momentum $L_{tot}$ can be written as Equation (2.10). Where $J_{w_i}$ is the Element from $I_{w_i}$ in the direction of the motor's rotation axis. $\omega_{o_i}$ is the element of $\omega_o$ in the respective rotation axis.

$$L_{tot}^{(cog)} = I_{tot}^{(cog)}\omega + \begin{pmatrix} J_{w_1}(\omega_{w,1} + \omega_{o_1}) \\ J_{w_2}(\omega_{w,2} + \omega_{o_2}) \\ J_{w_2}(\omega_{w,3} + \omega_{o_3}) \end{pmatrix} \tag{2.10}$$

Equation (2.10) can then be inserted into Equation (2.5). Where $\vec{u}$ is the torque input into the motors. This results into Equation (2.11) [6].

$$I_{tot}^{(cog)}\dot{\omega_o} = \vec{\omega_o} \times (I_{tot}^{(cog)}\vec{\omega_o}) - \vec{\omega_o} \begin{pmatrix} J_{w_1}(\omega_{w,1} + \omega_{o_1}) \\ J_{w_2}(\omega_{w,2} + \omega_{o_2}) \\ J_{w_2}(\omega_{w,3} + \omega_{o_3}) \end{pmatrix} - \vec{u} \tag{2.11}$$

## 2.4 Choice of Design

There are three main requirements on the system. It has to be small, cheap and fast.

To get a system that can conquer every possible angular momentum in a certain range and to avoid slow and very complicated control algorithms, there are at least two CMGs needed. Building a CMG is quite difficult, so prebuilt CMGs would be needed. A quick internet research showed that CMGs are rather expensive. Small educational CMGs can be bought for roughly 2000$ [2]. Although they are great in theory, CMGs are not suitable for this project.

Reaction wheels are quite cheap. The wheels can be built quite easily, small electric motors can be found in every model shop. Therefore, controlling the system with reaction wheels seems to be the better alternative.

To control a three-dimensional torque, a three-dimensional torque has to be produced. As shown by [7], one thruster would suffice if one can use external torques. But using reaction wheels, [6] prooves that there are at least three wheels needed.
A three-wheel configuration seems perfect. Every possible torque within a certain range can be produced by accelerating the wheels accordingly. Adding more wheels would mean higher cost, more mass to slow down and more energy to be provided. So in this paper, the possibilities of a system with three reaction wheels orthogonally arranged will be analyzed.

[2] http://www.gyroscope.com/d.asp?product=CMG

A configuration with only two CMGs might be the smallest, but they are far too expensive. A three-wheel configuration is able to slow down angular speed very fast and seems to be one of the cheapest solutions. Performance and size seem to be in a perfect ratio.

Chapter 3

# Hardware

## 3.1 Material and Limits

### 3.1.1 Motors

The choice of the motors is based on the following calculations and assumptions.

- The system can be modeled as a box with homogenous density with measures $a = 10cm \times b = 10cm \times c = 10cm$ and mass $m_{ob} = 0.85kg$.
- The flywheel can be modeled as a disc with homogenous density with radius $r_{wh} = 50mm$ and mass $m_{wh} = 60g$.
- After the system is thrown in the air, it has a frequency of 3Hz in one axis.
- The motors have a constant torque over the frequency.
- The system should stop in 0.7sec.

$$J_{ob} = \frac{m_{ob}}{12}\left(a^2 + b^2\right)$$

$$J_{wh} = m_{wh} r_{wh}^2$$

$$f_{wh} = \frac{J_{ob} f_{ob}}{J_{wh}}$$

$$T = \frac{m_{wh} r^2 2\pi f_{Wh}}{t}$$

| | |
|---|---|
| $J_{ob}$ | Moment of inertia of the whole object |
| $J_{wh}$ | Moment of inertia of one flywheel |
| $f_{ob}$ | Frequency of the whole object |
| $f_{wh}$ | Frequency of one flywheel |
| $T$ | Torque |

This results in a motor with a torque of 38mNm at a rotation speed of 1700 rpm. Due to that and the low price for it's specifications the Como Drills 719RE380 which has 37.5 nNm @ 19000 rpm is an appropriate choice.

### 3.1.2 Energy

The angular momentum caused by the person who throws the device, has to be absorbed by the flywheels very fast. So the requirements to the energy source are a high power and energy density. Thus a LiPo Akku 7,4 V / 1000 mAh / 10C BEC which is also an approved solution for other flying objects suits the requirements very well.

### 3.1.3 Control device

Possible solutions are a FPGA, a micro controller without a board or a single-board micro-controller. In order to build the prototype as quickly as possible and to have the opportunity to use existing libraries a single-board micro controller is the best option. Our favorite candidates for boards were "Arduino Mega Atmel Atmega2560 MCU Board" and "Arduino 32Bit ARM Cortex-Plattform". Because of the superior 32bit technology, higher clock speed and bigger memory in order to log the flight, we chose the "Arduino 32Bit ARM Cortex-Plattform".

### 3.1.4 Measurement

#### Angular Velocity

In the calculation for the Motor (Section 3.1.1) it was assumed, that the maximum angular velocity is 3Hz which is $1080\frac{grad}{sec}$. Moreover the measurement is needed in all three axes.

#### Acceleration

To determine weather the system is in free fall or not, a accelerometer is needed. If the sensor is mounted in the center of mass of the box, then the acceleration drops to zero if the system is thrown in the air.

#### Conclusion

Due to that the “Gyro Breakout - MPU-6050 SEN-11028” was chosen, which can measure an angular velocity up to $2000\frac{grad}{sec}$ in all three axes and the acceleration in all three axes.

## 3.2 Realization

### 3.2.1 Whole Box Assembly

The following demands on the prototype were identified.

- Compactness, in order to be ergonomically.
- Robustness, in order to stand a high number of tests.
- Stiffness, in order to get less error on the gyroscope and accelerometer.
- An inertia tensor with small non diagonal values in a coordinate system of the rotation axes of the motors.
- Lightweight, in order to stop the system fast.

The tradeoff was between robustness, stiffness and good inertia tensor versus compactness and weight.

#### Housing

For the housing a customary plastic box meets the demands, because of its good robustness and stiffness compared with its weight. All components, as well as the motor assembly shown in the section Motor Assembly 3.2.1, are mounted with laces or screws to the plastic box.

### Bought-in Components

| Part | Name | Quantity |
|---|---|---|
| Motor | Como Drills 719RE380 | 3 |
| Motor Controller | Motor Driver 15A IRF7862PBF ROB-09107 | 3 |
| Battery | Conrad energy LiPo ET Flugakku 7.4 V / 1000 mAh (10 C) Stecksystem BEC / XH | 1 |
| Micro controller board | Arduino 32Bit ARM Cortex-Plattform | 1 |
| Level Converter | Logic Level Converter BOB-08745 | 3 |
| Gyro | Triple Axis Accelerometer & Gyro Breakout - MPU-6050 SEN-11028 | 1 |

### Motor Assembly

The motors are mounted with laces to an aluminum sheet, which is a very light and small construction. In order to keep the none diagonal values of the system small, the remaining parts, such as the micro controller, are mounted in proper position. Fig 3.1 shows how the motors including the flywheels are placed on the aluminum sheet. The drawings for the aluminum sheet and the flywheels can be found in the appendix.

### The First Prototype

In the first prototype the motor assembly was mounted directly with screws to the ground of plastic box. The benefit was the light weight. Unfortunately the gyroscope and accelerometer were irritated when one ore more motors accelerated.

### The Final Prototype

In order to reduce the measurement errors caused by the motors the use of a stiffer box and an aluminum sheet to reenforce the ground of the plastic box provided a remedy.

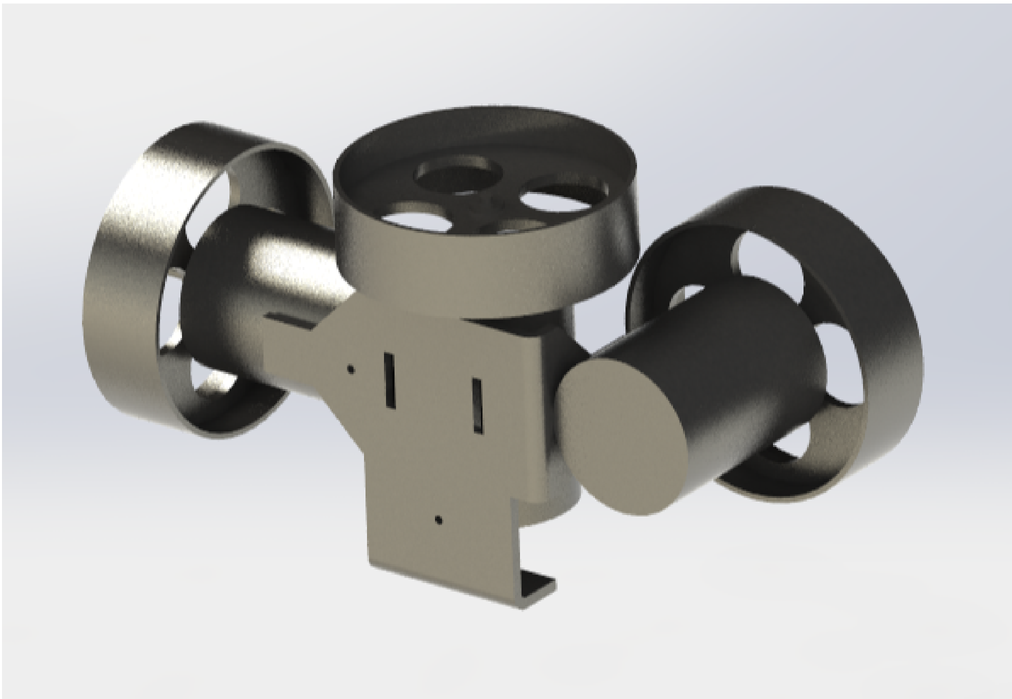


Figure 3.1: Motor Assembly

Chapter 4

# Solving the Control Problem

The controlled system contains the controller, the motors and the physical object that is controlled.

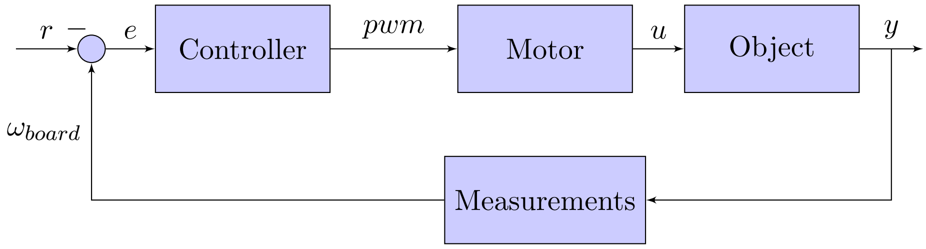



Figure 4.1: The System that needs to be controlled

The known variables of the system are the current angular velocity $\omega_{board} \in \mathbb{R}^3$ of the object and the PWM ratio of the input. In the simulations the torque is the input $u \in \mathbb{R}^3$.

## 4.1 One Dimension

In one dimension the system can only rotate on one axis (e.g. the unit vector in z-direction $\vec{e}_z$) parallel to the axes of the control momentums. This reduces the Equation (2.11) to Equation (4.1). Where $J_{tot_z}$ is the inertia torque in z-Axis and $\omega_{o_z}$ is the rotation speed around the fixed axis. $u_z$ is the torque input in the motor.

$$J_{tot_z}\dot{\omega}_{o_z} = -u_z \tag{4.1}$$

### 4.1.1 P-Controller

The change of the Angular Momentum (Equation (2.3)) is directly related to the torque. In one dimension a P-controller can control this system by changing the applied torque $u$ on the object.

$$u(t) = K_p \cdot e(t) \tag{4.2}$$

**Simulation**

The system was modelled in the Open Dynamics Engine [1]. A delay between the reaction of the motor of 80ms was added and the Controller was reevaluated every 20ms. A P-Controller steered the system to $w_z = 0$ (See Figure 4.2).

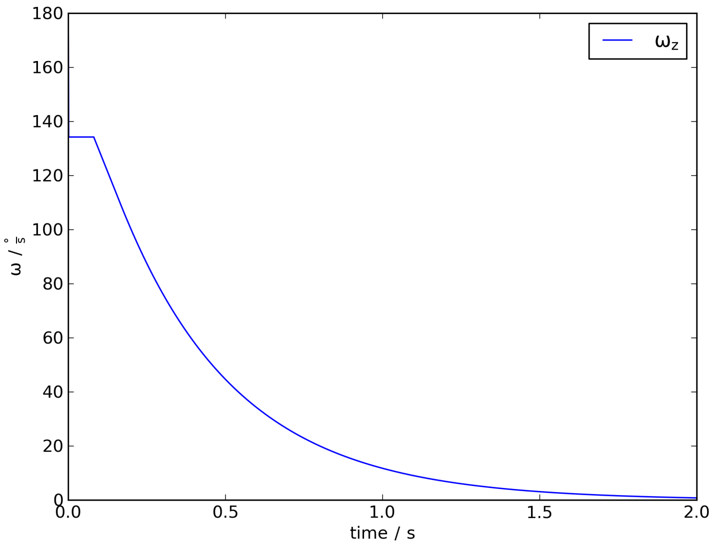



Figure 4.2: Simulating the system in one dimension.

### 4.1.2 PI-Controller

In the simulation we were able to control the torque directly, this is not possible on the hardware. The motors are controlled by changing the voltage. Without the ability to control the torque on the motor, a P Controller does not work effectively.

A DC-motor's torque can be expressed as Equation (4.3) [8, Equation (5.28)], where $\Phi$ is the motors flux and $I$ the current flowing into the motor.

$$M \propto \Phi I \tag{4.3}$$

[1] http://www.ode.org/

Combining basic DC motor characteristics [8, Equation (5.26-28)], the torque can be written as Equation (4.4).

$$M = \frac{k\Phi}{2\pi R_a} \cdot (U - k\Phi n) \tag{4.4}$$

It is clear that when the rotation frequence $n$ of the motor equals $\frac{U}{k\Phi}$ no more torque is generated. To compensate for this effect an integral part is added.

The PI Controller is defined by Equation (4.5)

$$u(t) = K_p \cdot e(t) + \int_0^t K_i \cdot e(\tilde{t})\, d\tilde{t} \tag{4.5}$$

The proportional factor $K_p$ and integral factor $K_i$ can be determined by experiments, a common way in the industry to configure a PI-Controller.

### 4.1.3 Results

The PI-Controller in Section 4.1.2 was implemented in the software and tested. The sytem turns on the z axis and was started with a certain velocity, then the control algorithm starts and stops the turning box.

Figure 4.3 shows the angular velocity $\omega_z$ and the PWM ratio of the output. It stabilizes using this PI controller.

## 4.2 Three Dimensions

### 4.2.1 Max-Controller

To stop the system as fast as possible, a simple Max-Controller could work. As soon as the gyro meassures an angular velocity, the motors react with a torque as big as possible to compensate and stop the system. Simulating such an algorithm provides very nice results for small angular velocities. Figure 4.4 shows the output of a simulation using ODE with a Max-Controller that updates the control torques every 20ms.

In reality, this algorithm does not work at all.
Possible explanations include a quite long reaction time of the motors. The simulated motors deliver a requested torque in the next simulation steps. Real motors need some reaction time. Simulating a motor reaction time of 80ms shows a swinging system. (Figure 4.5) Here, the control torques were delayed by 80ms before applying them on the motors.

Also, switching the power of the motors that fast and that strong overheats the system. So it is not a practical solution.

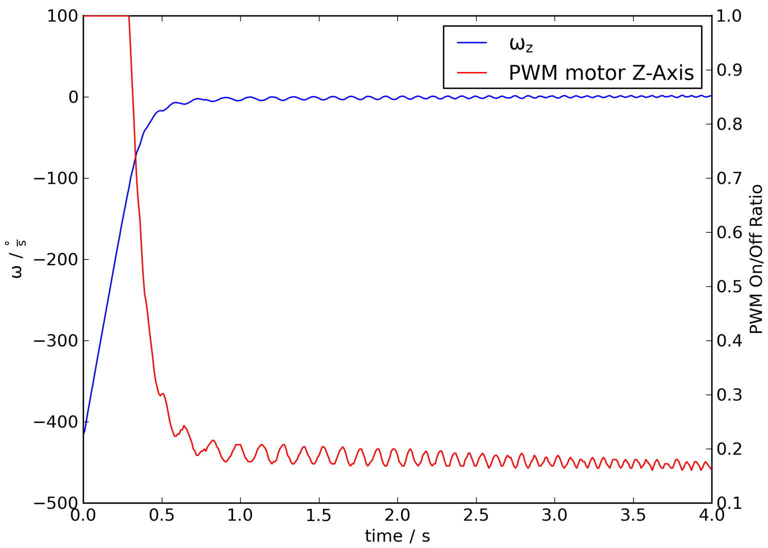



Figure 4.3: Measuring the rotation on the z-axis

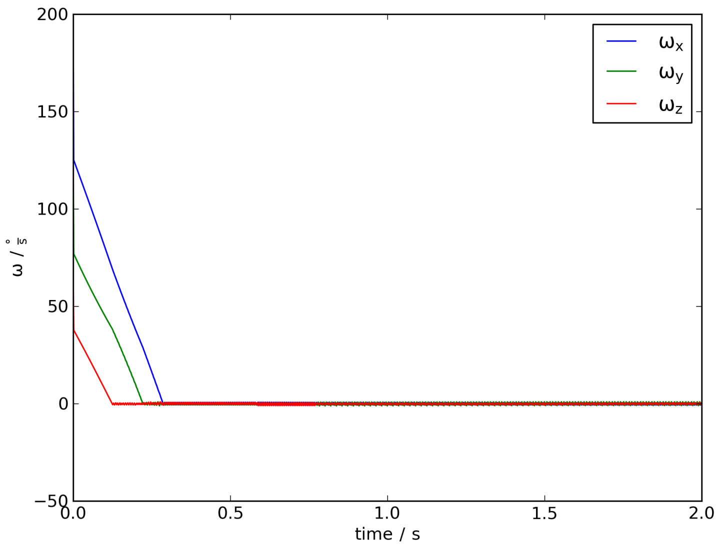



Figure 4.4: Simulation using a simple Max-Algorithm

### 4.2.2 PI-Controller

To reduce the effect of the motor delay, the motor torques can be scaled with the absolute value of the angular velocity. Like this, as soon as they get close to

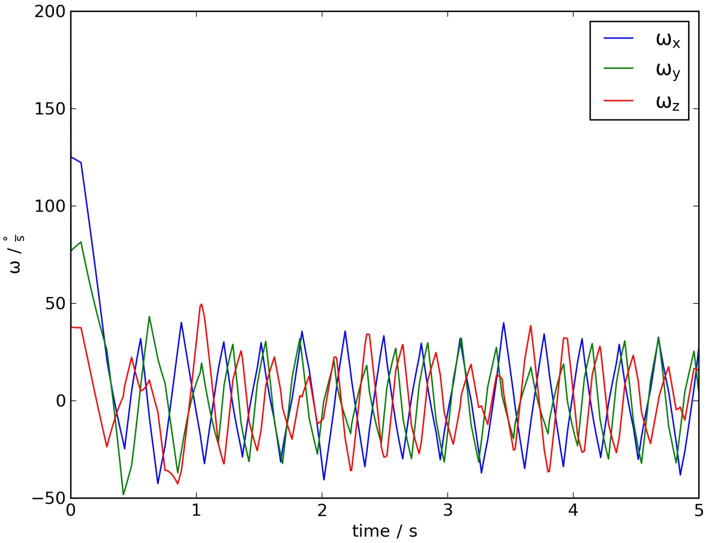



Figure 4.5: Simulation using a delayed Max-Algorithm

zero, less torque will be produced. So acceleration into the wrong direction will be minimized.
In one dimension, controlling the torque with a P-controller works great. Simulations including the reaction time of the motors show quite promising results.

Figure 4.6 shows a simulation with a P-controlled torque. The three dimensional omega converges to zero. Again, the motors have a 80ms delay and the controller updates the torques every 20ms.

This time, the system does not overheat. The changes are small enough so that it can handle them. Reality and simulation show the same results for small omegas. Figure 4.7 and Figure 4.8 show the behaviour of our system in the real world. As described in Section 4.1.2, we controlled the motors with PWM and a PI-controller, because we can not control the torque directly.

### 4.2.3 Further Ideas

Simulations with the PI-Algorithm, bigger angular velocities and same maximal torque show diverging behaviour. Figure 4.9 shows the simulated reaction of the system for a quite huge angular speed. Motors have a 80ms delay, the control torques are updated using a P controller every 20ms.

A possible explanation can be found looking at Equation (2.11). Using control theory[6] and assuming that all wheels have the same inertia J, we get the control

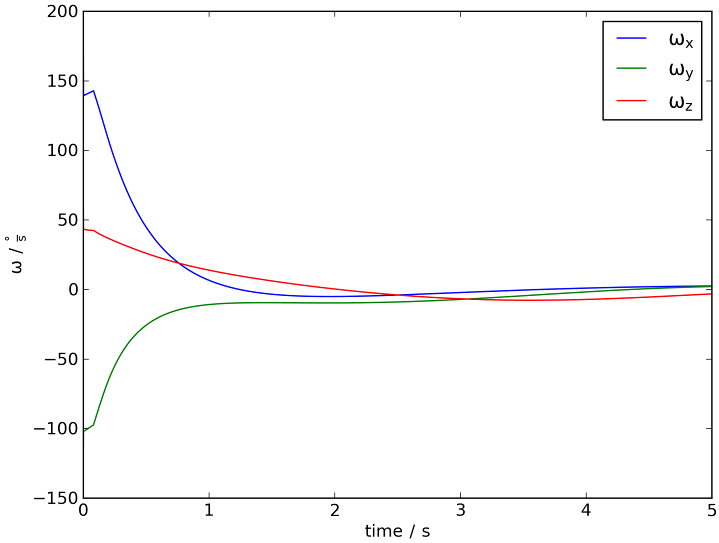



Figure 4.6: Simulated system, P-controlled torque

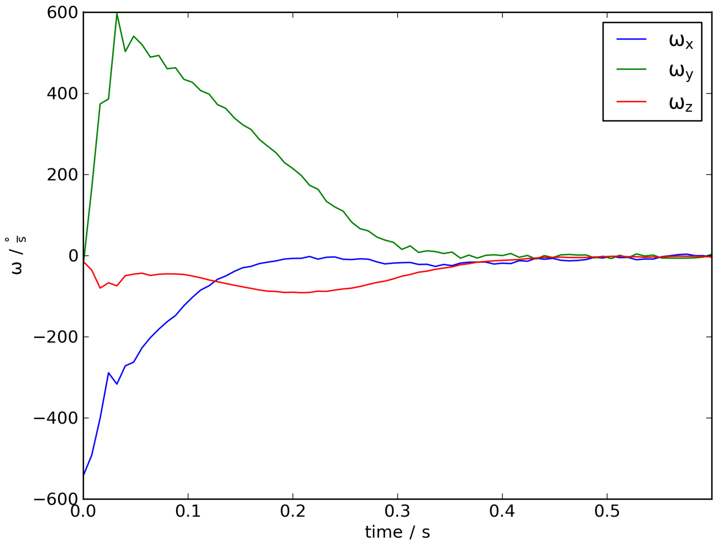



Figure 4.7: Real system, PI-controlled PWM, simple omega

algorithm

$$\vec{u} = P\vec{\omega}_o + \vec{\omega}_o \times (I\vec{\omega}_o + J(\vec{\omega}_o + \vec{\omega}_w)) \tag{4.6}$$

Using this, a three reaction wheel system can be stabilized. [6]

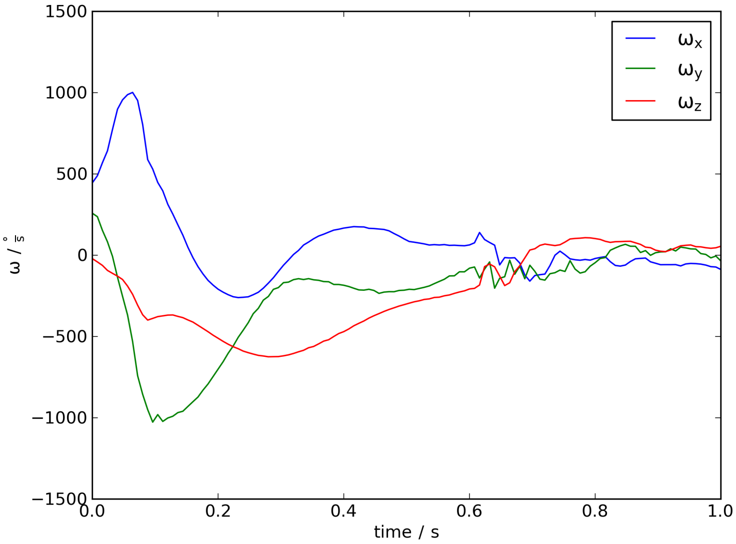



Figure 4.8: Real system, PI-controlled PWM, crazier omega

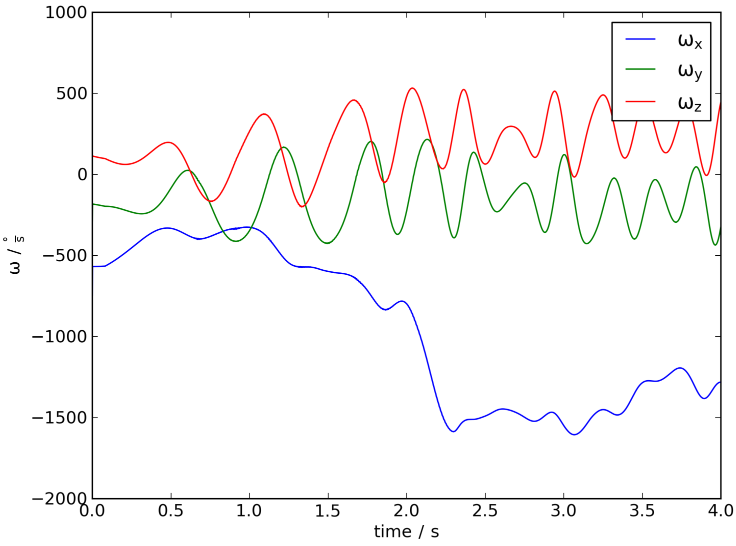



Figure 4.9: P-Controller: Diverging behaviour for huge starting values

Chapter 5

# Discussion and Conclusion

### 3D

Equation (4.6) can be split in two parts: $P\vec{\omega}_o$ accounts for the current speed of the device. Its absolute value depends linearly on $\vec{\omega}_o$.
$\vec{\omega}_o \times (I\vec{\omega}_o + J(\vec{\omega}_o + \vec{\omega}_w))$ represents the torques produced by changing the position of the coordinate system (see Equation (2.4)) and the rotating wheels. Its absolute value depends quadratically on $\vec{\omega}_o$. So we can neglect this part if $\vec{\omega}_w$ and $\vec{\omega}_o$ are small enough. The remaining equations represents a simple P-controller. Hence the starting $\vec{\omega}_o$ is assumed to be small, $\vec{\omega}_w$ will be small enough and the second part can be neglected.

For big $\vec{\omega}_o$, the second summand gets important. The resulting controller differs from a P-controller. So a P controller only works up to a certain $\vec{\omega}_{crit}$.

For bigger omegas, the basic layout of our system is still sufficient. But to implement the controller provided in Equation (4.6), one would need detailed information about the applied torques and the current motor speeds. This could be realized by measuring the motor currents using Hall-sensors. (see Equation (4.3)) This would lead to higher cost and much more complicated algorithms. So it was not implemented.

Experiments with our box showed that $\vec{\omega}_{crit}$ is not reached when thrown without excessive speed. The system stops with a P-controlled torque a PI-controlled PWM. (see Section 4.1.2 and Section 4.2.2) If someone uses a stabilized camera, he or she tries to make it spin as slow as possible. So ignoring high angular speeds is a valid assumption.

### 1D

Restricting two dimensions, e.g. by putting it on a flat surface, Equation (4.6) gets much simpler. Modelling the restraints of a x-y-plane, $\vec{\omega}_o$ can be written as

$$\vec{\omega}_o = \omega_o \vec{e}_z \tag{5.1}$$

Hence rotation in x and y direction is constrained, only the z component of $\vec{u}$ is of interest. Substituting Equation (5.1) in Equation (4.6) and looking at the component in z-direction leads to a P-controlled torque:

$$\begin{aligned}\vec{u} \cdot \vec{e}_z &= (P\omega_o \cdot \vec{e}_z + \omega_o \cdot \vec{e}_z \times (I\vec{\omega}_o + J(\vec{\omega}_o + \vec{\omega}_w))) \cdot \vec{e}_z \\ &= P\omega_o \cdot \vec{e}_z \cdot \vec{e}_z + (\omega_o \cdot \vec{e}_z \times (I\vec{\omega}_o + J(\vec{\omega}_o + \vec{\omega}_w))) \cdot \vec{e}_z \\ &= P\omega_o\end{aligned} \tag{5.2}$$

So the 1D P/PI-controller is just a special case of Equation (4.6).

### Project Goals

A system idea was developed and a prototype build. Different controller ideas were checked for feasibility and first results simulated. A simple PI Controller resulted in a very stable and good result. The goal to build a device that stops its own rotation was achieved.

# Bibliography

[1] Balke, H.: Einführung in die Technische Mechanik - Kinetik, Auflage 3. Springer (2011)

[2] Gerlach, O.: Attitude stabilization and control of earth satellites. Space Science Reviews, Volume 4, Issue 4, pp.541-582 (1965)

[3] Heiberg, C.J., Bailey, D., Wie, B.: Precision spacecraft pointing using single-gimbal control moment gyroscopes with disturbance. Journal of Guidance, Control and Dynamics (2000)

[4] Cornick, D.E.: Singularity avoidance control laws for single gimbal control moment gyros. In Irvine, R.B., Ritter, J.W., eds.: Guidance and Control Conference. (February 1979) 20–33

[5] Lappas, V., Steyn, W., Underwood, C.: Attitude control for small satellites using control moment gyros. Acta Astronautica (2002)

[6] Blenden, R.: Regenerative power optimal reaction wheel attitude control. Master's thesis, University of Colorado (2010)

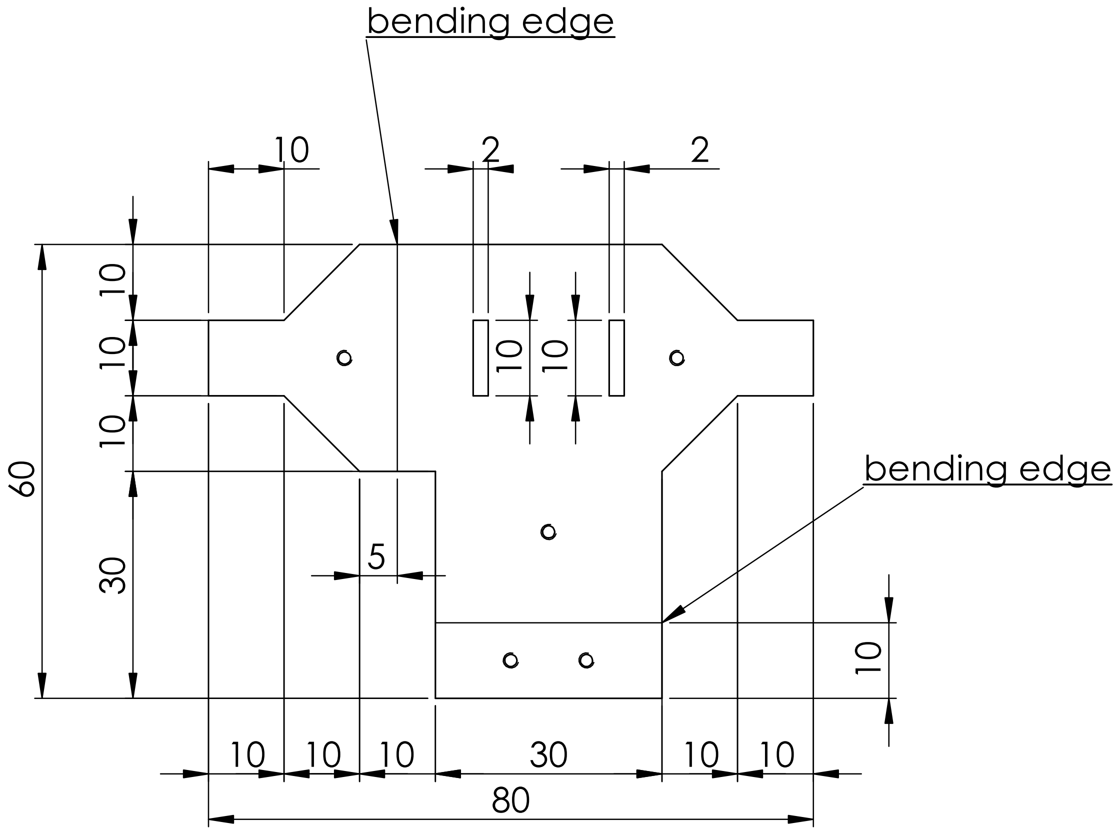
[7] Aeyels, D.: Stabilization of a class of nonlinear systems by a smooth feedback control. Systems & Control Letters **5**(5) (1985) 289 – 294

[8] Hameyer, K.: Electrical machines i: Basics, design, function, operation. Lecture Notes (2001)

[9] Harms, H., Amft, O., Winkler, R., Schumm, J., Kusserow, M., Tröster, G.: Ethos: Miniature orientation sensor for wearable human motion analysis. Proceedings of IEEE Sensors conference, IEEE (2010) We used the Sensor to understand flight characteristics.

Appendix A

# Drawings




all bores M2

| | | | | |
|---|---|---|---|---|
| UNLESS OTHERWISE SPECIFIED:<br>DIMENSIONS ARE IN MILLIMETERS<br>SURFACE FINISH:<br>TOLERANCES:<br>LINEAR:<br>ANGULAR: | FINISH: | DEBUR AND BREAK SHARP EDGES | DO NOT SCALE DRAWING | REVISION |

| | NAME | SIGNATURE | DATE |
|---|---|---|---|
| DRAWN | Andreas Steger | | |
| CHK'D | | | |
| APPV'D | | | |
| MFG | | | |
| Q.A | | | |

TITLE:

# Motorhalterung

MATERIAL: Alublech 2 mm

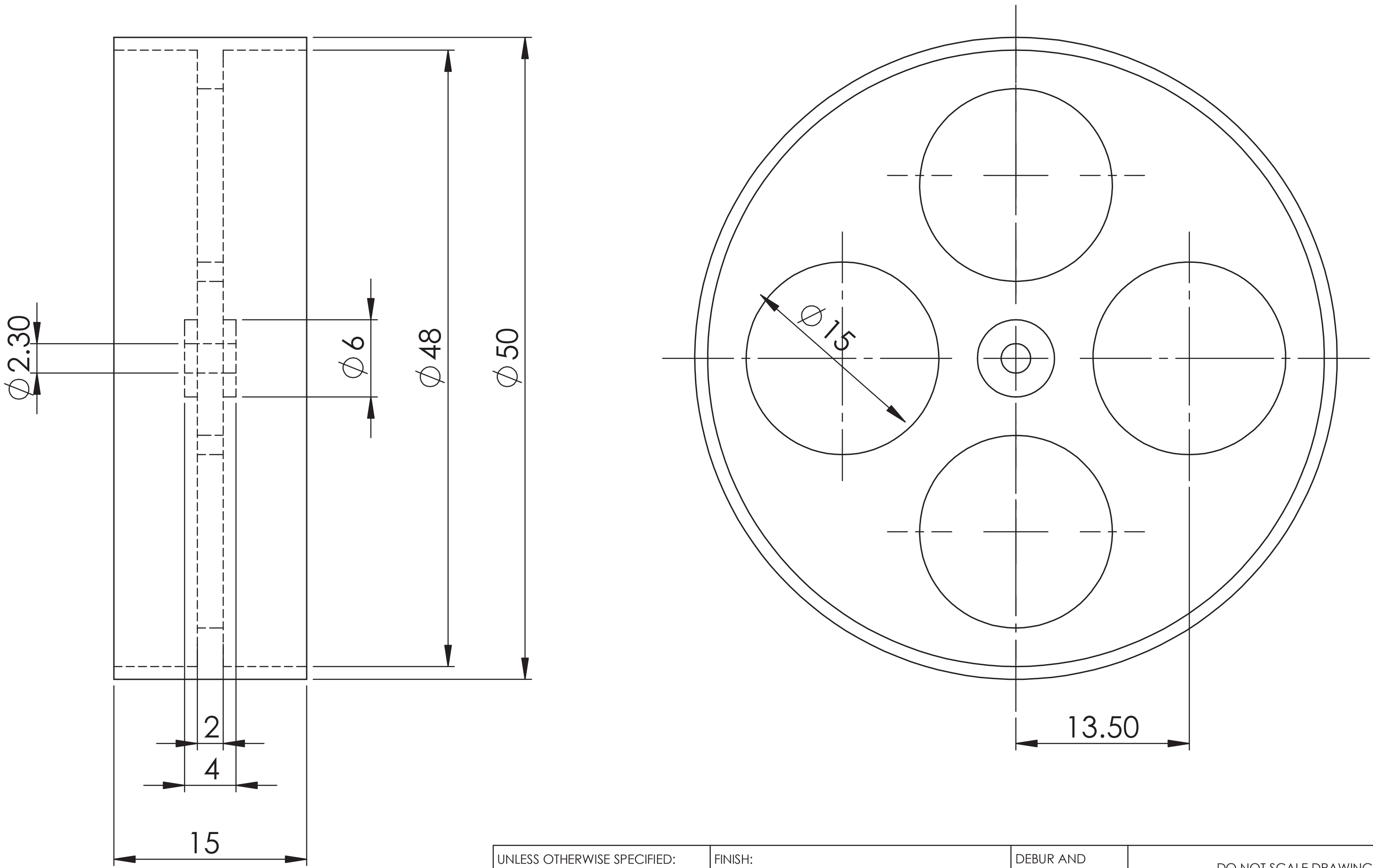
DWG NO. Motorhalterung

A4

WEIGHT:

SCALE:1:1

SHEET 1 OF 1




UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MILLIMETERS
SURFACE FINISH:
TOLERANCES:
LINEAR:
ANGULAR:

FINISH:

DEBUR AND BREAK SHARP EDGES

DO NOT SCALE DRAWING

REVISION

| | NAME | SIGNATURE | DATE |
|---|---|---|---|
| DRAWN | Andreas Steger | | |
| CHK'D | | | |
| APPV'D | | | |
| MFG | | | |
| Q.A | | | |

MATERIAL: Automatenstahl

WEIGHT:

TITLE: Drallscheibe

DWG NO. Drallscheibe

A4

SCALE:1:1

SHEET 1 OF 1