UNIVERSITEIT • STELLENBOSCH • UNIVERSITY

The Feedback Control Of A Robotic Gymnast

by

Henry Kotzé

Mechatronic Project 448

Report submitted in partial fulfilment of the requirements of the module Mechatronic Project 448 for the degree Baccalaureus in Engineering in the Department of Mechanical and Mechatronic Engineering at the University of Stellenbosch

Study leader: Dr. J.A.A Engelbrecht

October 2018

Declaration

By submitting this thesis electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: 2018/10/27 .....

Abstract

The Feedback Control of a Robotic Gymnast

H. Kotzé

Department of Mechanical and Mechatronic Engineering,
University of Stellenbosch,
Private Bag X1, Matieland 7602, South Africa.

Thesis: BEng (Mechatronics)

October 2018

In this report a design method for the swinging and balancing of the under-actuated robotic gymnast was researched, simulated and tested on a physical model. The electronic, mechanical and software designs are discussed to show how the physical model was constructed, controllers implemented and data acquired respectively.

Uittreksel

Die Terugvoer Beheerwet van 'n Robotiese Gimnas

("The Feedback Control of a Robotic Gymnast")

H. Kotzé

Departement Meganiese en Megatroniese Ingenieurswese,
Universiteit van Stellenbosch,
Privaatsak X1, Matieland 7602, Suid Afrika.

Tesis: BIng (Megatronika)

Oktober 2018

In die projek word die swaaiende en balanseering beheerwette vir 'n robotiese gimnas genavors, ontwerp en getoets op 'n fisiese model. Die eletroniese, meganies en sagteware ontwerpte wat die resultate in the report toegelaat het word bespreek.

Acknowledgements

I would like to express my sincere gratitude to the following people and organisations.

Dr. Japie Engelbrecht for supervising the project throughout the year. He provided critical feedback on progress, guided me in the correct direction and gave me confidence to complete the project.

The Electrical and Electronic Department for allowing the use of their equipment and facilities. They allowed me to work with confidence ensuring the facilities and equipment are maintained with no interruption to my work.

Mr. Croukamp, Mr. van Eenden and Mr. Gift for assisting me with the mechanical and electronic designs. The atmosphere in the manufacturing labotorium was always welcoming.

The group of final year student working together in the 4th floor labs. You all made the experience more worthwhile and encouraged me through difficult times.

Dedications

Hierdie verslag word opgedra aan my ouers wat my ondersteun het tydens die 4 jaar om die graad van ingenieurswese te ontvang en die Here vir sy genade.

Contents

List of Figures

List of Tables

Nomenclature

Constants

g =	$9.81\,\mathrm{m/s^2}$

Variables

I	Inertia	$[\,\mathrm{kg{\cdot}m^2}\,]$
m	mass	$[\,\mathrm{kg}\,]$
l	Lenght	$[\,\mathrm{m}\,]$
L	Lenght	$[\,\mathrm{m}\,]$
R	Reaction Force	$[\,\mathrm{N}\,]$
x	Coordinate	$[\,\mathrm{m}\,]$
$\ddot{x}$	Acceleration	$[\,\mathrm{m/s^2}\,]$
θ	Rotation angle	$[\,\mathrm{rad}\,]$
ϕ	Rotation angle	$[\,\mathrm{rad}\,]$
τ	Torque	$[\,\mathrm{N{\cdot}m}\,]$

Vectors and Tensors

$\overrightarrow{\boldsymbol{q}}$	Physical vector, see equation ()...

Subscripts

a	Adiabatic
a	Coordinate

Chapter 1

Introduction

1.1 Problem Statement

A feedback control system for a robotic gymnast that is able to swing from the "hanging" position to the "handstand" position must be designed, implemented and verified. Feedback control loops must be designed that use the "legs" of the gymnast to swing the "body" of the gymnast from the "hanging" position to a "handstand" position and then balance the gymnast on top of the horizontal bar. A mathematical model for the dynamics of the swinging robotic gymnast system must be derived or sourced from literature. The dynamics are analysed to propose an appropriate feedback control architecture that actuates the "legs" of the gymnast using feedback from sensors that measure the swinging motion of the gymnast on a horizontal bar. A practical demonstrator must be constructed and the correct operation must be demonstrated.

1.2 Literature Study

Previous research attempts to the swinging and balancing of the robotic gymnast was to use 2 controllers. The first controller was responsible for swinging the robotic gymnast from the hanging position up towards the inverted position. When the swing-up controller has brought the gymnast near the inverted position, a new controller was used to bring the gymnast to balancing. The different approaches used in these 2 controllers by the various researchers are summarised below and concludes on the approach used in the report.

Spong (1995) implements the swing-up controller by using partial feedback linearisation which results in a linear response from either the actuated or unactuated pendulum. Using non-collocated linearisation he is able to allow the unactuated pendulum to follow a desired trajectory and shows this using

the phase portrait of the zero dynamics of the system the system will converge to the unstable equilibrium position. When the swing-up controller brings the system to the inverted balancing position the controllers will switch using a linear quadratic regulator (LQR) to balance the system.

Spong (1995) continues further how the swing-up can be achieved using collocated linearisation which linearises the response of the actuated pendulum. This allows the actuated pendulum to follow a desired trajectory and Spong (1995) provides a energy based trajectory that increases the energy in the system. The increase of energy in the system results in the pendulums to rise to the balancing position where a LQR controller is used to balance the system. Both of these 2 implementation were succesfull in swinging and balancing the robotic gymnast in simulation.

Brown and Passino (1997) provided two manually tuned nonlinear controllers for the balancing of the robotic gymnast that was tested against a designed LQR controller. These two are the direct fuzzy controller (DFC) and a fuzzy model reference learning controller (FMRLC). The gains selected for the DFC controller were based of the LQR controller implemented and successfully balanced the gymnast. However the LQR controller provided smoother state trajectories and lower control input. The FMRLC uses no explicit dynamical reference model and instead the outputs of the plant using normalising gains are directly fed into the second fuzzy system. The FMRLC controller was a significant improvement over the DFC, but yet again did not perform as well as the LQR controller.

Brown and Passino (1997) attempts to swing the acrobat to the inverted balancing position using the energy based trajectory proposed by Spong (1995) without the partial linearisation feedback. Brown and Passino (1997) was able to swing and balance the acrobat using this approach, but required greater input and were not as smooth as using with partial feedback linearisation.

The mathematical derivation of the robotic gymnast was mostly done using the Euler-Lagrangian equation. This approach was used due to the mechanical energy in the system are easily identified. Derivation of the robotic gymnast are done using simple approximation as point masses shown in (Lehl, 2012) and (Tedrake, 2009).

Based on the literature study the approached decided to used was the feedback linearisation with the energy based trajectory. The dynamics of the robotic gymnast will be derived without the approximation mentioned above. The linear controller to balance the robotic gymnast will be using the full state feedback.

1.3 System Overview

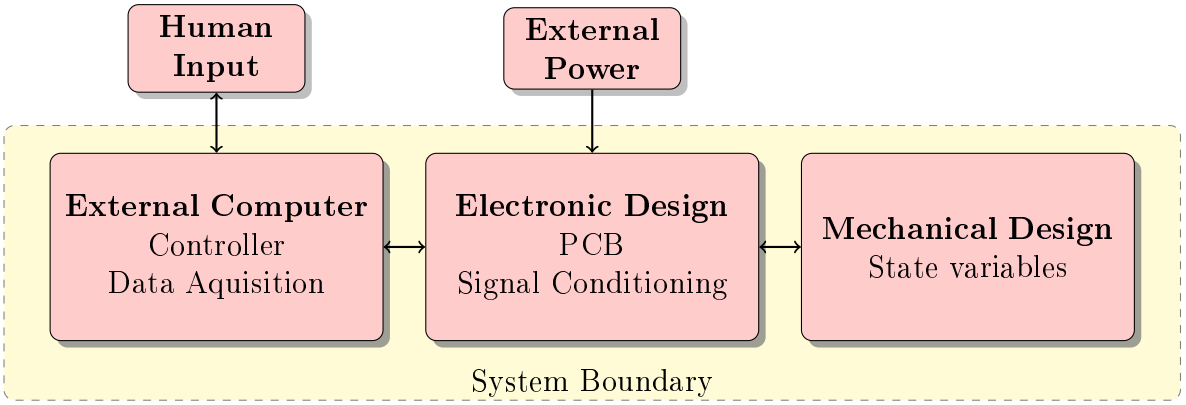


Figure 1.1: System Overview of the Feedback Control of Robotic Gymnast

Figure 1.1 provides an overview of the various subsystems that the project contains with the external factors that influence it. The project was subdivided into these subsystems because they can be developed separately with little interaction between each other. An brief overview on each subsystem is presented here.

The external computer will contain the theoretical controller that determines the input to the mechanical components and receives information from the electronic design about the state of the system. This allows for the verification of system parameters, debugging and experimental tests.

The electronic design acts as the middle-man between the mechanical design and external computer. It provides instructions to the mechanical design based on commands received from the external computer while sending data to the external computer. The electronic design contains the Printed Circuit Board (PCB) that conditions all the signals for processing.

The mechanical design was responsible for creating a physical model that represents the mathematical model describing the system. The correct sensors was selected to measure the state variables while providing interfaces for the electronic design.

External inputs to the system were the power that was supplied to the system and inputs by the user to test the various subsystem.

1.4 Project Execution

The execution of the project occurred in a sequence of steps to achieve the results presented in the report. It is presented to provide the reader an understanding of how the project was developed in separate subsystems and combined in the end.

First the mathematical model of the system was derived by using the appropriate approach. The derived mathematical model was then implemented on a simulation program where the dynamics of the system was verified and inspected.

Following the successful implementation of the mathematical model the various controllers were designed and implemented on the simulation program. The behaviour of the simulated responses were inspected and analysed.

The simulation provided the specification for the mechanical design to commence and created the physical model that provided an acceptable representation of the mathematical model.

During manufacturing of the mechanical design the electronic design started. Conceptual designs were created capable of meeting the requirements and the selected design was manufactured. The electronic design was then tested to ensure it performs as designed with the opportunity to create revisions.

Following the successful testing of the electronic design, the programming of the microcontroller and external computer started. This included the programming of the controller, data acquisition system and the conversions of the sampled data.

Once the microcontroller could provide the external computer with system state information the system identification tests occurred to determine the various system parameters. These new system parameters were used in the simulation program to update the existing controllers and verify the responses in simulation.

The updated controllers were then implemented onto the external computer for the system experiments to start. From these system experiments the response of the experiments were compared to those of the simulation.

The report was written throughout the sequence of steps mentioned above and was completed and reviewed at the end.

1.5 Report Outline

A brief overview of each chapter in the report is provided here. It acts as a primer for the reader and the identification of sections that may interest the reader more.

Chapter 2 explains the system concepts that is reference throughout the report. It contains the mathematical derivation of the robotic gymnast and the simulated model. The system parameters with system identification tests are shown and shows the simulated model is a acceptable representation of the physical model.

Chapter 3 describes the controller architecture to the swinging and balancing of the robotic gymnast. The specification for the controllers and the simulated responses of the controller are provided.

Chapter 4 contains the designs of the mechanical and electronic systems of the project. The various components used in the designs are discussed and explained.

Chapter 5 describes the software implemented to provide the report with these results. It explains the architecture of the software and the various functions implemented.

Chapter 6 provides the practical results of the controllers explained in chapter 3 and hypothesise unexpected behaviour in the experiments.

Chapter 7 concludes the report with a summary of the report and recommendation for future endeavours on the Feedback Control of a Robotic Gymnast.

Chapter 2

Conceptualisation and Modelling

2.1 System Concepts

The report contains many variable names and use of terminology for concepts that is used throughout the report. These variables and terminologies are defined here.

The double pendulum is a underactuated system which is defined as a system where the input to the system cannot command all of the state variables an instantaneous acceleration (Tedrake, 2009). This is due to the control input only actuating the lower pendulum and the energy in the lower pendulum must be transferred to the upper pendulum to initiate an acceleration.

The robotic gymnast is describe as a double pendulum consisting out of an actuated- and unactuated pendulum as seen in Figure 2.1. The position of the unactuated pendulum is described by the angle θ whereas the actuated pendulum is described by ϕ relative to θ. The angle's θ and ϕ are the independent parameters that describe the entire system.

There are 2 position of interest where the system contain special characteristics. These 2 positions are the stable- and unstable equilibrium position. In the stable position the system is at rest hanging downwards where both θ and $\phi = 0\,\text{rad}$. It is stable due to the system containing negative real poles resulting in the system returning to this position when disturbed. The unstable equilibrium position is where $\theta = 2\pi\,\text{rad}$ and $\phi = 0\,\text{rad}$ resulting in the robotic gymnast balancing in the inverted position. In this position the system contains positive real poles and any disturbance will cause the system to grow away from this position.

2.2 Mathematical Model

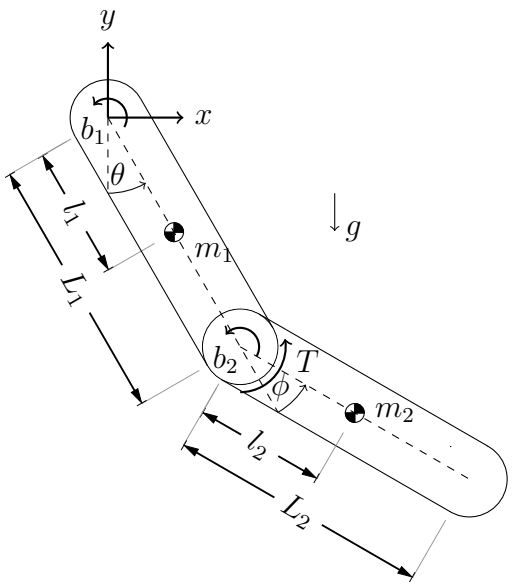


Figure 2.1: Free Body Diagram of the Double Pendulum

The approach taken to derive the mathematical model of the robotic gymnast is presented in this section. It is presented to allow the reader to understand parameters used throughout the report and critical to the implementation of the swing-up controller. The swinging of the robotic gymnast consist of non-linear behaviour and it is require to fully derive the dynamics of the system. The strenuous mathematical steps are removed from the reader and provided in Appendix A.1. A summary of the motivation and paradigm approach to the derivation is provided here.

Figure 2.1 shows the free body diagram of the robotic gymnast which was modelled as two pendulums connected together with a hinge. Each pendulum was modelled as having their mass distributed arbitrary along their axis. A torque is actuating the lower pendulum and friction was modelled as a function of the angular velocity.

The Euler-Lagrange equation shown in (2.1) was used to derive the dynamics of the system by analysing the energy of the system which is easily defined as the potential energy T and the kinetic energy V of the 2 pendulums.

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \vec{\dot{q}}} - \frac{\partial \mathcal{L}}{\partial \vec{q}} = 0 \tag{2.1}$$

$$\mathcal{L} = T - V \tag{2.2}$$

Using the Euler-Lagrange equation leads to the condense equations shown in (2.3) and (2.4),

$$d_{11}\ddot{\theta} + d_{12}\ddot{\phi} + h_1 + \psi_1 = 0 \tag{2.3}$$

$$d_{21}\ddot{\theta} + d_{22}\ddot{\phi} + h_2 + \psi_2 = \tau \tag{2.4}$$

where the coefficients are defined as

$$d_{11} = I_a + I_b + m_2(L_1^2 + l_2^2 + 2L_1 l_2 \cos(\phi)) \tag{2.5}$$

$$d_{12} = I_b + m_2(l_2^2 L_1 l_2 \cos(\phi)) \tag{2.6}$$

$$h_1 = -m_2 L_1 l_2 \sin(\phi)\dot{\phi}^2 - 2m_2 L_1 l_2 \sin(\phi)\dot{\phi}\dot{\theta} \tag{2.7}$$

$$\psi_1 = (m_2 l_1 + m_2 L_1) g \cos(\theta) + m_2 l_2 g \cos(\theta + \phi) + f_{c_1} \tag{2.8}$$

$$d_{21} = I_b + m_2(l_2^2 + L_1 l_2 \cos(\phi)) \tag{2.9}$$

$$d_{22} = I_b + m_2 l_2^2 \tag{2.10}$$

$$h_2 = m_2 L_1 l_2 \sin(\phi)\dot{\theta}^2 \tag{2.11}$$

$$\psi_2 = m_2 l_2 g \cos(\theta + \phi) + f_{c_2} \tag{2.12}$$

The friction that develops in the pendulums are for now represented by the f_{c_1} and f_{c_2} terms and will be expanded in the system identification section.

2.3 Simulation Model

The mathematical model derived in the previous section was required to be implemented on a simulation program to test the controllers used. Simulating the model allows the designer to understand how system parameters influence the dynamics of the system and the verification of controllers implemented. It will be presented by discussing the non-linearities added to represent the physical model better.

Simulation of the robotic gymnast was done using *MATLAB Simulink*. The differential equations shown in equation (2.3) and (2.4) were implemented using the *MATLAB Function* box. It was required to write $\ddot{\phi}$ and $\ddot{\theta}$ as the subject in each of the *MATLAB Function* box to allow MATLAB to simulate the model.

Non-linear behaviour introduced by sensors and components were added such as saturation of the motor torque, gearbox backlash and quantisation of sensory data. These non-linearities were implemented to allow the simulation to be an acceptable representation of the physical system. The Simulink Model used for simulation is shown in Figure 2.2.

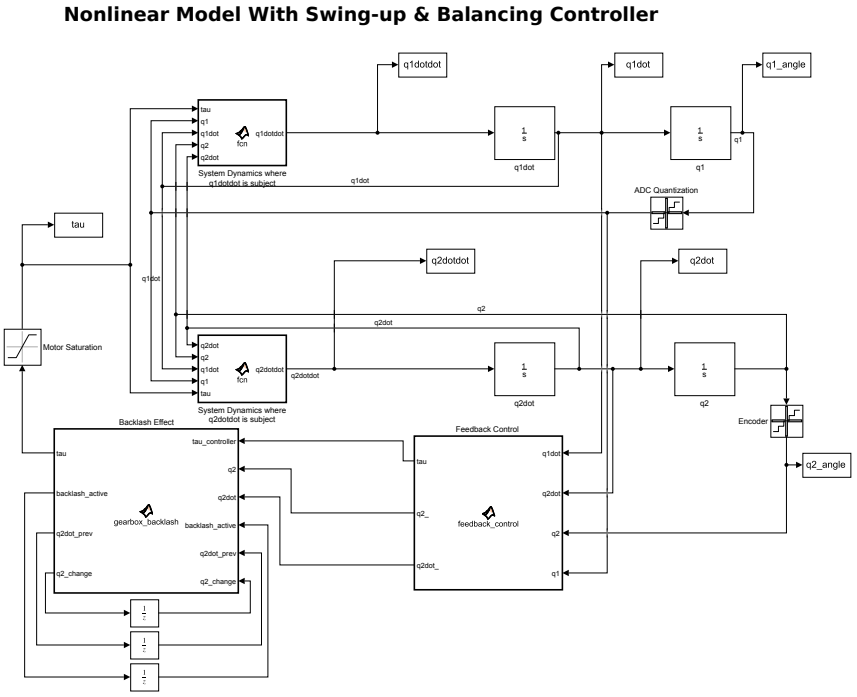


Figure 2.2: MATLAB Simulink Model

2.4 System Identification

The system identification tests are done to determine the characteristics that describe the behaviour of the system. These characteristics include the damping ratio's and natural frequencies of the system. These characteristics will be presented by showing measured responses and how these responses can be modelled.

The project started off with a previous physical model which provided realistic system parameters to allow the simulation to be an acceptable representation of a physical model. From using these previous system parameters the simulation provided a set of specification for the new mechanical design. All responses shown and values calculated are based of the new mechanical design parameters shown in Table 2.1.

The system is described by 2 independent parameters and is expected to contain 2 natural frequencies each accompanied by a damping coefficient. The first natural frequency of the system was determined by inspecting the response of the system when starting at a initial condition and keeping $\phi = 0\,\text{rad}$ throughout the response. This was done by using a lightweight PVC pipe that has negligible effect on the weight of the system. The actuated pendulum and unactuated pendulum are constrained to this pipe to ensure the 2 pendulums stay in-line with each other and thus ensuring $\phi = 0\,\text{rad}$. The response of the

System Parameter	Value
L_1	0.235 m
L_2	0.314 m
I_A	$0.0022\,\text{kg} \cdot \text{m}^2$
I_B	$0.0054\,\text{kg} \cdot \text{m}^2$
m_1	0.576 kg
m_2	0.492 kg
l_1	0.205 m
l_2	0.238 m

Table 2.1: System Parameters

system is shown in Figure 2.3 starting at a initial condition of roughly $\theta = \frac{\pi}{6}$. The accuracy of the initial conditions is of little importance, but must allow the response to contain a few oscillation to accurately determine the parameter of interest.

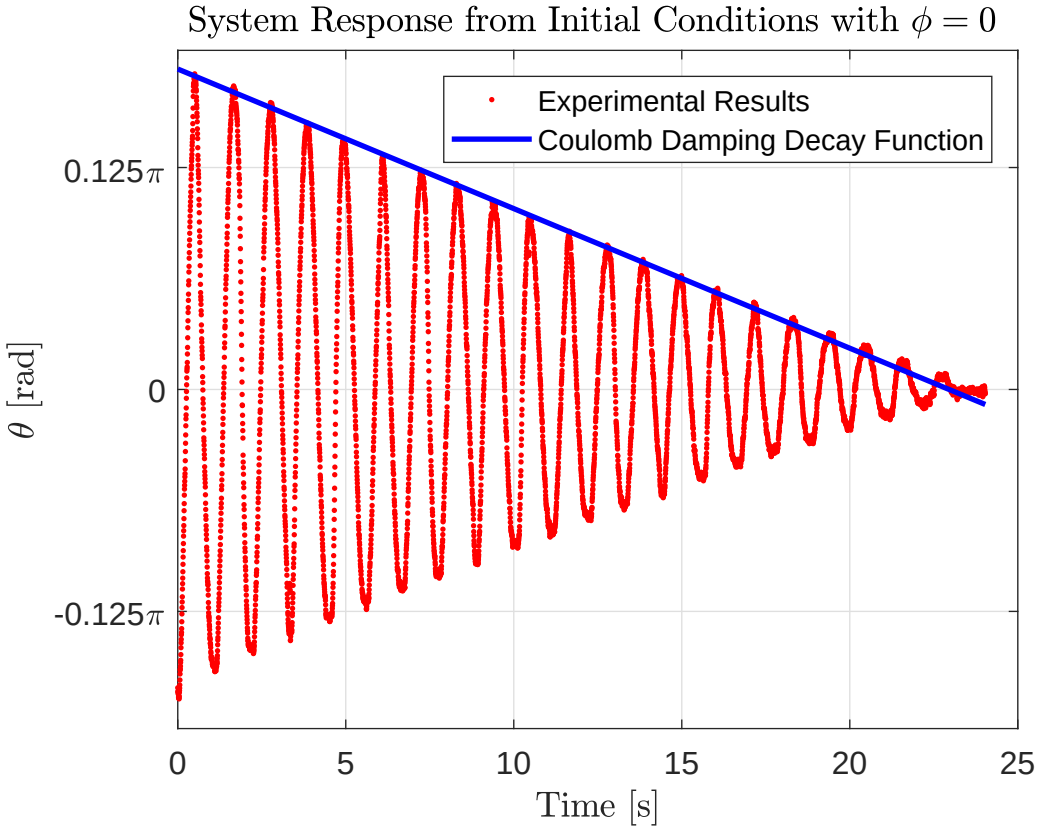


Figure 2.3: Initial Condition System Response while $\phi = 0\,\text{rad}$

The second natural frequency was determined by analysing the response of the system when ϕ starts at a initial condition and keeping $\theta = 0\,\text{rad}$ throughout the response. This was accomplished by constraining the unactuated pendulum using hard stops. Figure 2.4 shows the measured response of the system when ϕ starts at a initial condition and keeping $\theta = 0\,\text{rad}$.

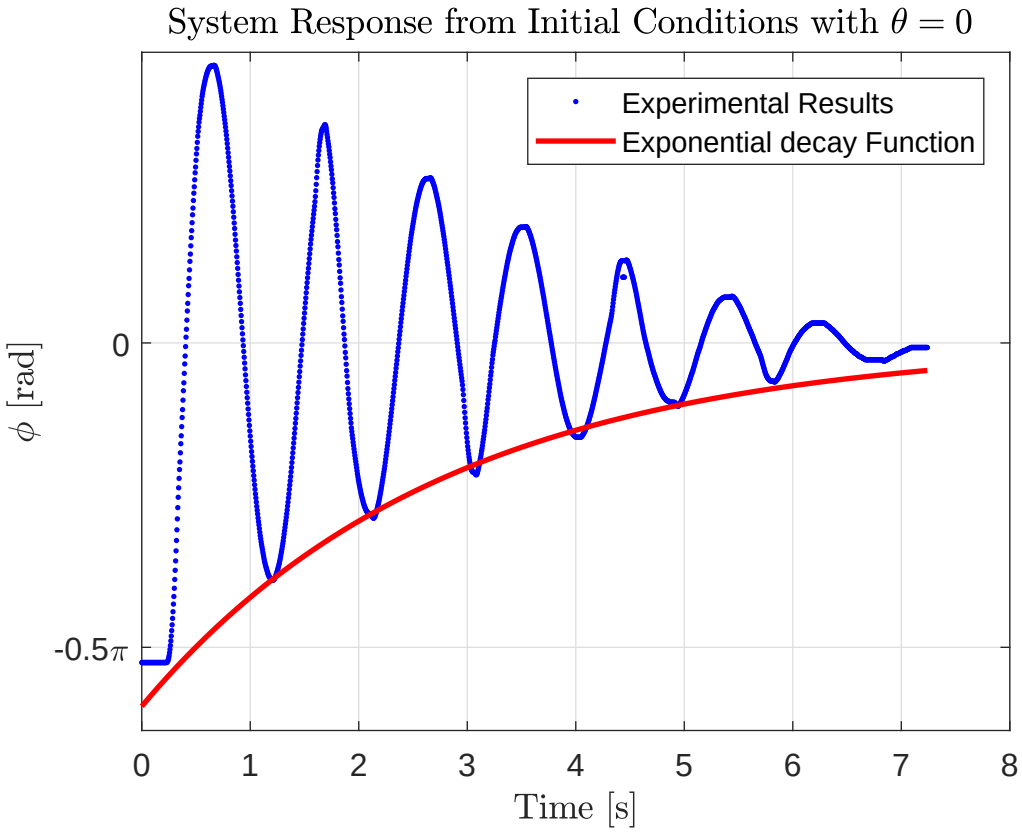


Figure 2.4: Initial Condition System Response while $\theta = 0\,\text{rad}$

The responses shown in Figure 2.3 and 2.4 can be characterise as equation (X) where the frequency of oscillation is due to the damped natural frequency. This damped natural frequency was determined by measuring the time difference between peaks. The practical results contains a variation on the time difference between peaks and thus the mean of the time differences were calculated and shown in Table **??**.

The friction forces, f_{c_1} and f_{c_2} that were set as unknowns in the derivation of the robotic gymnast will be expanded on in the following text by analysing how the decaying of the responses can be characterised.

The response shown in Figure 2.3 is under the influence of coulomb damping due to the response being characterised by the amplitude decaying linearly with a constant slope. Coulomb damping is caused by sliding friction and it's force is opposite to the direction of velocity (Inman, 2015). It is thus characterised as

$$f_{c_1} = \left\{ \begin{array}{cc} -\mu N & \dot{\theta} > 0 \\ 0 & \dot{\theta} = 0 \\ \mu N & \dot{\theta} < 0 \end{array} \right\}$$

It is shown in Inman (2015) that the slope is defined as

$$-\frac{2\mu N \omega_n}{\pi m g} \tag{2.13}$$

where N is the normal force. The slope seen in the decay function in Figure 2.3 was calculated using linear regression and by knowing the terms in equation (2.13) the combined μN term can be calculated and shown in Table **??**.

The response shown in Figure 2.4 is under the influence of viscous damping due to the amplitude decaying exponentially with time and this behaviour is modelled by the following equation:

$$\tau(t) = Ae^{-\zeta\omega_n t}$$

where ω_n is the natural frequency, ζ the damping ratio of the system and A represent the initial condition. The damped natural frequencies of the system has already been determined and linear regression was used to determined the best ζ that will fit the measured data. The decaying function is shown in Figure 2.4 with the ζ value shown in Table **??**. It is visible from the response that the damping ratio fits the data well and only starts to deviate near steady state.

The damping force, f_{c_2} that develops between the stator and rotor of the hinge can then be characterised as $2\zeta\omega_n(\dot{\phi} - \dot{\theta})$. The subtraction of $\dot{\theta}$ is due to the rotor of the hinge rotating relative to the stator.

2.5 Model Validation

The model implemented in simulation must be able describe the physical model to an acceptable degree to allow any further developments on the simulated model. The simulated model will be validated by comparing the experimental system characteristic values to those attained in simulation.

Table 2.2 shows the experimental values determined in the previous section against the simulation characteristic and indicates the simulation model

System	ω_{n_1}	ω_{n_2}	ζ_1	ζ_2
Experimental	5.692	6.793		
Simulation	5.654	6.704		

Table 2.2: Experimental Characteristics vs Simulation Model Characteristic

represents the physical model well.

Figure 2.5 and 2.6 provides an visually verification that the damping effects are modelled to an acceptable degree. It is visible from the responses that the simulated response becomes out of phase with time. This is expected due to the simulated model's natural frequency being larger and at this low frequency the effect of small difference is much larger over a large period.

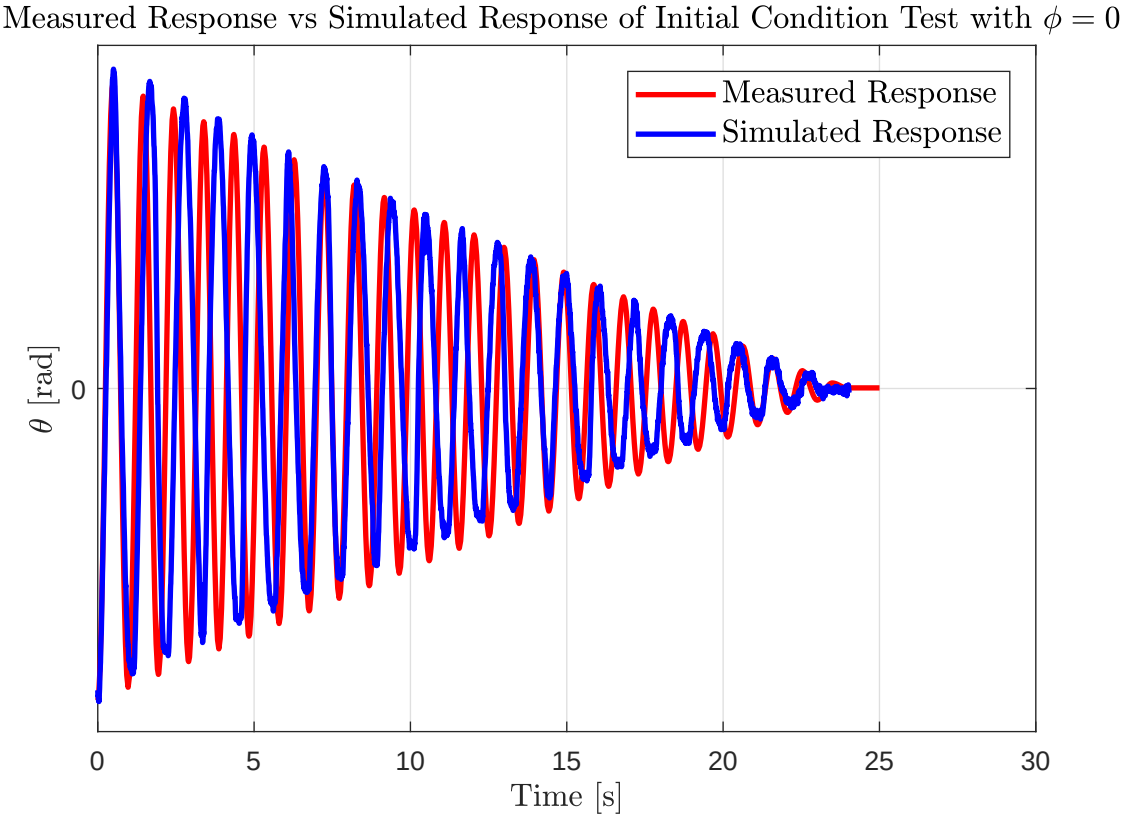


Figure 2.5: Comparison between Simulated and Measured Response with $\phi = 0$ rad throughout

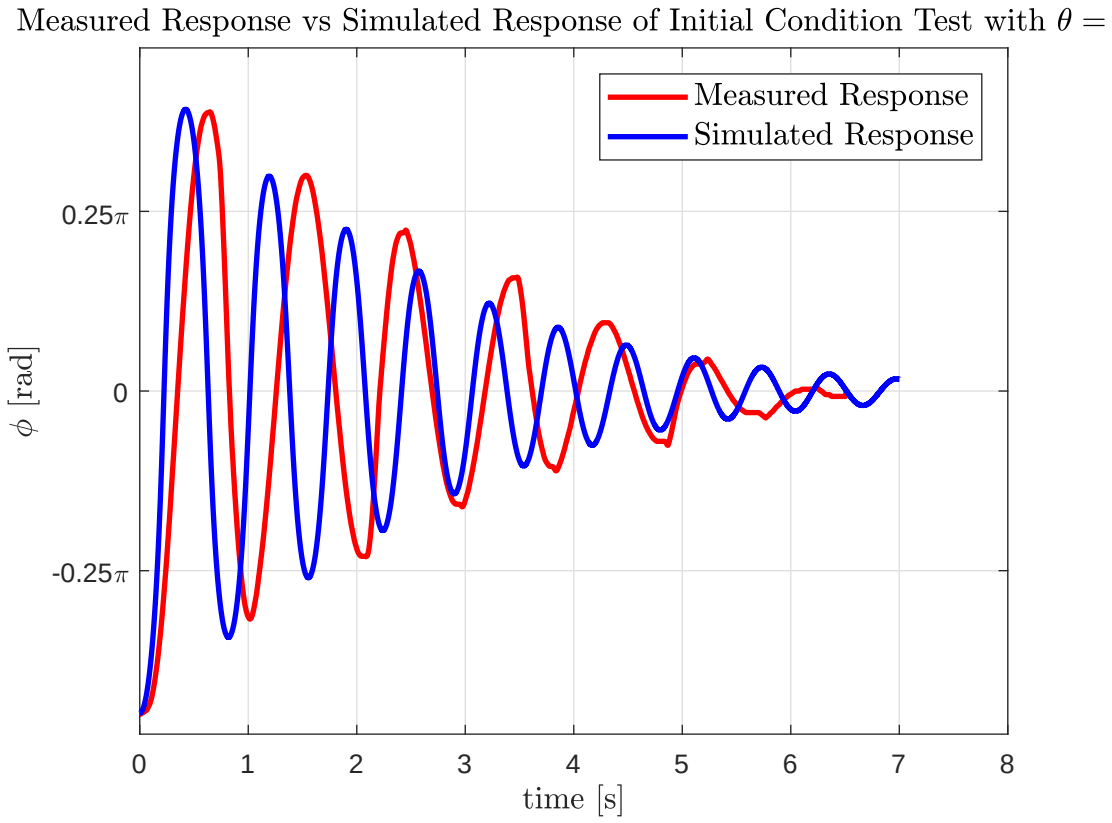


Figure 2.6: Comparison between Simulated and Measured Response with $\theta = 0\,\text{rad}$ throughout

Chapter 3

Feedback Control Design

3.1 Balancing Controller

Provided the robotic gymnast is in the vicinity of the unstable equilibrium position, a balancing controller is required to balance the system in the inverted position. The design approach was based on the premise that the swing-up controller will swing the robotic gymnast to the vicinity of the unstable equilibrium position where the balancing controller will take over. This section will focus on the aspects required to implement this balancing controller.

3.1.1 Controller Architecture

Figure 3.1 shows the block diagram that implements the balancing controller, and it is clear that the state space representation of the system was used. This requires the system to be a linear time invariant (LTI) system, but the system describe in equation (2.3) and (2.4) are not linear. This requirement was satisfied by linearising the system.

Another aspect of the balancing controller is that there are no reference input to instruct the controller to guide the robotic gymnast to the unstable equilibrium position. This is due to the system being linearised at the unstable equilibrium position and results that the unstable open-loop poles describe the

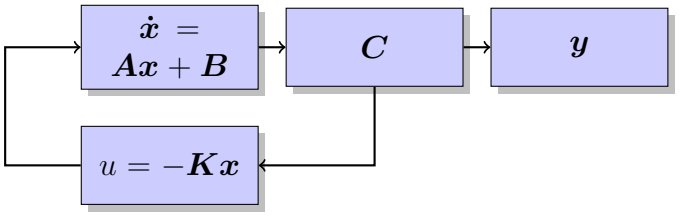


Figure 3.1: State Space Representation of the Balancing Controller

system response from initial conditions from the inverted position. Feedback is used to move these poles to stable locations and results in the closed loop system to decay to the unstable equilibrium position from initial conditions around the inverted position.

3.1.2 Requirements/Specifications and Constraints

The requirements set out for the balancing controller were to bring the robotic gymnast to the unstable equilibrium position from an initial condition range of:

$$\vec{q_i} = \left\{ \begin{array}{c} \theta > \frac{\pi}{1.1} \, \Lambda \, \theta < \frac{-\pi}{1.1} \\ \phi > \frac{\pi}{12} \, \Lambda \, \phi < \frac{-\pi}{12} \end{array} \right\}$$

The response must have a settling time of 0.8 seconds and a percentage overshoot M_p of 5%.

These requirements was selected to give the swing-up controller enough margin to bring the system to the vicinity of the unstable equilibrium position where the balancing controller is capable of balancing.

3.1.3 Plant Linearisation

As mentioned previously, to implement the state space representation the system must be a LTI system. This was achieved by using the Taylor Series Expansion to linearise the system at the unstable equilibrium position.

The independent parameters, ϕ and θ, will be condensed from now on as a vector describes as

$$\vec{q} = \begin{bmatrix} \theta \\ \phi \end{bmatrix}$$

The system is linearised at

$$[\vec{q_s}, \dot{\vec{q}_s}, \ddot{\vec{q}_s}]^T = [\pi, 0, 0, 0, 0, 0]$$

with the mathematical details shown in Appendix A.3. This linearised model can then be written in the state space form to implement a feedback gain. The state space variables are chosen as $\Delta\vec{q}$ and $\Delta\dot{\vec{q}}$ which results in the state space representation as:

$$\dot{\vec{q}} = \boldsymbol{A}\Delta\vec{q} + \boldsymbol{B}u$$

$$\vec{y} = \boldsymbol{D}\Delta\vec{q} + \boldsymbol{0}u$$

The poles of the system are identified by determining the eigenvalues of the $\boldsymbol{A}$ matrix. The linearised system remains a coupled system which results that the quadratic eigenvalue problem shown in equation (3.1) was required to be

solved to identify the poles. The solved quadratic eigenvalue problem results in the following eigenvalues using the system parameters in Table 2.1.

$$Q(\lambda) = \lambda^2 M + \lambda C + K \tag{3.1}$$

$$\vec{s} = \begin{bmatrix} 10.5742 \\ 4.9498 \\ -11.4905 \\ -5.0215 \end{bmatrix}$$

The eigenvalues of the system are all real indicating the response of the system when disturbed is an exponential function. This can be explained by realising the linearised system is modelled as a single pendulum. Once the single pendulum is disturbed from the unstable equilibrium position it would continue to rotate downwards and not with an oscillatory response.

3.1.4 Full State Feedback Design

The poles of the system are pairs of positive and negative real poles that indicate an unstable system. This is expected due to the system being linearised at the unstable equilibrium position. When the linearised system is at rest, any disturbance will result in a theoretically infinite growth of the state variables, but this behaviour can be controlled by introducing feedback.

These poles will be moved to the desired position by using the method of dominant poles. The method of dominate poles chooses a pair of the poles for the closed-loop system and select the other open-loop poles to have real parts with much larger natural frequencies. This allows the higher-order system response to be characterised as a second-order response (Gene F. Franklin, 2015).

Assuming a second order system and using the requirements defined, the pole locations can be calculated using equation (3.2) and (3.3)

$$M_p = \exp(\frac{-\pi\zeta}{\sqrt{1-\zeta^2}}) \tag{3.2}$$

$$t_s = \frac{4.6}{\zeta\omega_n} \tag{3.3}$$

and knowing the poles are described seen in equation (3.4)

$$p = \zeta\omega_n \pm j\omega_d \tag{3.4}$$

results in the desired pole locations as:

$$\vec{p} = \begin{bmatrix} -5.7500 + j5.8662 \\ -5.7500 - j5.8662 \\ -24.6429 \\ -24.6429 \end{bmatrix}$$

3.1.5 Simulation Response

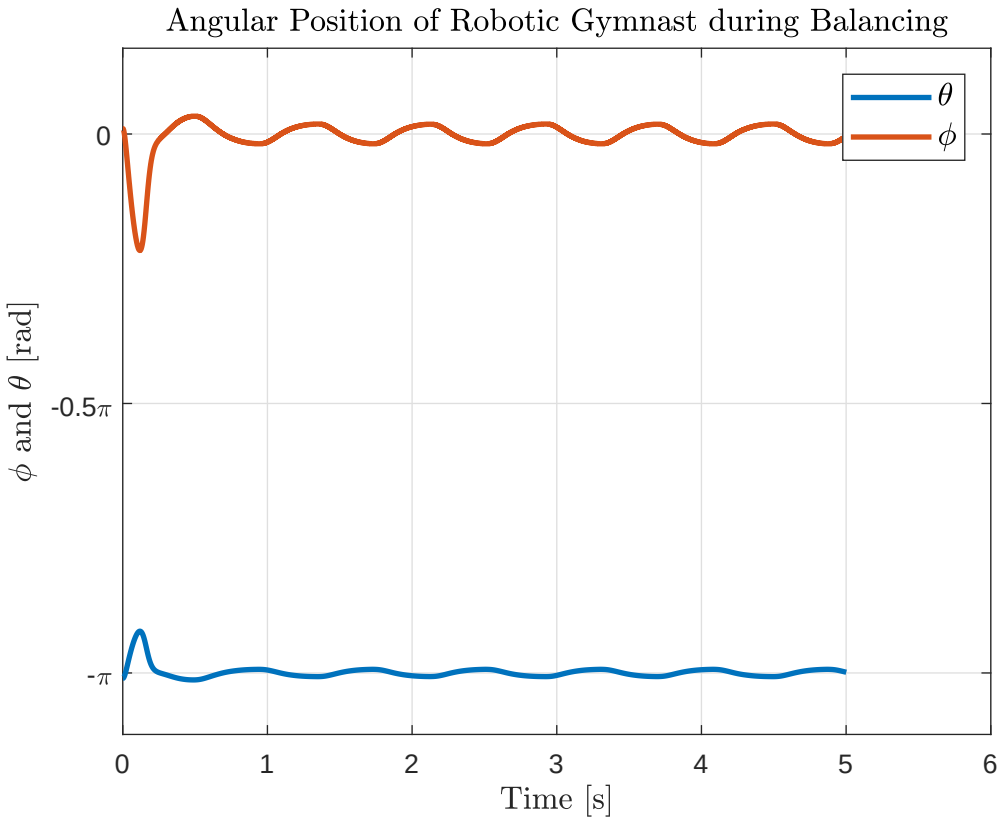


Figure 3.2: Balancing of the Robotic Gymnast

Figure 3.2 shows the robotic gymnast balancing around the unstable equilibrium position from an initial condition of

$$\vec{q} = \begin{bmatrix} \frac{\pi}{0.1} \\ \frac{\pi}{10} \end{bmatrix}$$

The oscillatory response around the inverted position is due to the non-linear coulomb damping that was approximated as viscous damping. This oscillatory response was decided to be acceptable due to the assumption that the effect of coulomb damping is negligible at low velocities. This assumption is visible

in Figure 2.3 where the response depart from the linear decay function near the stable equilibrium position.

The response shows that the controller meets the requirement of balancing the robotic gymnast from initial condition of $q \in []$ and reaches steady state within 0.8 seconds. The overshoot requirement of

3.2 Swingup Controller

For the robotic gymnast to swing from the stable equilibrium position to the unstable equilibrium position the feedback control must incorporate the non-linearities of the system. How these non-linearities of the system are incorporated and the design approach to swinging will be explained in the following section.

3.2.1 Controller Architecture

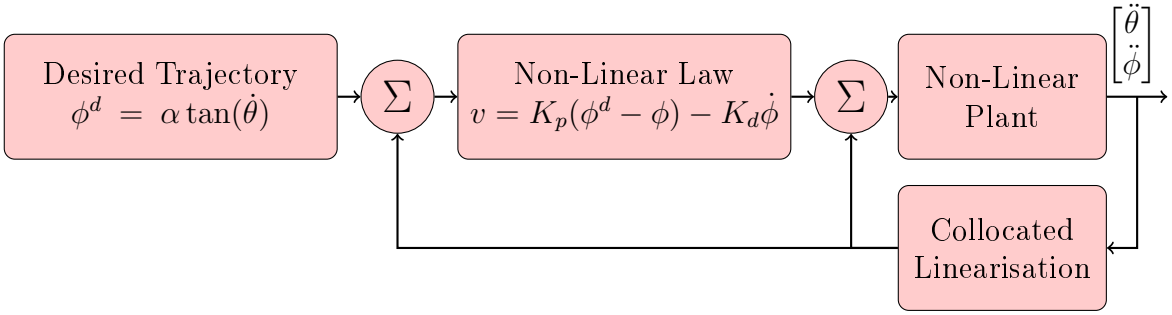


Figure 3.3: Block Diagram of the Non-Linear Controller

Figure 3.3 shows a high-level block diagram of the multiple parts of the swing-up controller. These parts are required to work in unison to allow the robotic gymnast to swing.

The collocated linearisation block is the foundation on which the entire controller was built. It creates a linear response from one of the outputs of the plant. This linear response can then be used to follow a desired trajectory.

The swing-up controller implements classical control theory approach where gains are selected to characterise the response of the system based on the error of the desired trajectory and the actual trajectory of the system.

Each of the blocks shown in the block diagram will be discussed in the section below.

3.2.2 Requirements/Specifications and Constraints

The requirements of the swing-up controller is to swing the robotic gymnast up under 30 seconds in the vicinity of the unstable equilibrium position. The vicinity of the unstable equilibrium position is defined as $\theta = 2\pi\,\text{rad} \pm \frac{\pi}{30}$ and $\phi \in [-5°, 5°]$.

Constraints that are placed on the controller are not to use 50% of the stall torque of the motor during the entire swing-up of the robotic gymnast.

The first requirement is to provide a feasible solution to the swing-up of the robotic gymnast and allow the swing-up sequence to be captivating. The second requirement is to ensure the linear approximation of the system is acceptable when the balancing controller is active to bring the system to the inverted position and balance. The constraint placed was to increase the safety in testing and prolong the life of the motor.

3.2.3 Feedback Linearisation

It has been shown that it is not possible to linearise the dynamics of the gymnast by means of static state feedback and non-linear transformation (Murray and Hauser, 1991), but it is possible to achieve a linear response from one of the outputs of the plant by implementing a non-linear feedback. This non-linear feedback is the partial feedback linearisation.

Collocated linearisation is a form of partial feedback linearisation where a non-linear control input τ is used to linearise the response of the actuated pendulum $\ddot{\phi}$. By analysing equation (2.4), the input τ was chosen to cancel all the non-linearities of the system and add an additional outer loop control input v_2 as seen in equation (3.5). This results in the unactuated pendulum to see a indirect force and the problem can be reduced to finding the outer loop control input to force the actuated pendulum to swing upwards (Spong, 1995).

The derivation of the collocated linearisation is shown in Appendix **??**.

$$\tau = d_{21}\ddot{\theta} + v_2 d_{22}\ddot{\phi} + h_2 + \psi_2 \tag{3.5}$$

$$d_{11}\ddot{\theta} + h_1 + \psi_1 = -d_{12}v_2 \approx F \tag{3.6}$$

$$\ddot{\phi} = v_2 \tag{3.7}$$

The subtle practical implication of using collocated linearisation is that the system being controlled must be well defined. If this is not the case the non-linear input τ will introduce other unwanted dynamics that could lead to undesirable behaviour.

3.2.4 Nonlinear Control Law

The ability to control the actuated pendulum to follow a desired trajectory, provides the possibility to increase the energy of the system if the correct trajectory is chosen. The increase of energy in the system will cause the pendulums to rise from their stable equilibrium position and start swinging upwards. The desired trajectory for ϕ was chosen as equation (3.8) determined by Spong (1995).

$$\phi^d = \alpha \arctan(\dot{\theta}) \tag{3.8}$$

This desired trajectory was derived by analysing a single pendulum and approximating the force it experience as seen in equation (3.6). By using this approximation Spong (1995) shows that the desired trajectory will increase the energy in the system. The desired trajectory also tries to allow the actuated pendulum to swing in phase with the non-actuated pendulum and by this approach the energy of the actuated pendulum is transferred to the unactuated pendulum (Spong, 1995).

The outer loop control input, v_2, then implements the classical control approach where gains are selected to characterise the response of the system based on the error seen in equation (3.9)

$$v_2 = K_p(\phi^d - \phi) - K_d\dot{\phi} \tag{3.9}$$

The coefficient α used in equation (3.8) constrains the actuated pendulum to stay within a interval of $\phi \in [-\beta, \beta]$ where $\alpha < \beta$ (Spong, 1995). This provides better control over the system to stay within the null controllability region when the system reaches the unstable equilibrium position.

Another side-effect of using the non-linear controller is that at rest the system will not start to swing-up. At rest, the condition are: $\phi = 0\,\text{rad}$ and $\dot{\phi} = 0\,\text{rad/s}$, and results that the control output seen in equation (3.9) will be zero. This effect was overcome by giving the system a small initial condition to start the swing-up controller.

3.2.5 Simulation Response

Figure 3.4 shows the swing-up controller swinging the robotic gymnast from the stable equilibrium position to the vicinity of the unstable equilibrium position with gain constants used shown in Table 3.1. There are a few interesting occurrences in the responses that were mentioned in the previous section to bring the reader's attention.

Firstly the robotic gymnast is required to start at an initial condition for the swing-up control law to be active and this is seen with θ starting at 11°.

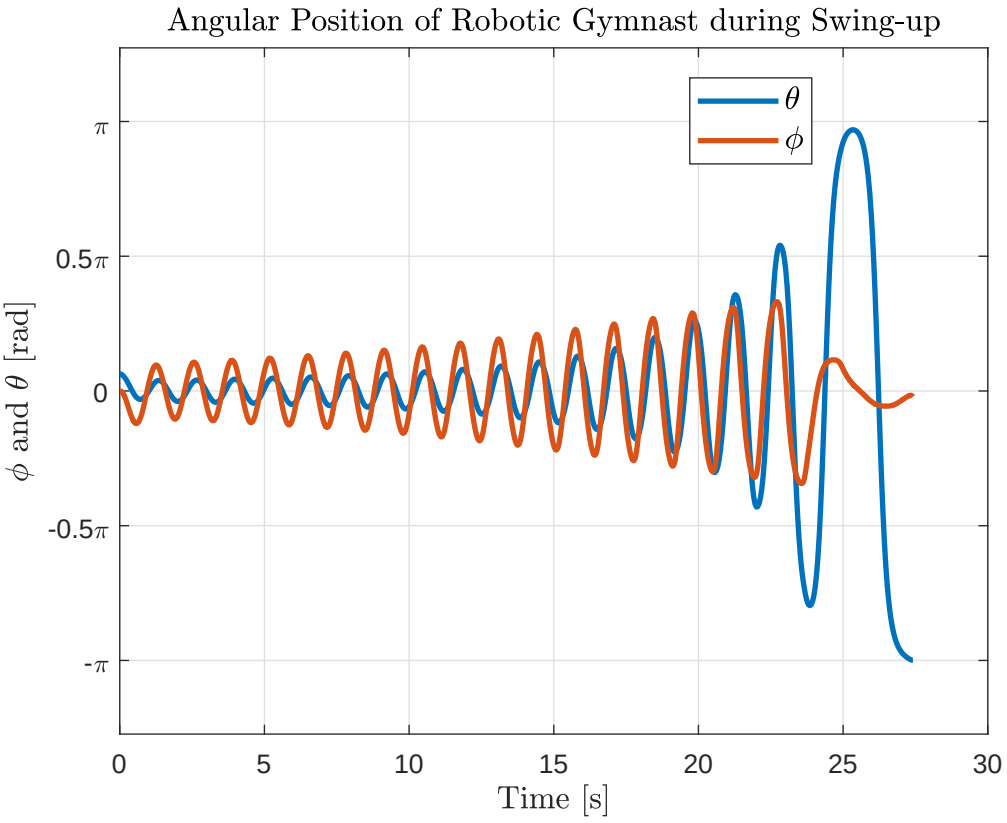


Figure 3.4: Swing-up of the Robotic Gymnast

Gain Constant	Value
K_p	58
K_d	14

Table 3.1:

Secondly the amplitude of ϕ is seen to suddenly decrease when the system nears the inverted position. This is due to α being reduced when the robotic gymnast nears the inverted position.

The response shows the swing-up controller meets the designed requirements by swinging the robotic gymnast to the vicinity of the unstable equilibrium within 30 seconds and θ and ϕ are in the designed region for the balancing controller to bring the system to the inverted position.

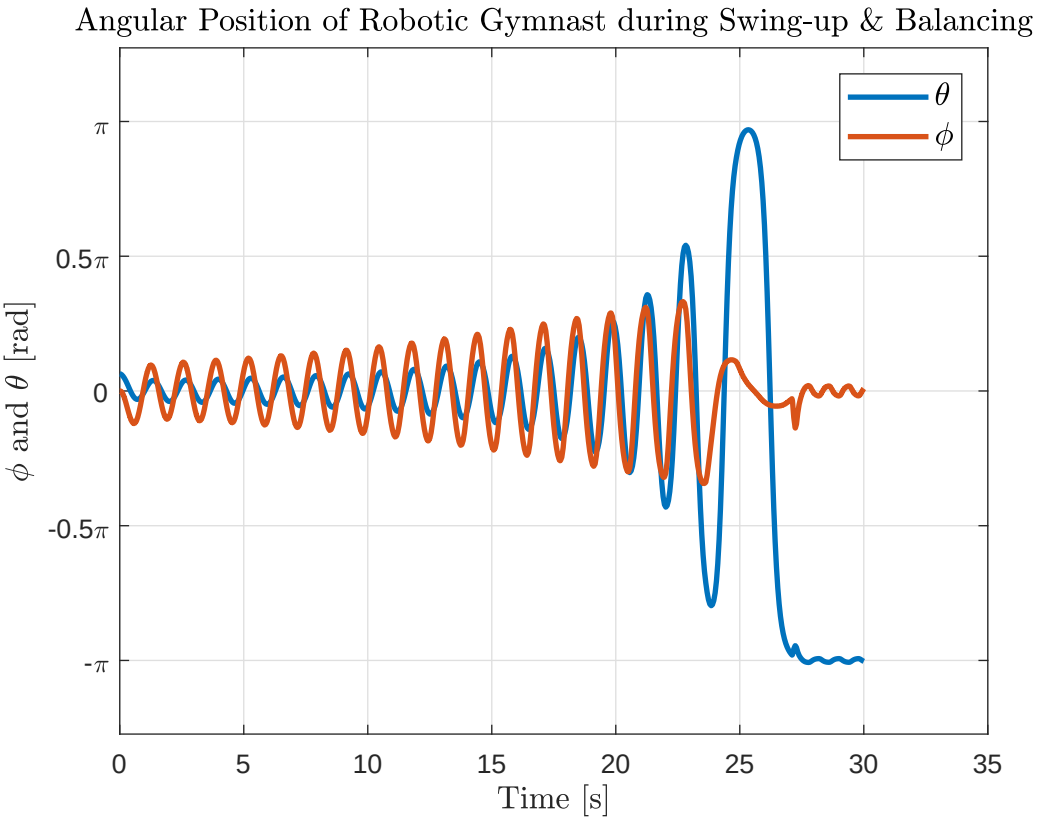


Figure 3.5: The Swing-Up & Balancing of the Robotic Gymnast

3.3 Simulation Results

Once both controllers were capable of meeting the requirements set out, they needed to be combined to achieve the swing-up and balancing of the robotic gymnast. Figure 3.5 shows the response of both controllers combined that achieved the swing-up and balancing.

During the swing-up the α value was reduced as the robotic gymnast started nearing the unstable equilibrium position as seen below.

$$\alpha = \left\{ \begin{array}{ll} \frac{\pi}{2} & \frac{-\pi}{2} < \theta < \frac{\pi}{2} \\ \frac{\pi}{12} & \\ \frac{\pi}{24} & \frac{-\pi}{1.1} > \theta\theta > \frac{\pi}{1.1} \end{array} \right\}$$

Chapter 4

Hardware Design and Implementation

4.1 Mechanical Hardware

The physical model allowed the abstract mathematical model to be implemented and experiments to be conducted. It will be presented by discussing the various components required and design decision made to implement a physical model that represents the mathematical model.

It is important that the physical model holds the assumption made during the derivation of the robotic gymnast. These assumptions include planar dynamics of the robotic gymnast and rigid body dynamics. The assumption of rigid body dynamics were easily met due to the forces acting on the pendulums results in negligible strain and thus elongation can be ignored. The assumption of planar dynamics comes in affect with the connection between the rotating shaft and the non-actuated pendulum. If the assumption holds there should be no vibration of the pendulum in any other direction than the rotating plane. This assumption is true in the mechanical design due to 4 screws used to keep the unactuated pendulum perpendicular to the shaft.

4.1.1 Assembly

Figure **??** emphasises important components that are discussed in the following section and explains their significance of use.

The electrical slipring converts the rotating wires that lead to the motor mounted on the unactuated pendulum to stationary wires allowing for free rotation and easy connection to the electrical design.

The bearing housing holds the ball-bearings in place ensuring for no vibration and misalignment. These ballbearings were press-fitted into the housing, ensuring a secure connection.

The potentiometer's shaft is connected to the shaft by means of a rubber tube. The rubber tube was chosen due to easy connection and allows for misalignment. The delay of measurement due to elasticity of the rubber is negligible due to the system experiencing low acceleration.

4.1.2 Structural Force Analysis

The stress analysis of the rotating shaft is presented to provide the reader confidence in the shaft design. It will be presented by calculating the static yield safety factor and discussing whether the safety factor is sufficient.

The shaft was analysed as a 3 point supported beam shown in Figure 4.1 with a constant diameter throughout, using the smallest diameter in the design. The forces acting on the beam is the torque T due to the rotating pendulums, and the normal force F due the centripetal force of rotation and weight of the pendulums.

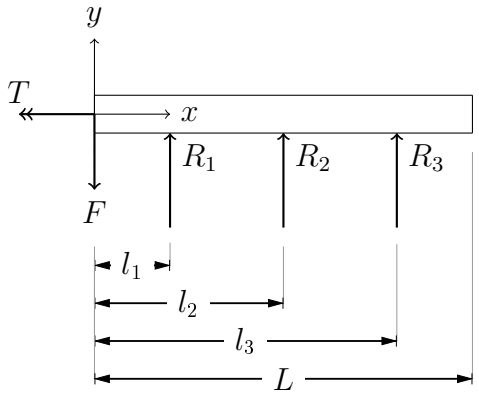


Figure 4.1: Model of Rotating Shaft as a Simplified Beam

The beam is a static undetermined beam and was solved using the principle of superposition and results in the reaction forces shown in equations (4.1), (4.2) and (4.3).

$$R_1 = 100\,\mathrm{N} \tag{4.1}$$

$$R_2 = \frac{8Fl_1[l_3^2 - l_2^2]}{l_3[4l_2^2 - 3l_3^2]} = 90\,\mathrm{N} \tag{4.2}$$

Stresses	$x = 0.1\,\mathrm{m}$	$x = 0.25\,\mathrm{m}$
σ_{f_1}	5	
σ_{f_2}	3.3	
σ_{f_3}	3.3	
τ_f	12	
M_{max}	12	

Table 4.1: Stresses developed in Shaft

$$R_3 = 33\,\mathrm{N} \tag{4.3}$$

The maximum bending stress occurs at $x = 0.1\,\mathrm{m}$ and the maximum torsion stress at $x = 0.25\,\mathrm{m}$. The safety factor of yielding will be determine at these two position.

Equations (4.5) and (4.4) are used to determine the maximum stresses at these two points and using the Mohor circle the principle stresses are determined. These stresses are shown in Table 4.1.

The conservative Von Mises theory shown in equation (4.6) was used to determine the safety factor on static yielding, where S_y is the yield stress of extruded aluminium. A safety factor on static yielding of 2 was determined. This is acceptable and indicates the shaft is capable of withstanding the load.

Fatigue due to cyclic loads were ignored due the shaft not rotating more than a 1000 rotation and the safety factor on static yielding on a conservative model is sufficient.

$$\sigma_f = \frac{M \cdot r}{I} \tag{4.4}$$

$$\tau_f = \frac{T \cdot r}{J} + \frac{F_{max}}{22} \tag{4.5}$$

$$n_s = \frac{S_y}{\sqrt{\frac{(\sigma_{f_1} - \sigma_{f_2})^2 + (\sigma_{f_2} - \sigma_{f_3})^2 + (\sigma_{f_3} - \sigma_{f_1})^2}{2}}} \tag{4.6}$$

4.1.3 Inertia of Pendulums

Figure 4.2 shows a functional drawing of the physical model to visually aid the reader in understanding how the inertia of the system was determined. In equation (2.3) and (2.4) the I_a and I_b represents the inertia of the unactuated and actuated pendulum about the axis coming out of the page passing through their center of mass respectively.

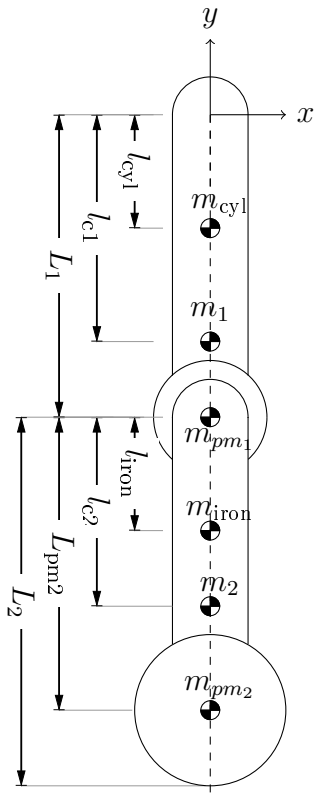


Figure 4.2: Simplified Drawing of Physical Model

Each pendulum contain parts that contribute to the pendulum's inertia due to each part representing a different physical form. How the inertia values shown in Table 2.1 were determined will be shown below with the physical system parameters shown in the Table 4.2

The actuated pendulum consist out of an aluminium square rod connected to the shaft and the motor mounting. The inertia of a square rod through it's center of mass is:

$$I_{cyl} = \frac{1}{12} m_{cyl}[w^2 + L_1^2]$$

The motor, gearbox and the motor mount was viewed as a point mass. It's inertia around the center of mass of the unactuated pendulum is:

$$I_{pointmass_1} = m_{pointmass_1} \cdot [L_1 - l_{c1}]^2$$

The total inertia of the of the unactuated pendulum around it's center of mass is :

$$I_a = I_{pointmass_1} + I_{cyl} + m_{cyl} \cdot [l_{c2} - l_{cyl}]^2$$

where the parallel axis theorem was used to move the aluminium rod's inertia to the center of mass of the unactuated pendulum.

Non-Actuated Pendulum	Value	Actuated Pendulum	Value
m_{cyl}	5	m_{iron}	
w_{cyl}	3.3	w_{iron}	
$m_{\text{pointmass1}}$	12	$m_{\text{pointmass2}}$	
l_{cyl}	12	l_{iron}	
L_{c1}	12	L_{c2}	
L_1	12	L_2	

Table 4.2: Physical Model Paramaters

Pendulum	Center of Mass [m]
Non-Actuated	0.2056
Actuated	0.2115

Table 4.3: Center of Mass for Each Pendulum from their Rotating Hinge

The actuated pendulum contains similar parts as the unactuated pendulum: a pointmass at the bottom and an iron rod connected to the motor shaft. Determining the inertia of the actuated pendulum is thus exactly the same as the unactuated pendulum.

During the simulation it was noticed that the ratio between the unactuated and actuated pendulum $\frac{I_b}{I_a}$ should be 1 or greater for the swing-up controller to bring the robotic gymnast to the unstable equilibrium position in a feasible time frame. This is the reason adding another pointmass to the actuated pendulum to compensate for the inertia the motor creates.

4.1.4 Center of Mass

Each pendulum can be seen as a system containing discrete components, where each components center of mass is easily identified. Both iron rod and aluminium rod center of mass is in the middle of their rod lenght and the rest are seen as point masses. However each of these components center of mass contribute to the center of mass of each pendulum.

The center of mass for each pendulum from their hinge of rotation shown in Table 4.3 was calculated using equation (4.7).

$$\vec{r} = \frac{\sum_i^j r_i m_i}{\sum_i^j m_i} \tag{4.7}$$

4.1.5 Motor

The chosen motor used during experiments was the *Faulhaber DC 3257 012 CR* micromotor. It was used in combination with the *Faulhaber Planetary Gearhead 32/3* Series. The gearbox is a 2 stage reduction gearbox with a overall rounded reduction ratio of 14:1. The motor is capable of providing a stall torque of 539 mNm which is a converted output torque from the gearbox of 7.646 Nm.

The motor terminal connection is connected directly to the PCB of the electronic design being routed through the slipring. The motor assembly contains a encoder for position measurements and it's signal and power connection is also routed through the slipring and connected to the PCB.

4.2 Electronic Hardware

The electronic hardware was a crucial component for the successful implementation of the robotic gymnast. The electronic design provided the means to determine the system characteristic and verification of the simulated model. It will be presented by discussing the various components implemented to achieve the results in this report.

4.2.1 System Description

Figure 4.3 provides a system overview on how the different subsystems functions together and what inputs are outside from the PCB design.

The micro-controller receives the different signals that has been correctly conditioned from supporting circuitry to interpret the dynamics of the system. From these signals it is able to output the correct signals to instruct the next command.

The digital logic circuit that consist out of logic level converters acquires the signal from the microcontroller and performs signal conditioning to interface with the motor driver and determines the correct direction to rotate the motor.

The motor driver controls the DC brushed motor based of the digital signals and provides a proportional feedback current that is delivered to the unity-gain amplifier.

The motor contains an encoder that indicates the direction and position of the rotor through digital signals that is sent through a digital logic filter to

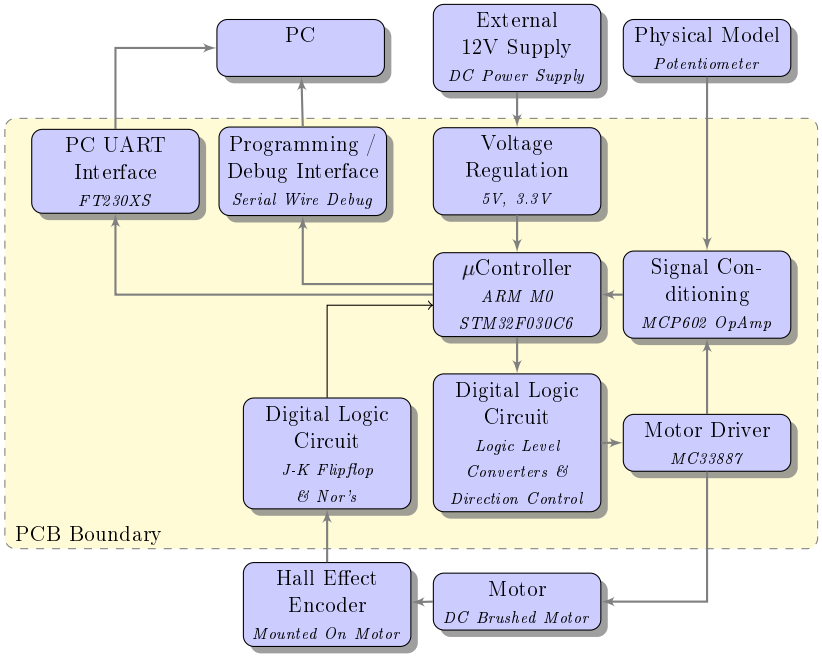


Figure 4.3: Electronic System Overview

retrieve only critical information from the encoder signals.

The physical model contains a potentiometer that measures the unactuated pendulum's angle and this signal is sent to the unity-gain amplifier.

The microcontroller will use the UART interface as it's data acquisition protocol to send the necessary information to the computer.

The micro-controller is programmed using the Serial Wire Debug (SWD) protocol to transfer the binaries from the computer.

Power is provided using a external 12V power-supply, which will power the motor, but also using a regulator to down convert/step to a 5V and 3.3V to power the microcontoller and the other peripherals.

4.2.2 Microcontroller

The microcontroller chosen is the STM32F030C6. The selection was done according to the ease of setting up, memory size, physical dimensions and the peripherals it provided. These selection are expanded below.

The STM32F030C6 is based of the ARM M0 architecture which is ARM's entry level microcontroller architecture. It requires little support circuitry to have a up and running microcontroller with only the SWD protocol to program

the chip.

It was difficult to determine the memory size specification for the project. This uncertainty ensured that the largest memory size the ARM M0 architecture could provide was selected.

The Electrical and Electronic Department's PCB manufacturing machine can only provide a minimum track width of 0.3 mm. This resulted in choosing a microcontroller whose footprint would meet this requirement.

Based on the conceptual design, the chosen microcontroller required to contain 2 ADC's channels, minimum of 5 GPIO's and 1 serial communication peripheral.

From these requirements and specification the STM32F030C6 was the best candidate to meet all of the above.

4.2.2.1 Programming / Debug Interface

The *Atollic TrueSTUDIO for ARM 8.0.0* Integrated Development Environment (IDE) was used for writing the source code which converts the source code to the Executable and Linkable Format (.elf) file. These .elf files is then written using the SWD protocol to the microcontroller. Debugging of the source code occurred using the same IDE which allows the programmer to inspect variables, timers and logic.

4.2.2.2 PC UART Interface

The purpose of the Universal Asynchronous Receive Transmit (UART) to serial communication was for the data acquisition of the system response, debugging and sending the control input that was computed on the external computer.

The external computer executed a Python script that was listening for any activity on the computer's driver port for information about the system. The Python script would react differently based on whether debugging, system identification tests or experiments occurred.

The UART to Serial chip used was the FT230x (USB to BASIC UART IC). It was chosen due to the easy support circuitry it requires with the option to use LED's to indicate any activity on the receive (Rx) & transmit (Tx) communication lines.

The UART to serial circuit was tested by doing a loopback test. The loopback test consist of connecting the Tx and Rx lines together. This results

Component	Supply Voltage [V]
Digital Logic, Op-Amp & Sensors	5
Microcontroller	3.3
Motor Driver	12

Table 4.4: Suppy Voltage's for the different components

in the circuit echoing anything that the receiver has sent. The schematic of the circuitry is shown in Appendix A.5.

4.2.3 Voltage Regulation

The various components used in the electronic design required different supply voltages and are tabulated in Table 4.4.

The 12 V supply was provided using an external bench power-supply and were done converted to 5 V and 3.3 V using linear voltage regulators. The schematic for each voltage regulator are shown in Appendix A.5, and includes a Light Emitting Diode (LED) to ensure the minimum load was met for each regulator. The LED also acts as a visual debugging method.

4.2.4 Potentiometer Sensor

4.2.4.1 Working Principle

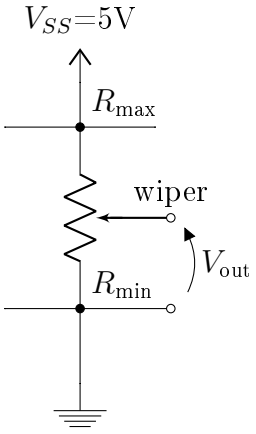


Figure 4.4: Simplified Model of a Potentiometer

The rotary position potentiometer consist out of a wiper that is attached to a rotating shaft. This wiper moves across a internal resistor as the shaft rotates and changes the effective resistance across the output terminal. It

provides thus a proportional voltage to it's position as seen in Figure 4.4 that indicates the position.

4.2.4.2 Interface

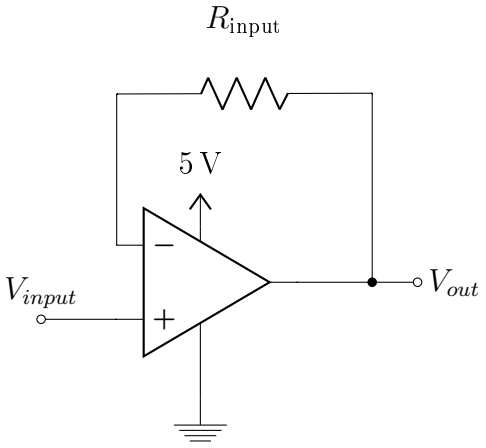


Figure 4.5: Unity Gain Amplifier Circuit

The signal produced by the rotary potentiometer varies from 4.95V and 50mV as it rotates from 360° to 0°. This signal was sent through a simple voltage divider circuit to reduce the signal to 3V and 15mV to be within the sampling limits of the microcontroller. The scaled voltage was then sent through a unity gain rail-to-rail amplifier, where the mirrored output signal is fed into the ADC.

The type of ADC used in the STM32F030C6 is a successive approximation register (SAR) and contains an internal capacitors that suffers from the effect of being depleted if the sampling frequency is to high (stm, 2017). Using an operational amplifier reduces the risk of depleting this internal capacitor because of the low output resistance. The schematic of the circuit is shown in Appendix A.5.

4.2.5 Magnetic Encoder

4.2.5.1 Working Principle

A rotating gear containing ferrous metal teeth is attach to the rotating shaft. The rotating metal teeth rotates near a hall-effect sensor which creates a change in the magnetic flux inside the hall-sensor. This change in magnetic flux is sensed by the hallsensor which produces a digital signal (Instruments, 2006).

4.2.5.2 Digital Interface

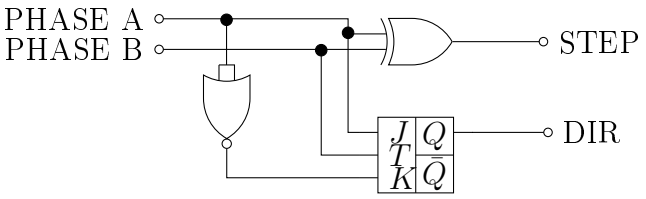


Figure 4.6: Digital Logic Circuit containing JK-Flipflops, XOR- and NOR Gates

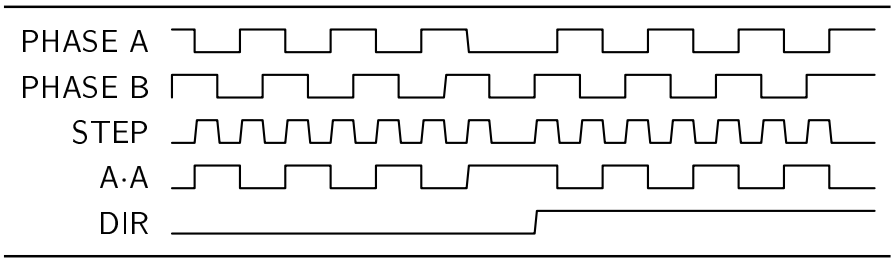


Figure 4.7: Waveform of the JK-Flipflop,XOR, and NOR Gate Circuit

The encoder contains a sold state hallsensor which provides 2 channels with a 90° phase difference between them. (Faulhaber, 2011). These 2 signals under go a hardware filter that produces 2 signals that indicate the direction of the motor and the incremental position. The hardware filter was implemented to reduce the processing time the microcontroller was required to do to determine the incremental position and the rotational direction.

The hardware filter consist out of a XOR, NOR and JK-Flipflop gates shown in Figure 4.6 and the schematic in Appendix A.5. The XOR gate produces the incremental change of position of the motor which is then read by the microcontroller using interrupts on rising and falling edges. The output of the XOR gate is shown in Figure 4.7.

The resolution of the encoder is 16 lines per revolution per channel, equalling to a combined 64 rising- & falling edges in total. The encoder is connected directly to the motor shaft whose speed is reduced by the 14:1 reduction gearbox. The encoder will thus rotate 14 times per output shaft revolution, increasing the resolution to 896 edges per revolution.

The NOR and JK-Flipflop combination produces the direction of the motor by determining whether phase A leads or lags phase B by 90°. This leading or lagging was indicated by a logical 1 or 0 which is read by the microntroller

shown in Figure 4.7.

4.2.5.3 Logic Level Converters

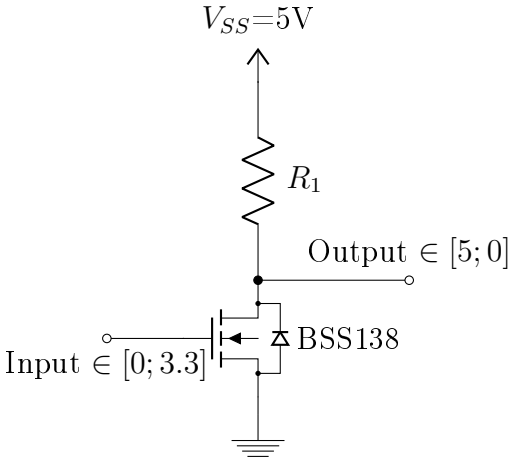


Figure 4.8: Logic Level Converter & Inverter Circuit

The microcontroller is required to interface with the motor driver and represent a logical high and low as a 3.3 V and 0 V respectively. The motor driver's logical high threshold is 3.5 V. It is thus required to use a logic level converter to allow reliable communication between the two devices.

The logic level converter used shown in Figure 4.8 uses the BSS128 transistor. The circuit shown also acts as a inverter where a logic low, 0 V by the micro-controller will be converted to a 5 V and a logic high, 3.3 V will be converted to 0 V. This side effect was overcome by inverting the desired respons es in software.

4.2.5.4 AND Digital Circuit

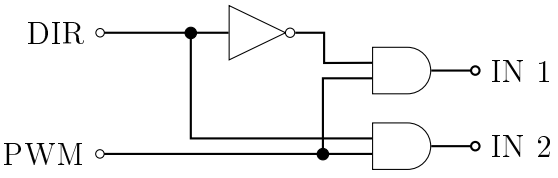


Figure 4.9: AND digital logic with inverter

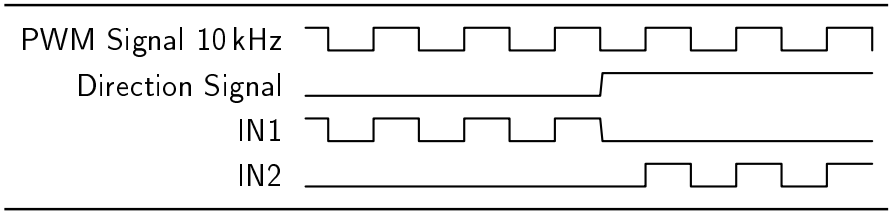


Figure 4.10: AND Digital Logic Circuit Waveforms

The motor driver contains 2 input pins which controls the voltage polarity of the motor terminals. Keeping the one input high and the other low will turn the motor in the one direction and switching the logical values on these inputs will turn the motor on the other direction. Adding speed control requires the PWM signal that the motor receives to be alternated on these inputs and are done by the AND digital circuit.

The AND circuit receives 2 signals from the microcontroller after it has been converted to the correct logic level: the PWM signal and a logic level signal indicating the desired direction. Based on the directional signal the AND circuit will switch the PWM signal between the 2 inputs of the motor driver while holding the other low as seen in Figure 4.10.

This hardware directional control was done in order to reduce the processing time the microcontroller was required to do to switch the generated PWM signal between the 2 inputs of the motor driver.

4.2.6 Motor Driver

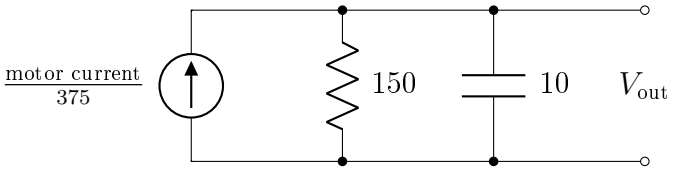


Figure 4.11: Simplified Circuit of Motor Feedback

The motor driver chosen was the MC33887. It was selected based on providing the motor up to 6 A of current, while withstanding the high current transients due to the fast switching of a inductive load (mot, 2007). The motor driver provides the motor with 12V DC which is externally provided by a DC power supply. The schematic of the motor driver is shown in Appendix A.5.

The motor driver is connected directly to the motor and responsible for directional and rotational control of the brushed DC motor. The motor driver

contains 2 half H-bridges that forms a full H-bridge which are Pulse-Width-Modulated (PWM) to control the speed of the motor and originates from the microcontroller. The selected frequency was 10kHz and is recommended by the manufacturers (mot, 2007). As discussed previously, the signals' logic level was first converted and then sent through the AND digital filter before the motor driver receives it.

The MC33887 provides a proportional current of $\frac{1}{375}$ of the current flowing through the high-side of the full H-bridge (mot, 2007). This current is sent through a resistor of $150\,\Omega$ to provide a voltage signal to represent the current. Due to the motor being controlled using PWM, the current is a periodic impulse signal making it almost impossible for the ADC to sample. This problem was overcome by adding a parallel capacitor to the resistor to create a ripple voltage. This ripple voltage is sent through an unity-gain amplifier as seen in Figure 4.5 before it is sampled by the micro-controller. The R_{input} resistance is the input resistance that the operational amplifier sees. This closes the feedback loop to implement torque control by the control system.

4.2.7 Verification Tests

4.2.7.1 Angle Sensor Measurements

4.2.7.2 Optical Encoder Measurements

4.2.7.3 PWM Duty Cycle to Motor Current

The model describing the system in equation (2.4) assumes the torque delivered to the system is instantaneously available. This is inaccurate due to the model describing the DC motor is a second-order differential equation containing it's own time constant. This model was not incorporated due to adding another control loop would add more delays to the control systems. This was overcome by mapping the torque the motor provided to the duty cycle the motor receives.

Experiments were done to determine the relationship between the duty-cycle of the PWM signal and the torque delivered by the motor. These experiments are done by incrementing the duty-cycle of the PWM signal that the motor receives when the shaft is kept fix against a hard-stop. The mean value of the output voltage from the circuit shown in Figure 4.11 was measured on the oscilloscope and mapped backwards to determine the torque using equation (4.8) with the constants shown in Table 4.5

$$\tau = \frac{V}{R} \cdot GR \cdot FR \cdot \alpha_t \tag{4.8}$$

Figure 4.12 shows the measured data with a line of best fit. It is clear that there exist a linear relationship between the duty-cycle of the PWM signal and

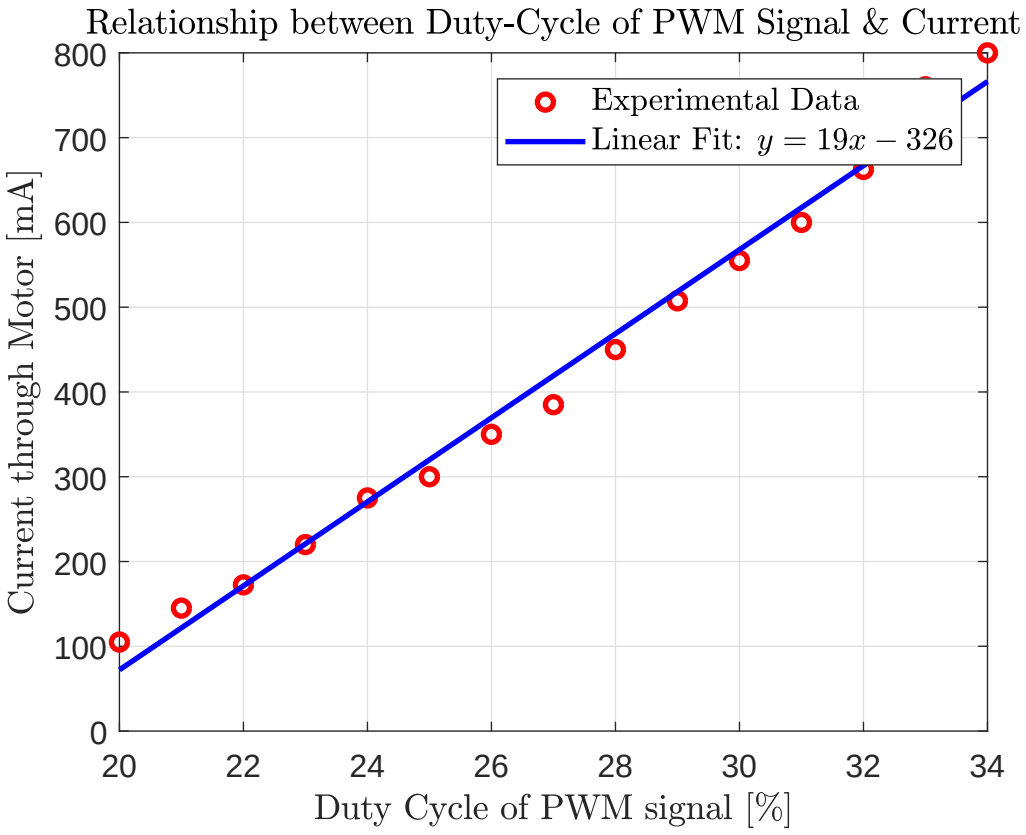


Figure 4.12: Relationship between Duty-Cycle of PWM Signal and Current through Motor

Constant	Description	Value
GR	Gear Ratio	14
FR	Feedback Current Ratio	375
α_t	Torque Constant	19.1×10^{-3}
R	Resistance	150

Table 4.5: Values of Constants used in Equation (4.8)

the torque provided. This equation of best fit will thus be used to output the correct torque determined by the control system.

Chapter 5

Software Design

5.1 Software Requirements

The software required to implement the retrieval of system state information, debugging and control laws are presented here. The software design played a central role in continuing the project pass the simulation phase. The software design allowed the determination of the system characteristic parameters and the verification of the simulation and controller.

5.1.1 Communication

Communication between the external computer and the microcontroller occurred differently based on whether an experiment was done or testing different parts of the electronic system. These difference are explained in this section.

If experiments are conducted the communication are bi-directional between the microcontroller and the external computer. The microcontroller streams the state variables of the system to the external computer in the structure shown in Figure 5.1. The star attached to the variables indicate that they are not sent in the correct units due to sending data types such as floats are computation hungry and are rather handled on the external computer. The external computer will then translate the variables into their correct units and compute the control input based on the control law. The external computer sends the control input back to the microcontroller to output the correct signals.

The structure used in Figure 5.1 was chosen as comma-seperated values (.csv) which makes it easy to write the data in a .csv file to analyse later.

The other state in which communication occurred was used for debugging purposes. In this state the *Python* script allows the user to type commands adhering to the structure shown in Figure 5.2. Based on the command used,

0	1	2	3	4	5	6	7
$	time	,	θ^*	,	ϕ^*	,	τ^*

Figure 5.1: Data Structure for Streaming Data during Experiments

the microcontroller would echo the same command back if it completed the command instructed. These commands included to receive the sampled values of signals, manual control over the duty-cycle of the PWM signal and directional control of the motor. It also acted as a soft layer for safety by sending commands to arm the system before experiments. A summary of the possible commands are given in Appendix A.4.

0	1	2	3	4	...	n-1	n
$	cmd	,	value	value		\r	\n

Figure 5.2: Data Structure for Sending Commands

5.1.2 Embedded Program

Figure 5.3 shows the main execution flow of the microcontroller based on factors that influence their states. A brief overview of the execution flow is described below.

On startup of the microcontroller, the various peripherals required for operation are initialised. They include timers for PWM generation, ADC for sampling and interrupts for encoder signals. The microcontroller will then initialise the various variables required for operation.

Once initialisation was completed the microcontroller will check via an interrupt if a byte over the serial communication has been received. If a byte has been received, the microncontroller will verify the command structure and execute based on this command.

Every 0.1ms the microcontroller will inspect the data arrays of the sampled signals. Direct Memory Access (DMA) was used for sampling and results in the sampled data to be automatically refreshed by hardware.

The microcontroller will then react whether a interrupt has occured to indicate a rising or falling edge on the encoder signal. This falling and rising edge indicates a incremental change of the position of the actuated pendulum

and the microcontroller will behave accordingly.

The microcontroller will than verify whether it is required to stream the data every 4ms to the external computer for system identification tests.

5.1.3 Controller

The controllers of the swing-up and balancing are implemented on the external computer in a *Python* script and required more information about the system state than what was received from the microcontroller. How these missing information was determined is explained in this section.

The information the swing-up and balancing controllers required was the angular velocity of both the actuated and unactuated pendulum. This information was acquired using numerical differentiation by using the 3-point backwards method shown in equation (5.1).

$$f'(x) \approx \frac{-3f(x) + 4f(x - \Delta t) - f(x - 2\Delta t)}{2\Delta t} \tag{5.1}$$

The swing-up controller implements multiple cosine and sine functions. These functions were discretised into a lookup table up to the accuracy provided by the sensors. The discretisation was done to decrease the processing time done to compute the next control input command.

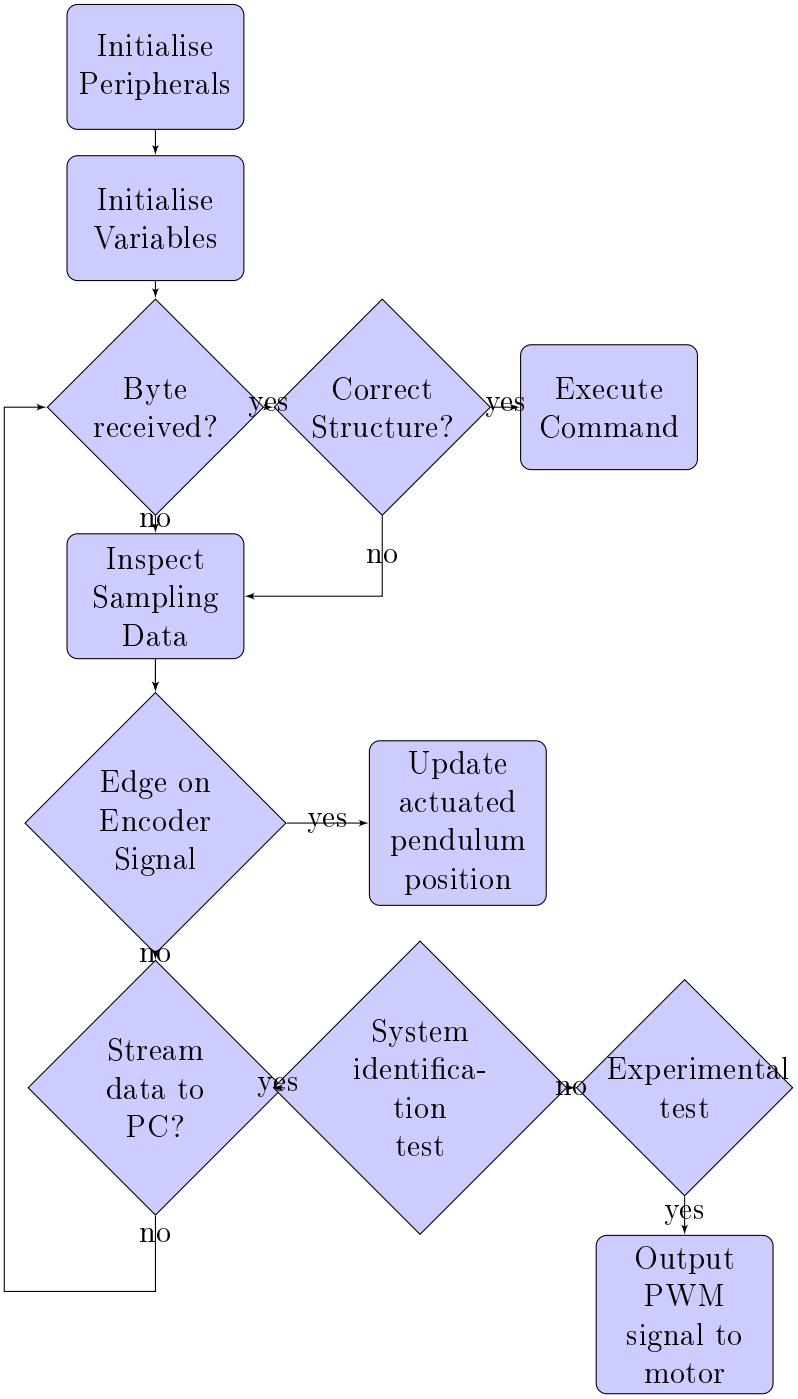


Figure 5.3: Embedded Software Flow

Chapter 6

Practical Results

6.1 Swingup Controller

This section provides the practical results of the swing-up controller and discusses the differences it contains with the simulation.

Figure ?? shows the practical results of the swing-up controller and contains interesting behaviour that the simulation does not contain. The swing-up controller did not require a initial condition as in the simulation. The system started to swing-up from the rest position because of the noise being introduced by the ADC. This noise is amplified by using the numerical differential algorithm to calculate the angular velocity of the pendulums that is required by the swing-up controller. The swing-up controller reacts on this small error which results in the motor to output a torque and provides this small initial condition.

6.2 Balancing Controller

6.3 Swingup and Balancing

Chapter 7

Conclusion

7.1 Summary

7.2 Recommendation

A short description of the various improvements that can be made if future attempts are made on *The Feedback Control Of A Robotic Gymnast* with the hope to improve the results given in the report.

Choosing a more advanced microcontroller that is able to use a baudrate greater than 115200 would improve response time and provide better control of the system.

Implementing a digital low pass filter to remove the noise on the sampled signal. Especially on the signals that are required to be numerically differentiated.

The script executing on the external computer can be written in a language such as C or C++ to increase the execution time of the script.

Appendices

Appendix A

A.1 Derivation of the Mathematical Model

$$x_1 = l_1 \cos(\theta)$$
$$y_1 = -l_1 \sin(\theta)$$

$$x_2 = L_1 \sin(\theta) + l_2 \sin(\theta + \phi)$$
$$y_2 = -L_1 \cos(\theta) - l_2 \cos(\theta + \phi)$$

$$\dot{x_2} = L_1 \cos(\theta)\dot{\theta} - l_2 \cos(\theta + \phi)(\dot{\theta} + \dot{\phi})$$
$$\dot{y_2} = L_1 \sin(\theta)\dot{\theta} + l_2 \sin(\theta + \phi)(\dot{\theta} + \dot{\phi})$$

$$x_2^2 = L_1^2 \cos(\theta)^2\theta^2 + l_2^2 \cos(\theta + \phi)^2(\dot{\theta} + \dot{\phi})^2 + 2L_1 l_2 \dot{\theta}(\dot{\theta} + \dot{\theta}) \cos(\theta) \cos(\theta + \phi)$$
$$y_2^2 = L_1^2 \sin(\theta)^2\theta^2 + l_2^2 \sin(\theta + \phi)^2(\dot{\theta} + \dot{\phi})^2 + 2L_1 l_2 \dot{\theta}(\dot{\theta} + \dot{\theta}) \sin(\theta) \sin(\theta + \phi)$$

$$x_2^2 + y_2^2 = L_1^2\theta^2[\cos(\theta)^2 + \sin(\theta)^2] + l_2^2(\dot{\theta} + \dot{\phi})^2[\cos(\theta + \phi)^2 + \sin(\theta + \phi)^2] +$$
$$2L_1 l_2 \dot{\theta}(\dot{\theta} + \dot{\phi})[\cos(\theta) \cos(\theta + \phi) + \sin(\theta) \sin(\theta + \phi)]$$

Using the following trigonometric identities

$$\cos(\gamma)^2 + \sin(\gamma)^2 = 1$$

$$\cos(\gamma) \cos(\alpha) + \sin(\gamma) \sin(\alpha) = \cos(\gamma - \alpha)$$

the above equation resolves to:

$$V_2^2 = L_1\dot{\theta}^2 + l_2^2(\dot{\theta} + \dot{\phi})^2 + 2L_1 l_2(\dot{\theta} + \dot{\phi})\dot{\theta} \cos(\phi)$$

The kinetic energy in the system consist of the fixed rotation of the under-actuated pendulum and the rotation and velocity of the actuated pendulum.

$$T = \frac{1}{2}I_A\dot{\theta}^2 + \frac{1}{2}I_B(\dot{\theta}+\dot{\phi})^2 + \frac{1}{2}m_2V_2^2$$

$$T = \frac{1}{2}I_A\dot{\theta}^2 + \frac{1}{2}I_B(\dot{\theta}+\dot{\phi})^2 + \frac{1}{2}m_2[L_1\dot{\theta}^2 + l_2^2(\dot{\theta}+\dot{\phi})^2 + 2L_1l_2(\dot{\theta}+\dot{\phi})\dot{\theta}\cos(\phi)]^2$$

The potential energy in the system is defined as

$$V = -m_1gl_1\cos(\theta) - m_2g[L_1\cos(\theta) + l_2\cos(\theta+\phi)]$$

The Lagrange is defined as

$$\mathcal{L} = T - V$$

$$\mathcal{L} = \frac{1}{2}I_A\dot{\theta}^2+\frac{1}{2}I_B(\dot{\theta}+\dot{\phi})^2+\frac{1}{2}m_2[L_1\dot{\theta}^2+l_2^2(\dot{\theta}+\dot{\phi})^2+2L_1l_2(\dot{\theta}+\dot{\phi})\dot{\theta}\cos(\phi)]^2+m_1gl_1\cos(\theta)+ m_2g[L_1\cos(\theta)+l_2\cos(\theta+\phi)]$$

$$\frac{\partial\mathcal{L}}{\partial\theta} = -m_1gl_1\sin(\theta) - m_2gL_2\sin(\theta) - m_2gl_2\sin(\theta+\phi)$$

$$\frac{d}{dt}\frac{\partial\mathcal{L}}{\partial\dot{\theta}} = I_A\ddot{\theta}+I_B\ddot{\theta}+I_B\ddot{\phi}+m_2L_1^2\ddot{\theta}+m_2l_2^2\ddot{\theta}+m_2l_2\ddot{\phi}+2m_2L_1l2\ddot{\theta}\cos(\phi)-2m_2L_1l_2\dot{\theta}\dot{\phi}\sin(\phi)+ m_2L_1l_2\ddot{\phi}\cos(\phi) - m_2L_1l_2\dot{\phi}^2\sin(\phi)$$

$$\frac{\partial\mathcal{L}}{\partial\phi} = -m_2L_1l_2(\dot{\theta}+\dot{\phi})\dot{\theta}\sin(\phi) - m_2gl_2\sin(\theta+\phi)$$

$$\frac{d}{dt}\frac{\partial\mathcal{L}}{\partial\dot{\phi}} = I_B\ddot{\theta} + I_B\ddot{\phi} + m_2l_2^2\ddot{\theta} + m_2l_2^2\ddot{\phi} + m_2L_1l_2\ddot{\theta}\cos(\phi) - m_2L_1l_2\dot{\theta}\dot{\phi}\sin(\phi)$$

The differential equation describing the dynamics of the system is

$$\frac{d}{dt}\frac{\partial\mathcal{L}}{\partial\vec{q}} - \frac{\partial\mathcal{L}}{\partial q} = B(\dot{q}) + \tau(q)$$

where $q = \begin{bmatrix}\theta\\ \phi\end{bmatrix}$

A.2 Collocated Linearisation

$$d_{11}\ddot{\theta} + d_{12}\ddot{\phi} + h_1 + \psi_1 = 0 \tag{A.1}$$

$$d_{21}\ddot{\theta} + d_{22}\ddot{\phi} + h_2 + \psi_2 = \tau \tag{A.2}$$

Starting from the condense equation (A.1) and (A.3), $\ddot{q_1}$ is solved in equation (A.1) and substituted in (A.1) resulting in

$$\bar{d_2}\ddot{\theta} + \bar{h_2} + \bar{\psi_2} = \tau \tag{A.3}$$

where the newly defined terms are given as:

$$\bar{d_2} = d_{22} - \frac{d_{21}d_{12}}{d_{11}}$$

$$\bar{h_2} = h_2 - \frac{d_{21}h_1}{d_{11}}$$

$$\bar{\psi_2} = \psi_2 - \frac{d_{21}\psi_1}{d_{11}}$$

τ can now chosen to linearise the terms in equation (A.3) as:
$\tau = \bar{d_2}v_2 + \bar{h_2} + \bar{\psi}$

A.3 Taylor Series Expansion Around Unstable Equilibrium Position

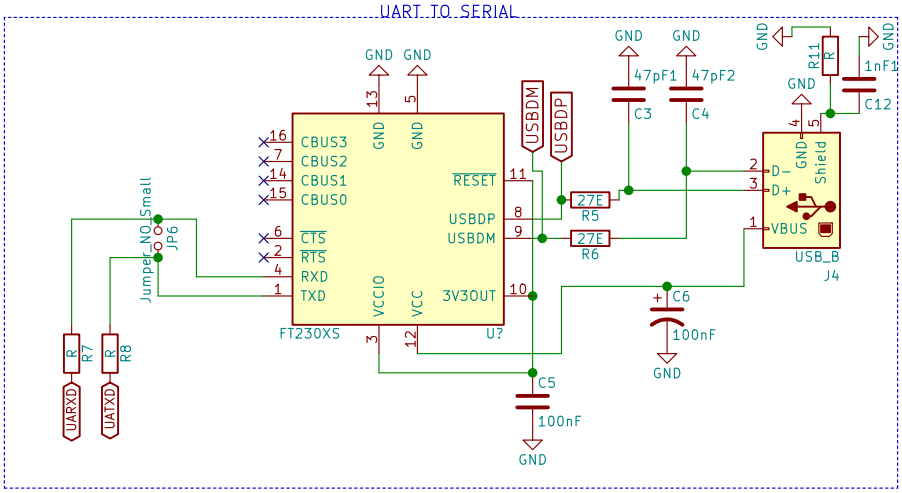
A.4 Communication Structure

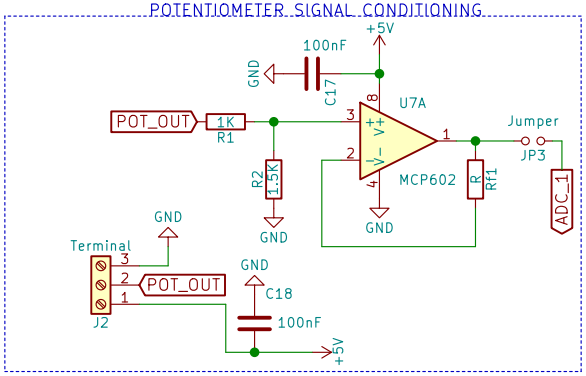
0	1	2	3	4	...	n-1	n
$	cmd	,	value	value		\r	\n

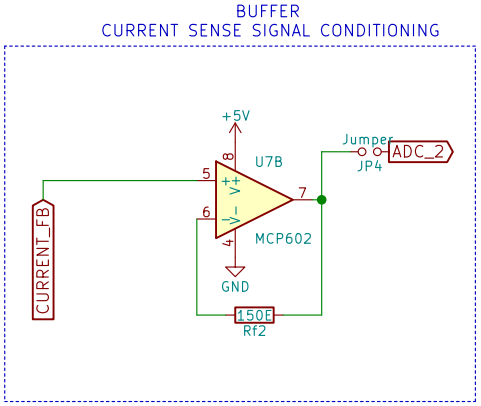
Figure A.1: Data Structure for Sending Commands

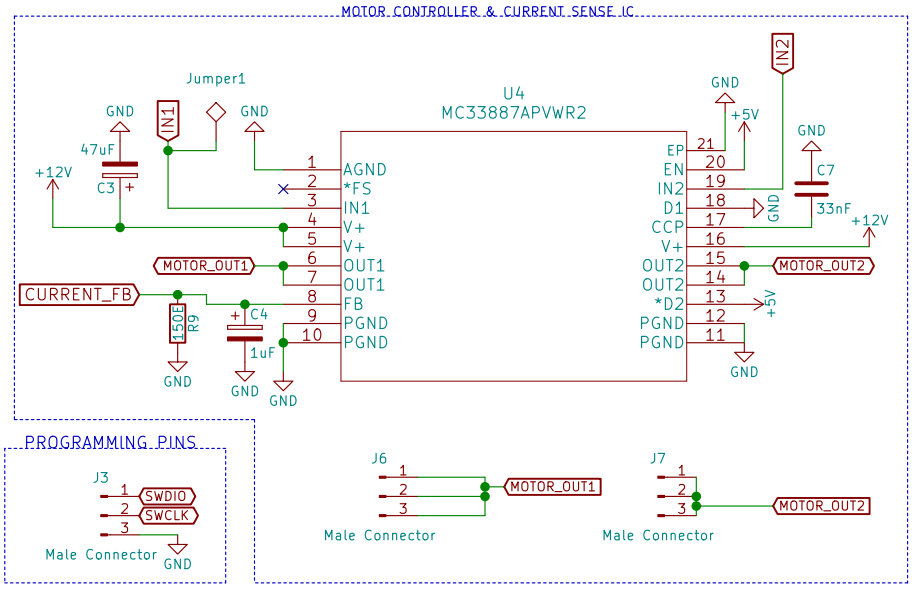
In Table A.1 the various commmands that are used in the command structure shown in Figure A.1 used for debugging purposes is explained with the possible value ranges that can be used.

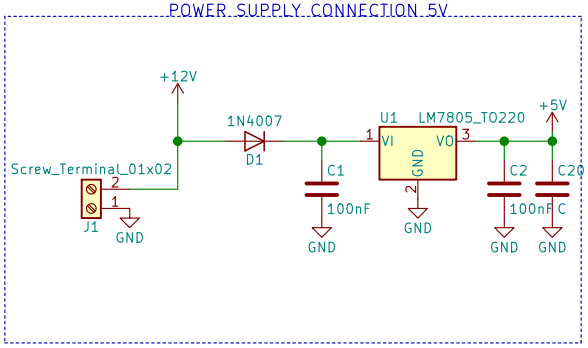
A.5 Electronic Design Schematic

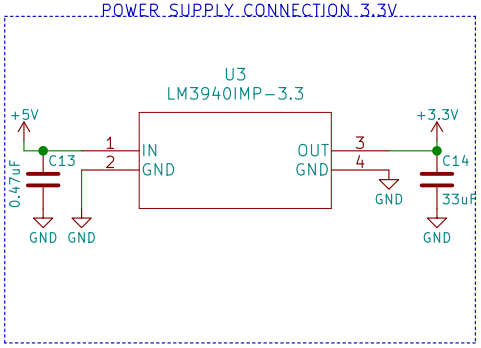




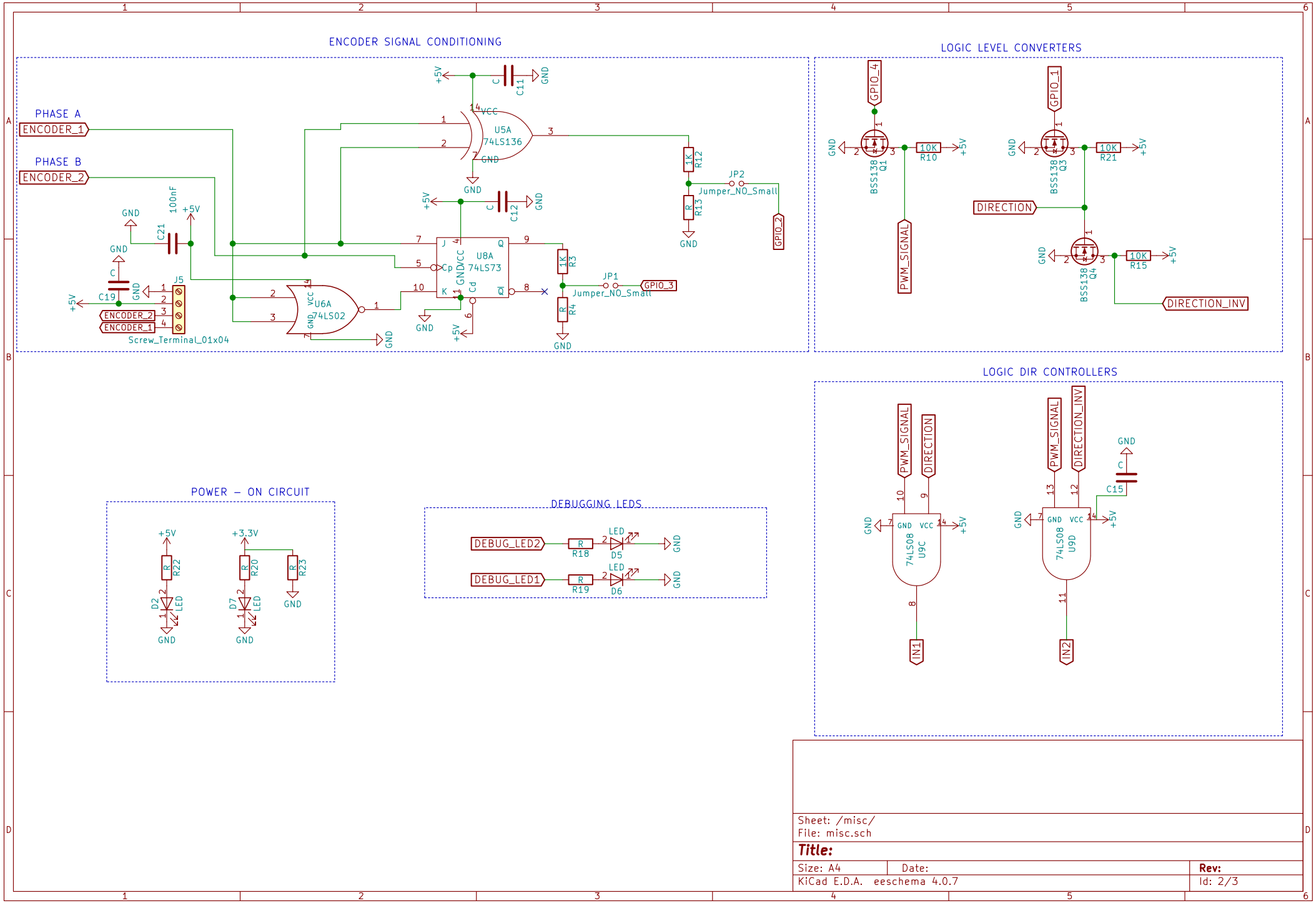


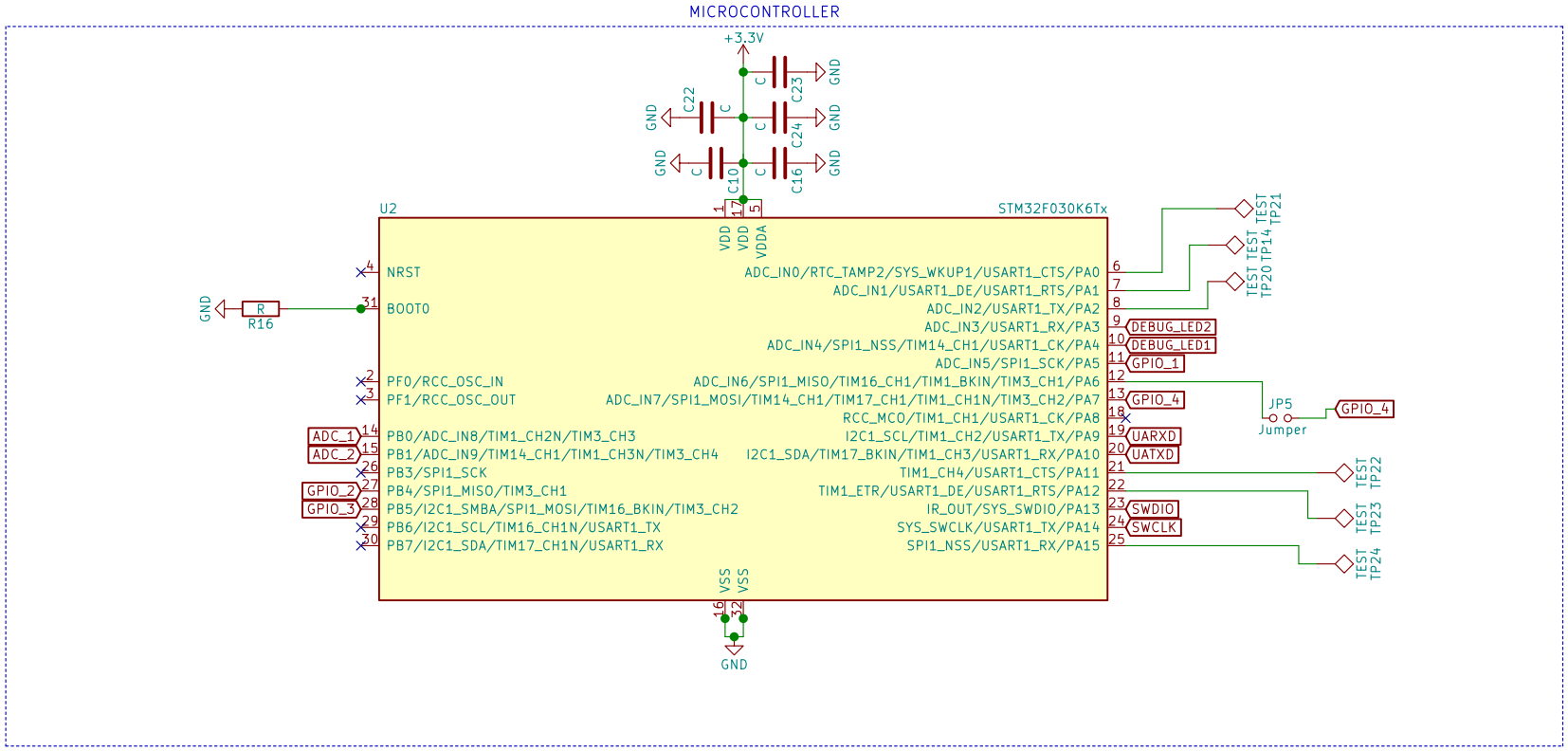






Sheet: /	
File: Acrobat.sch	
Title: Acrobat Schematic	
Size: A4	Date:
KiCad E.D.A. eeschema 4.0.7	Rev: Id: 1/3


ENCODER SIGNAL CONDITIONING
PHASE A
ENCODER_1
PHASE B
ENCODER_2
U5A
74LS136
U6A
74LS02
U8A
74LS73
Screw_Terminal_01x04
Jumper_NO_Small
Jumper_NO_Small
GPIO_3
GPIO_2
LOGIC LEVEL CONVERTERS
GPIO_4
GPIO_1
BSS138
BSS138
BSS138
DIRECTION
PWM_SIGNAL
DIRECTION_INV
LOGIC DIR CONTROLLERS
74LS08
74LS08
IN1
IN2
POWER – ON CIRCUIT
DEBUGGING LEDS
DEBUG_LED2
DEBUG_LED1
LED
LED
Sheet: /misc/
File: misc.sch
Title:
Size: A4
Date:
Rev:
KiCad E.D.A. eeschema 4.0.7
Id: 2/3



Sheet: /microcontroller/		
File: microcontroller.sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. eeschema 4.0.7		Id: 3/3

A.6 Techno-Economy Assessment

A.6.1 Budget

The comparison between the proposed and actual budget is discussed, why additional expenses occurred and where was savings possible.

A.6.2 Planning

A.6.3 Technical Impact

The impact of the results presented in this report on the field of control systems are discussed. The impact of this research on society and industry are discussed and whether the financial input was worthwhile.

A.6.4 Return on Investment

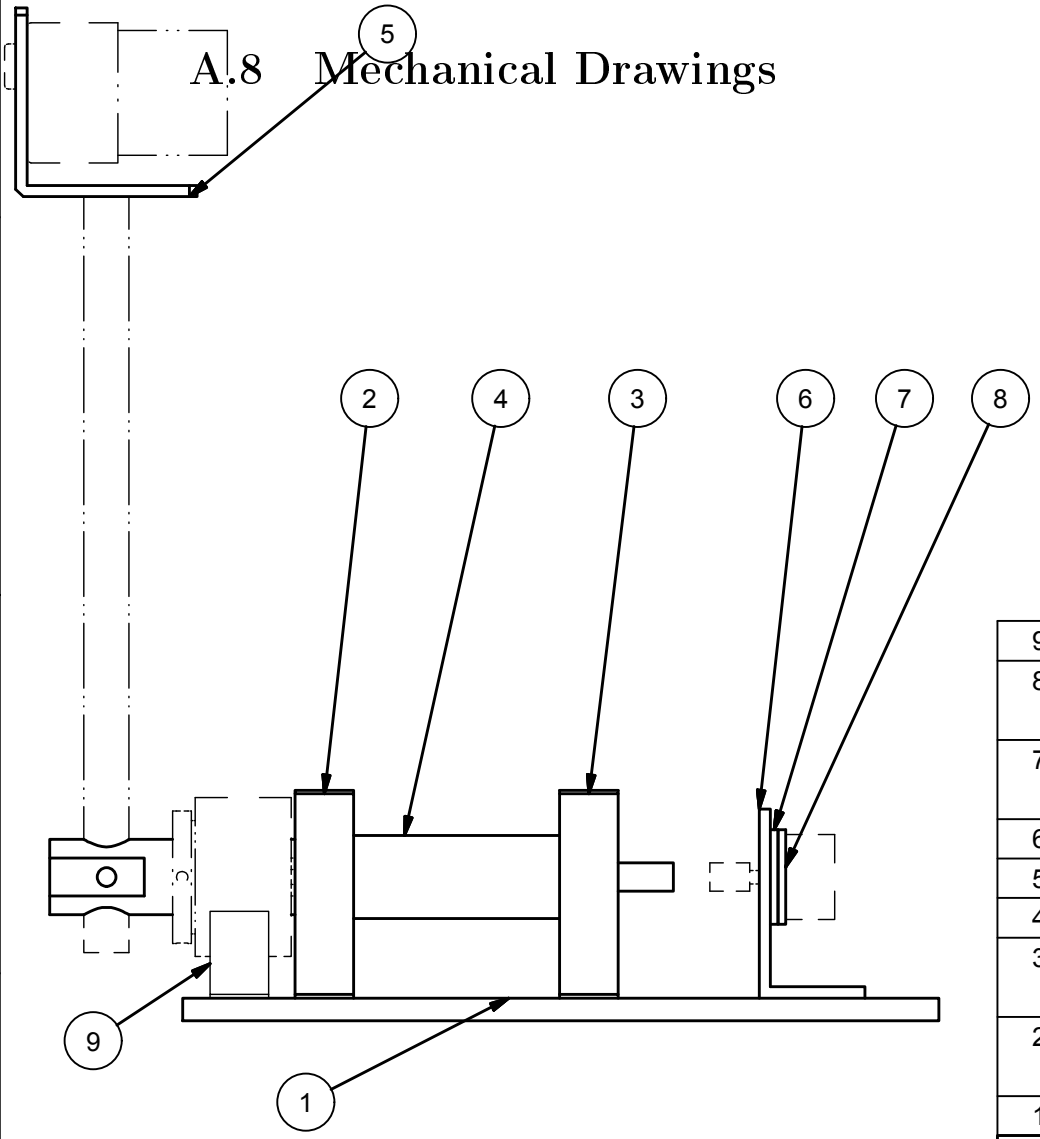
The short and long term value from a technical and economical perspective is provided in this section. A motivation on the continuing research in the field of underactuated robotics are given and the financial cost to further research in this field.

A.6.5 Potential for Commercialization

The potential for the commercialisation of the contents within the report is discussed and the value of this commercialisations is given.

A.7 Risk Analysis & Safety Procedures

A.8 Mechanical Drawings

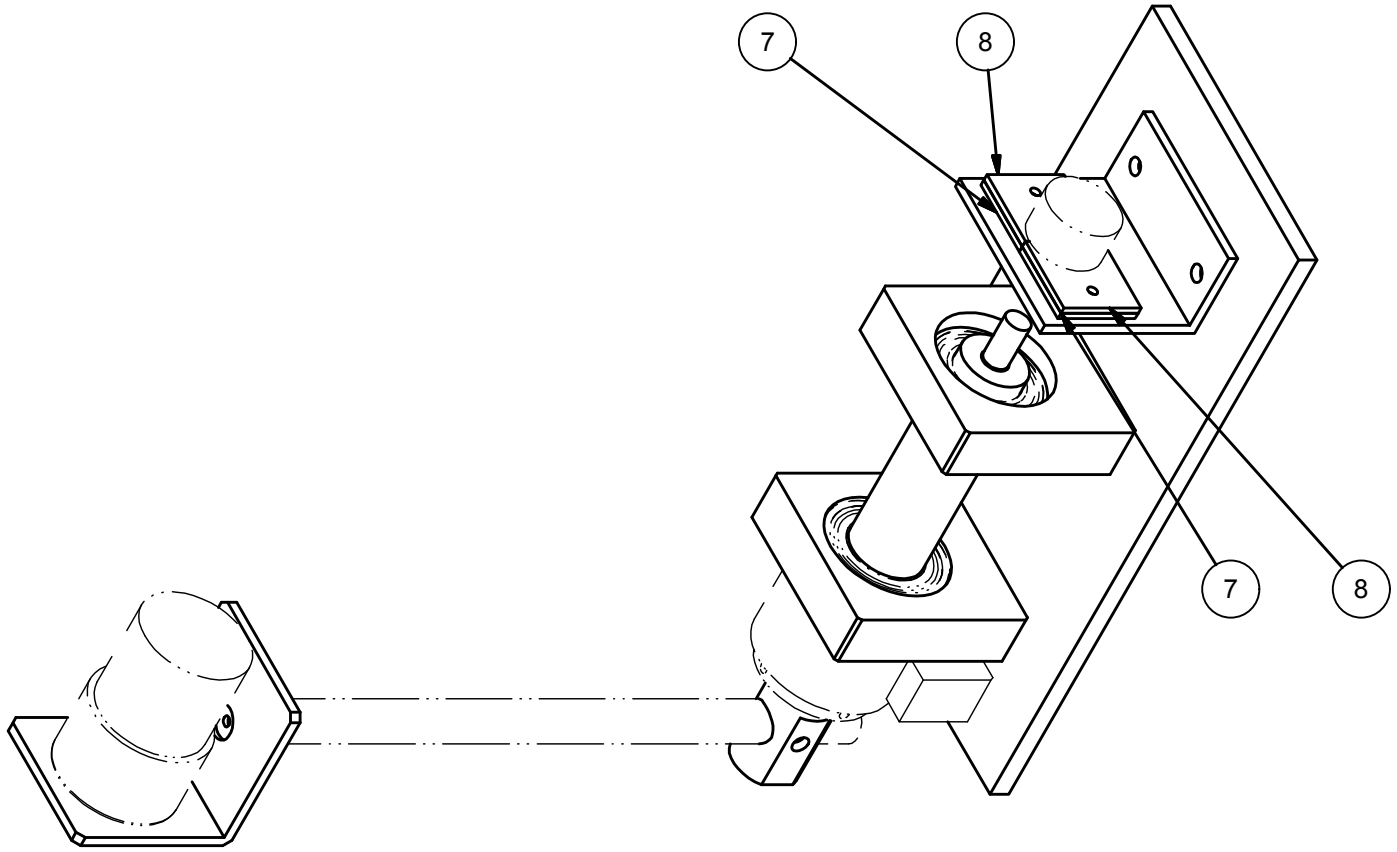


ITEM	BESKRYWING	AANTAL	MATERIAAL / SPESIFIKASIES
9	STANDOFFS	1	HK 09
8	SENSOR CONNECTION 2	2	HK 07
7	SENSOR CONNECTION 1	2	HK 06
6	SENSOR MOUNT	1	HK 04
5	MOTOR MOUNT	1	HK 08
4	SHAFT V3	1	HK 05
3	BOTTOM BEARING HOUSING	1	HK 03
2	UPPER BEARING HOUSING	1	HK 02
1	BASEPLATE	1	HK 01

UNIVERSITEIT VAN STELLENBOSCH

SKAAL OP A4 1:2
MATE IN mm
TITEL: FULL ASSEMBLY

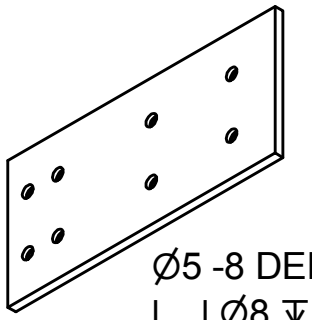
STUDENTE No. 19231865 | TEKENAAR H.KOTZé | NAGESIEN N/A | DATUM 6/24/2018 | VEL No. 1 VAN 2 VELLE | No. HK 00

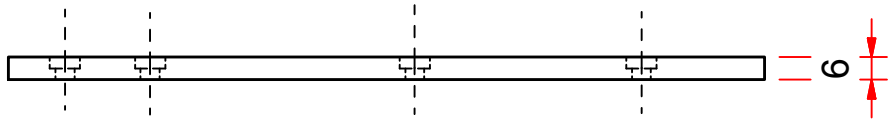


ITEM	BESKRYWING	AANTAL	MATERIAAL / SPESIFIKASIES

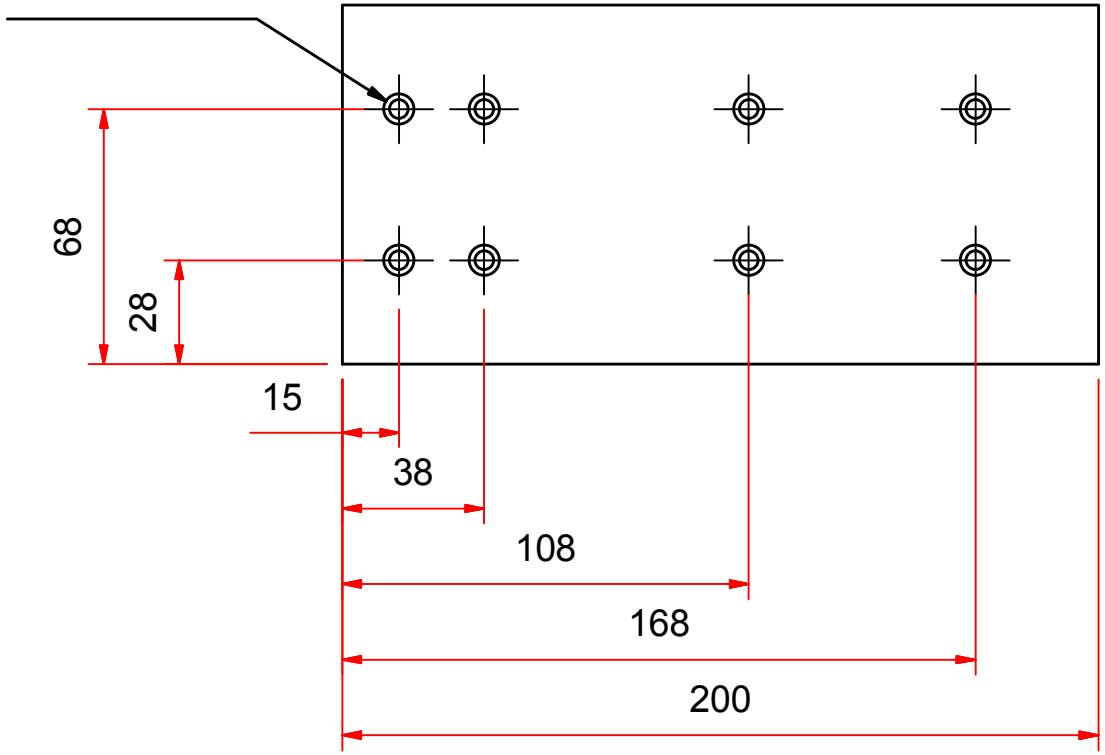
UNIVERSITEIT VAN STELLENBOSCH

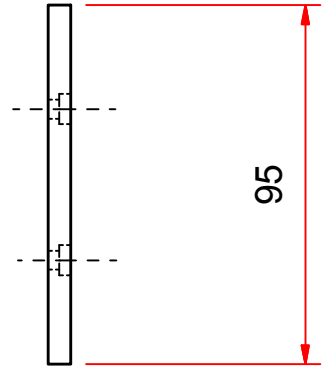
SKAAL OP A 4 1:2		TITEL: FULL ASSEMBLY			
MATE IN mm					
STUDENTE No. 19231865	TEKENAAR H.KOTZé	NAGESIEN N/A	DATUM 6/24/2018	VEL No.2 VAN 2 VELLE	No. HK 00





Ø5 -8 DEEP
⌴ Ø8 ↧ 3

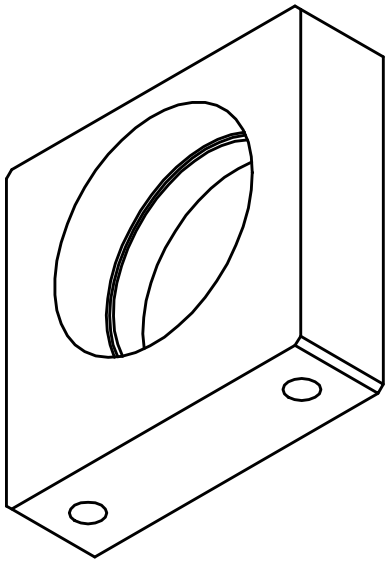


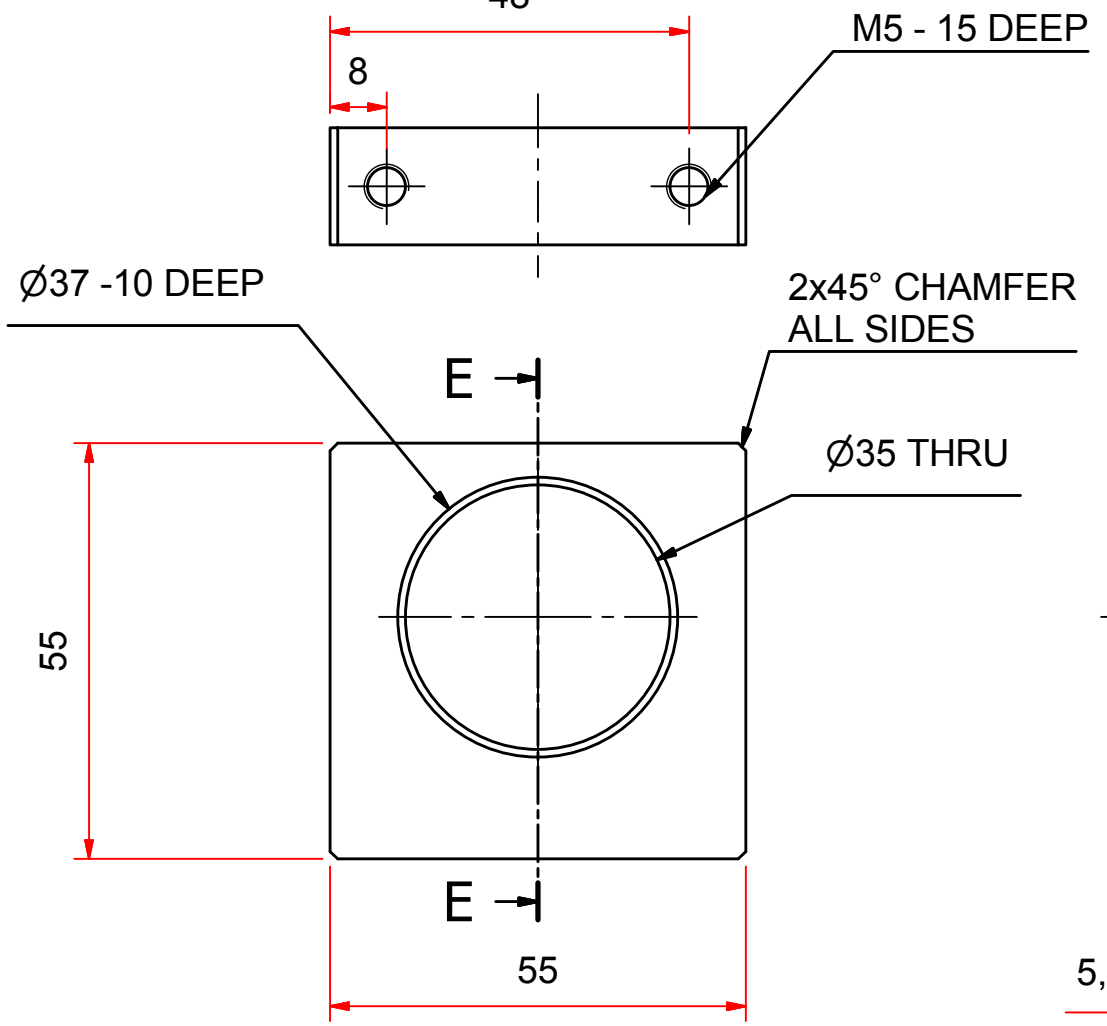


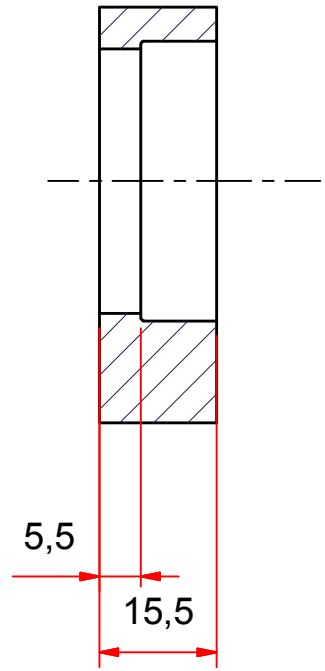
ITEM	BESKRYWING	AANTAL	MATERIAAL / SPESIFIKASIES
1	BASEPLATE	1	ALUMINIUM

UNIVERSITEIT VAN STELLENBOSCH

STUDENTE No. 19231865	TEKENAAR H. KOTZé	NAGESIEN N/A	SKAAL OP A4 1:2 MATE IN mm DATUM 6/23/2018	TITEL: BASEPLATE VEL No. 1 VAN 1 VELLE	No. HK 01







UNLESS OTHERWISE STATED
TOLERANCES ± 0,1
ANGLES 1°

ITEM	BESKRYWING	AANTAL	MATERIAAL / SPESIFIKASIES
1	BOTTOM BEARING HOUSING	1	ALUMINIUM

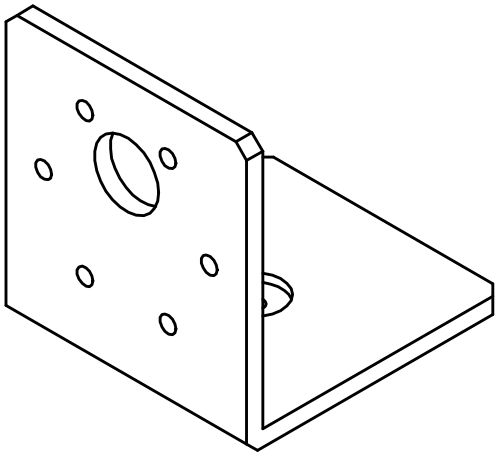
UNIVERSITEIT VAN STELLENBOSCH

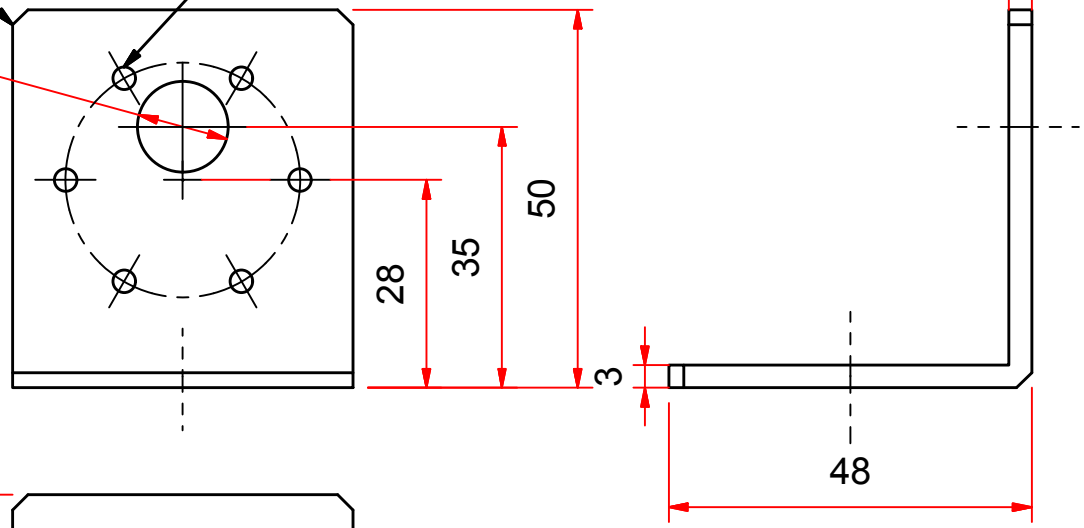
STUDENTE No. 19231865	TEKENAAR H. KOTZé	NAGESIEN N/A	SKAAL OP A4 1:1 MATE IN mm DATUM 6/23/2018	TITEL: BOTTOM BEARING HOUSING VEL No. 1 VAN 1 VELLE	No. HK 03

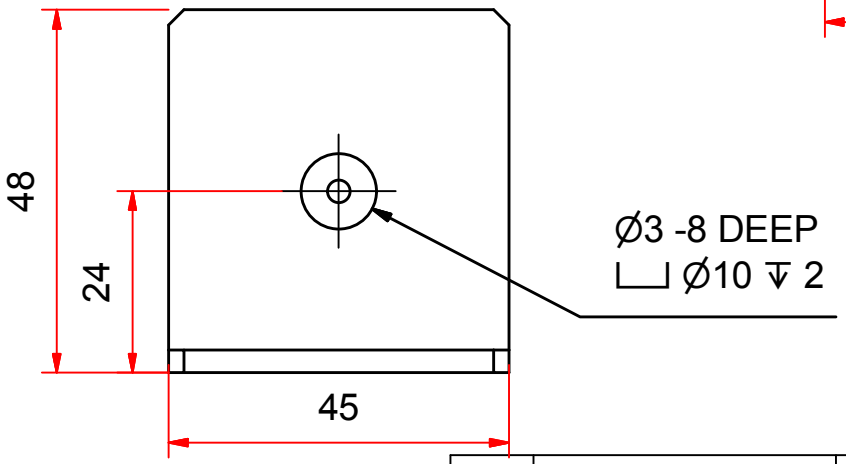
2x45° CHAMFER
ALL CORNERS

6 Ø3 HOLES
EQUI-SPACED ON PCD 31

Ø12 THRU







UNLESS OTHERWISE STATED
TOLERANCES ± 0,1
ANGLES 1°

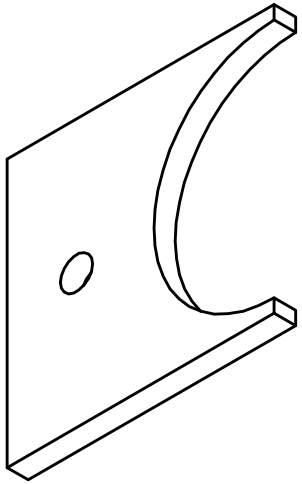
ITEM	BESKRYWING	AANTAL	MATERIAAL / SPESIFIKASIES
1	MOTOR MOUNT	1	ALUMINIUM

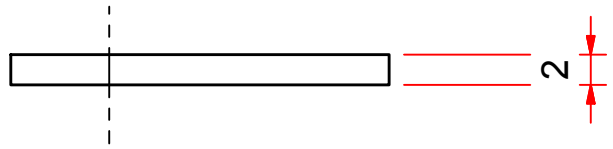
UNIVERSITEIT VAN STELLENBOSCH

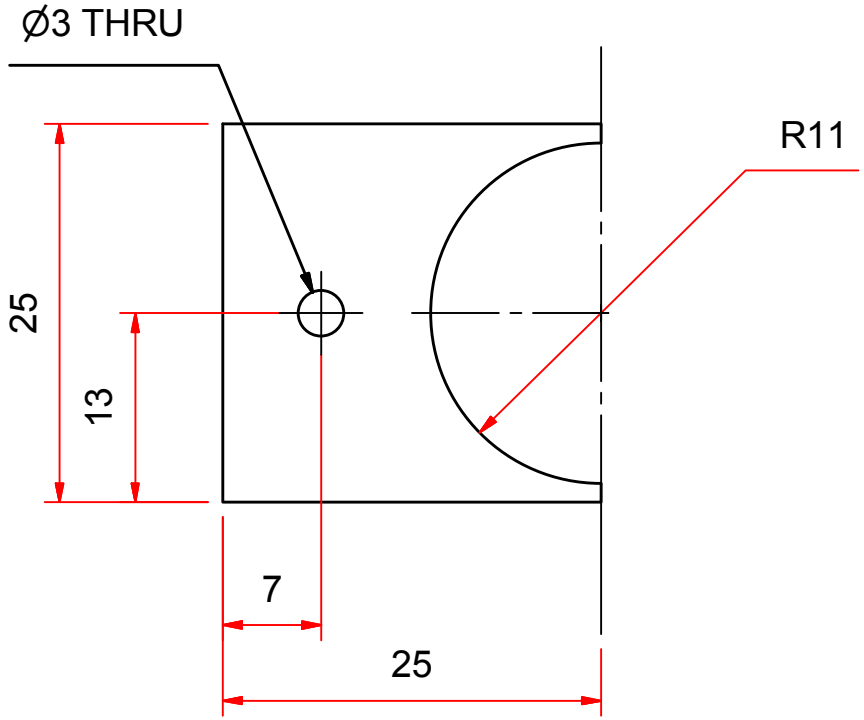
SKAAL OP A4 1:1
MATE IN mm

TITEL: MOTOR MOUNT

STUDENTE No.	TEKENAAR	NAGESIEN	DATUM	VEL No.	VAN	VELLE	No.
19231865	H KOTZé	N/A	6/23/2018	1	1		HK 08





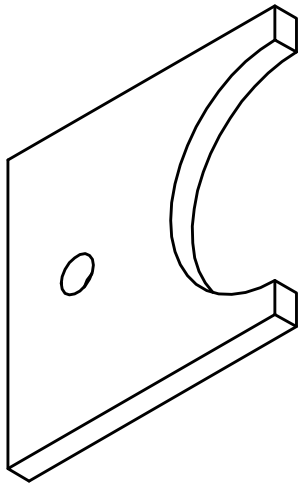


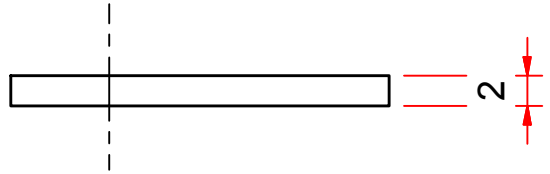
UNLESS OTHERWISE STATED
TOLERANCES ± 0,1
ANGLES 1°

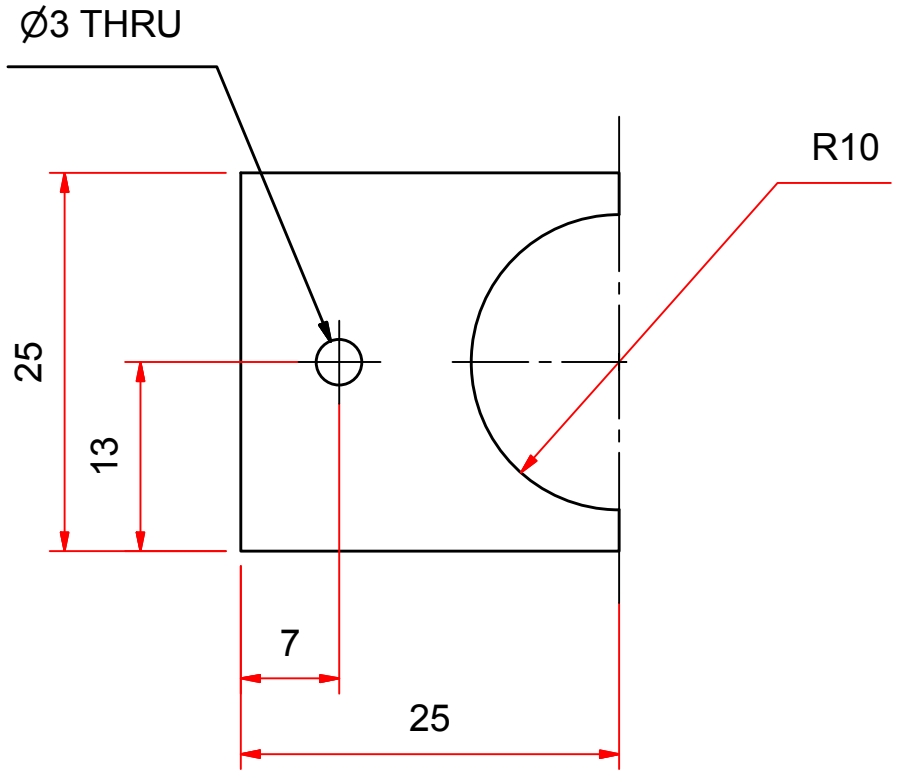
ITEM	BESKRYWING	AANTAL	MATERIAAL / SPESIFIKASIES
1	SENSOR CONNECTION 1	1	ALUMINIUM

UNIVERSITEIT VAN STELLENBOSCH

STUDENTE No. 19231865	**TEKENAAR** H.KOTZé	**NAGESIEN** N/A	**SKAAL OP A** 4 2:1 **MATE IN** mm **DATUM** 6/24/2018	**TITEL:** SENSOR CONNECTION 1 **VEL No.** 1 **VAN** 1 **VELLE**	**No.** HK 06





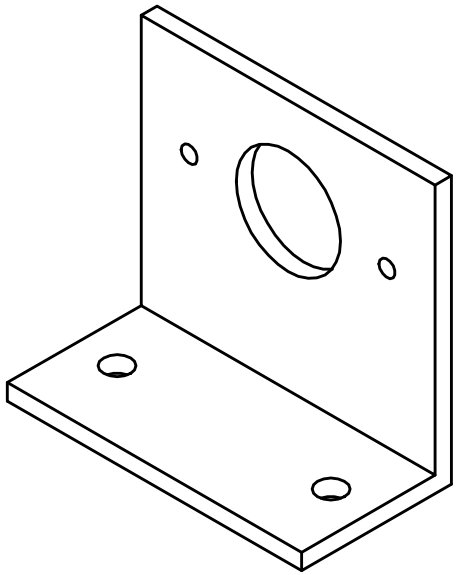


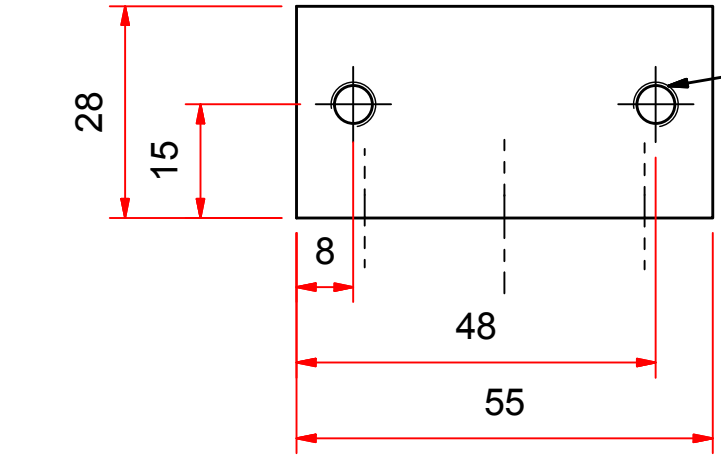
UNLESS OTHERWISE STATED
TOLERANCES ± 0,1
ANGLES 1˚

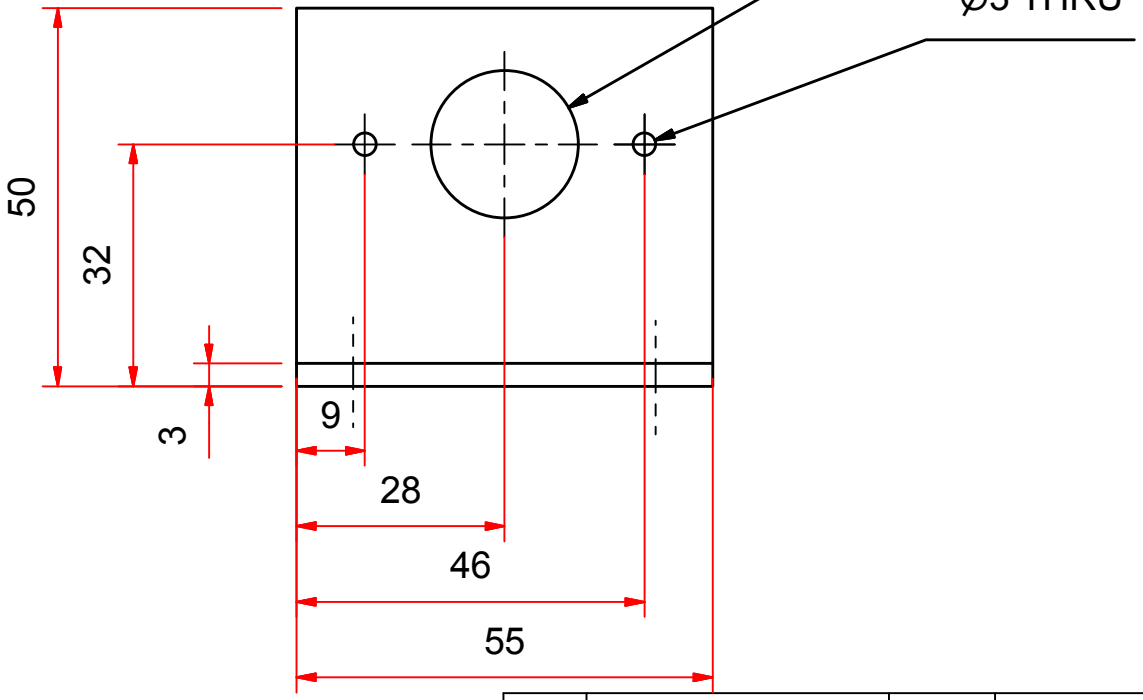
ITEM	BESKRYWING	AANTAL	MATERIAAL / SPESIFIKASIES
1	SENSOR CONNECTION 2	1	ALUMNIUM

UNIVERSITEIT VAN STELLENBOSCH

STUDENTE No. 19231865	**TEKENAAR** H.KOTZé	**NAGESIEN** N/A	**SKAAL OP A** 4 2:1 **MATE IN** mm **DATUM** 6/24/2018	**TITEL:** SENSOR CONNECTION 2 **VEL No.** 1 **VAN** 1 **VELLE**	**No.** HK 07





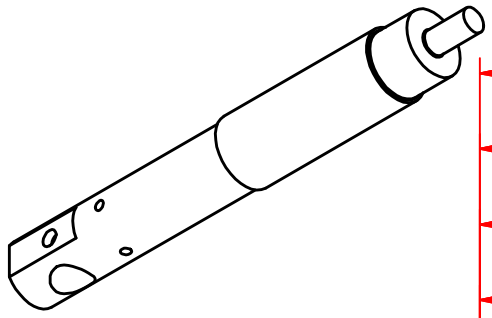


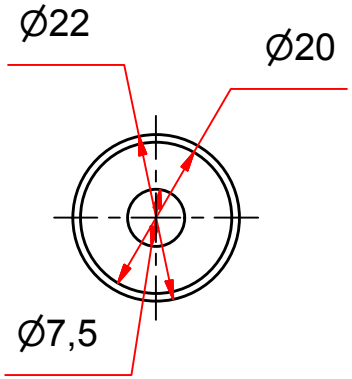
UNLESS OTHERWISE STATED
TOLERANCES ± 0,1
ANGLES 1°

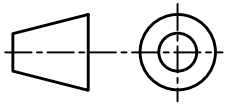
ITEM	BESKRYWING	AANTAL	MATERIAAL / SPESIFIKASIES
1	SENSOR MOUNT	1	ALUMINIUM

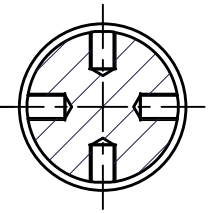
UNIVERSITEIT VAN STELLENBOSCH

STUDENTE No. 19231865	TEKENAAR H. KOTZé	NAGESIEN N/A	SKAAL OP A4 1:1 MATE IN mm DATUM 6/24/2018	TITEL: SENSOR MOUNT VEL No. 1 VAN 1 VELLE	No. HK 04







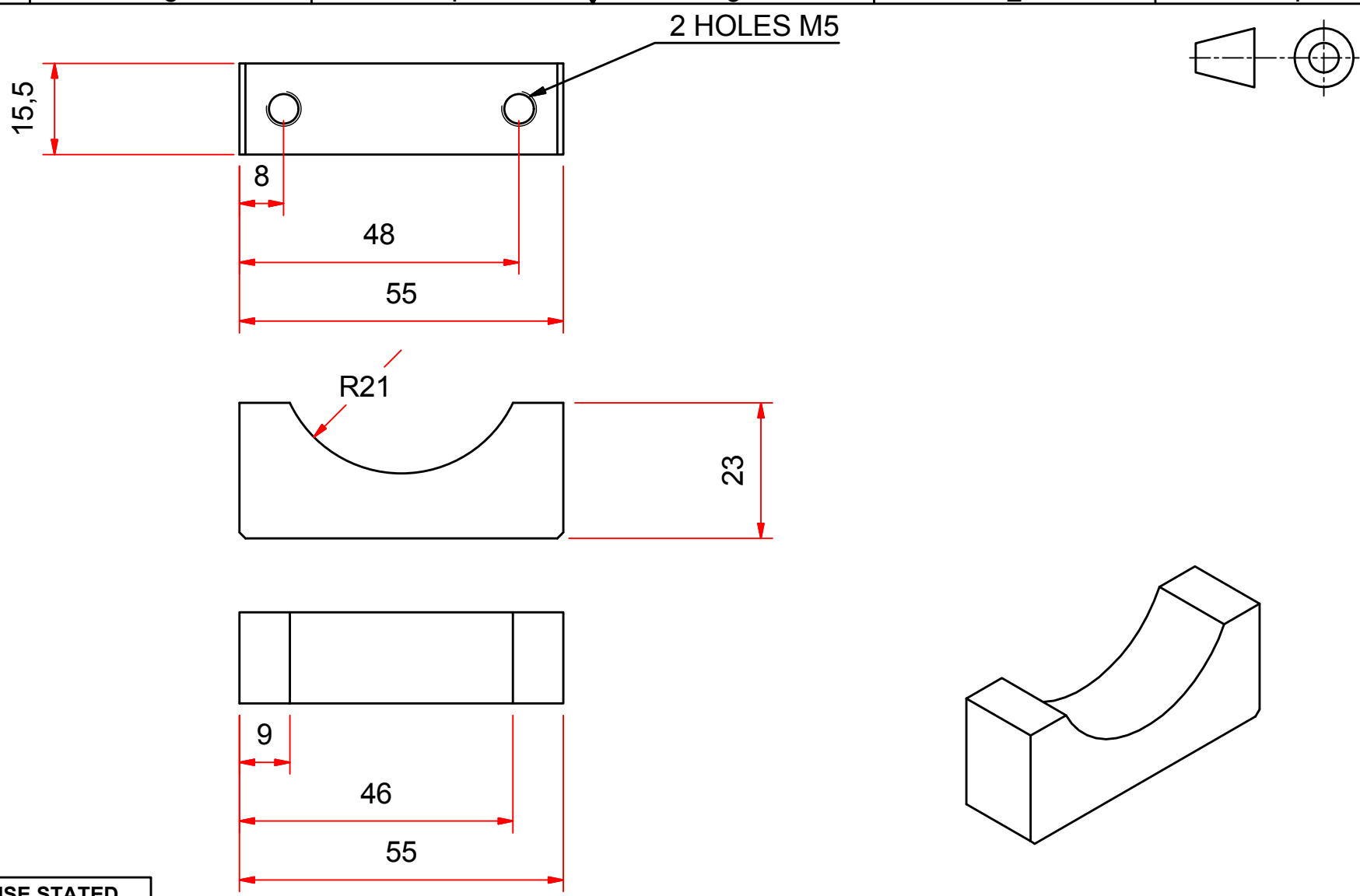


UNLESS OTHERWISE STATED
TOLERANCES ± 0,1
ANGLES 1°

ITEM	BESKRYWING	AANTAL	MATERIAAL / SPESIFIKASIES
1	SHAFT V3	1	ALUMINIUM

UNIVERSITEIT VAN STELLENBOSCH

STUDENTE No. 19231865	TEKENAAR H. KOTZé	NAGESIEN N/A	SKAAL OP A4 1:1 MATE IN mm DATUM 6/24/2018	TITEL: SHAFT V3 VEL No. 1 VAN 1 VELLE	No. HK 05



UNLESS OTHERWISE STATED
TOLERANCES ± 0,1
ANGLES 1°

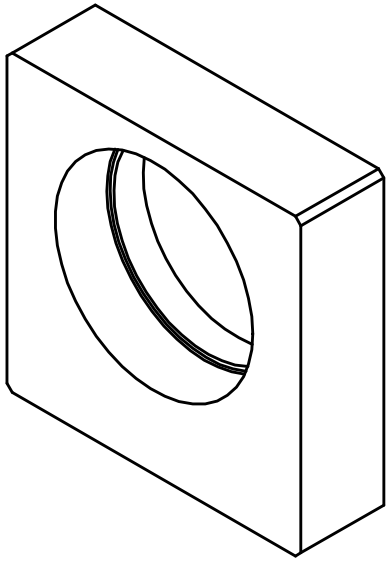
ITEM	BESKRYWING	AANTAL	MATERIAAL / SPESIFIKASIES
1	STANDOFFS	1	ALUMINIUM

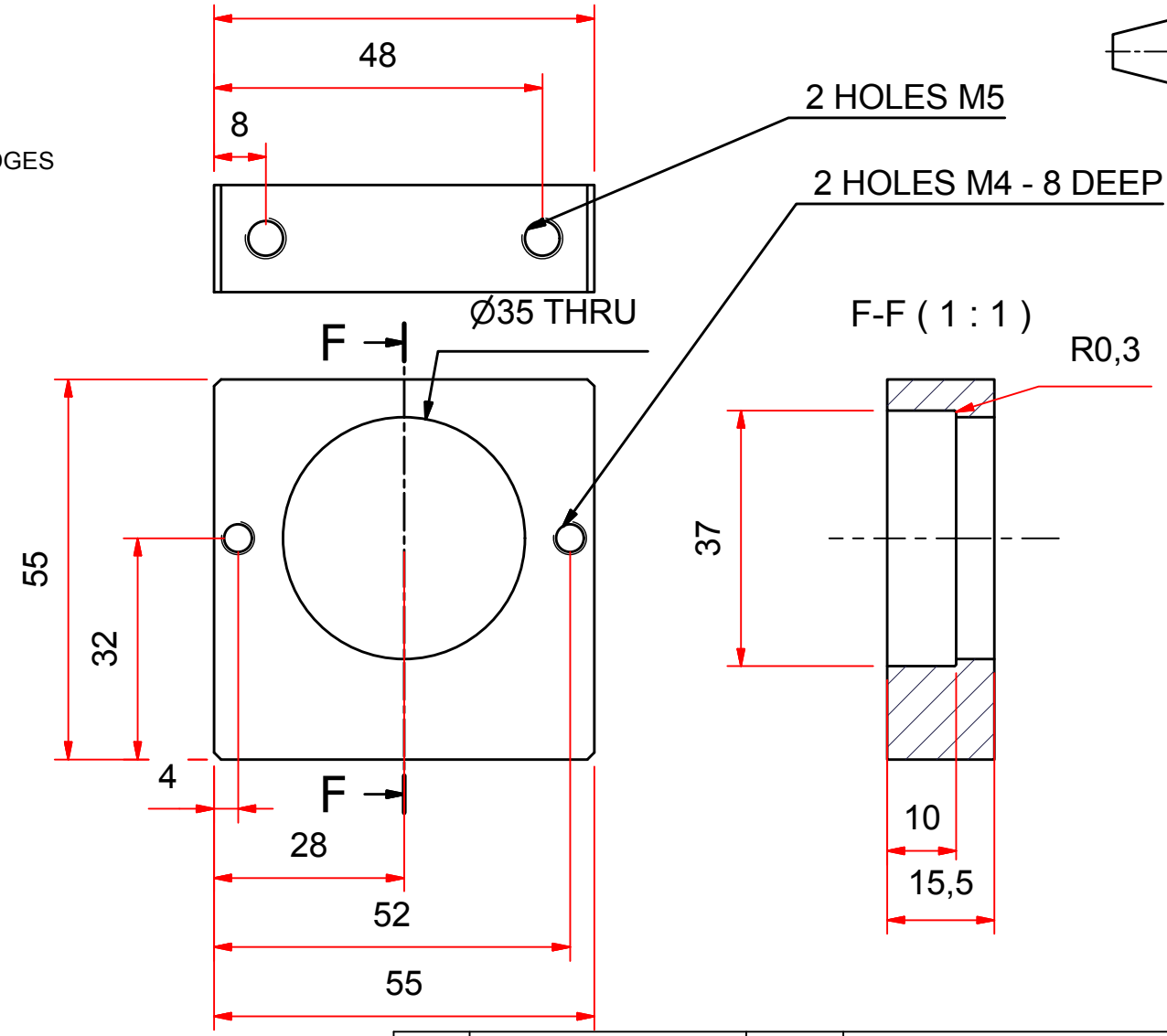
UNIVERSITEIT VAN STELLENBOSCH

STUDENTE No. 19231865	TEKENAAR H. KOTZé	NAGESIEN N/A	DATUM 6/23/2018	VEL No. 1 VAN 1 VELLE	No. HK 09

SKAAL OP A4 1:1
MATE IN mm
TITEL: STANDOFFS

NOTE: REMOVE ALL SHARP EDGES





UNLESS OTHERWISE STATED
TOLERANCES ± 0,1
ANGLES 1°

ITEM	BESKRYWING	AANTAL	MATERIAAL / SPESIFIKASIES
1	UPPER BEARING HOUSING	1	ALUMINIUM

UNIVERSITEIT VAN STELLENBOSCH

STUDENTE No. 19231865	**TEKENAAR** H. KOTZé	**NAGESIEN** N/A	**SKAAL OP A4** 1:1 **MATE IN** mm **DATUM** 6/23/2018	**TITEL:** UPPER BEARING HOUSING **VEL No.** 1 **VAN** 1 **VELLE**	**No.** HK 02

A.9 Microcontroller Settings

1. Description

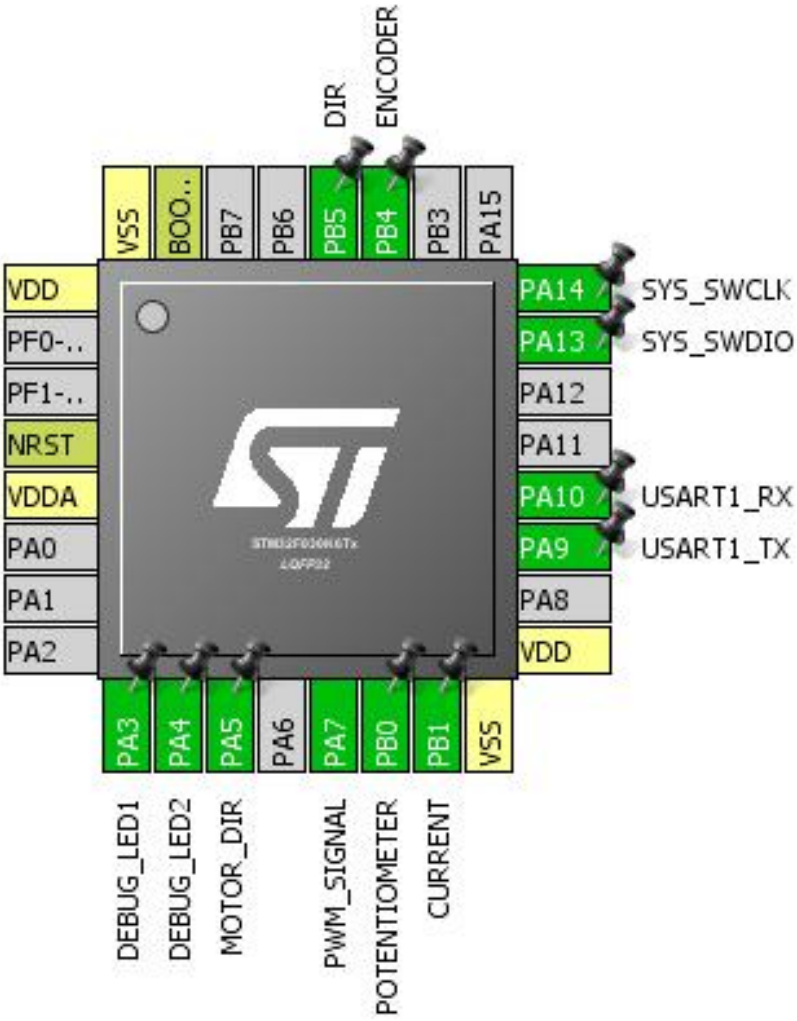
1.1. Project

Project Name	acrobat_v4
Board Name	acrobat_v4
Generated with:	STM32CubeMX 4.25.0
Date	10/18/2018

1.2. MCU

MCU Series	STM32F0
MCU Line	STM32F0x0 Value Line
MCU name	STM32F030K6Tx
MCU Package	LQFP32
MCU Pin number	32

2. Pinout Configuration

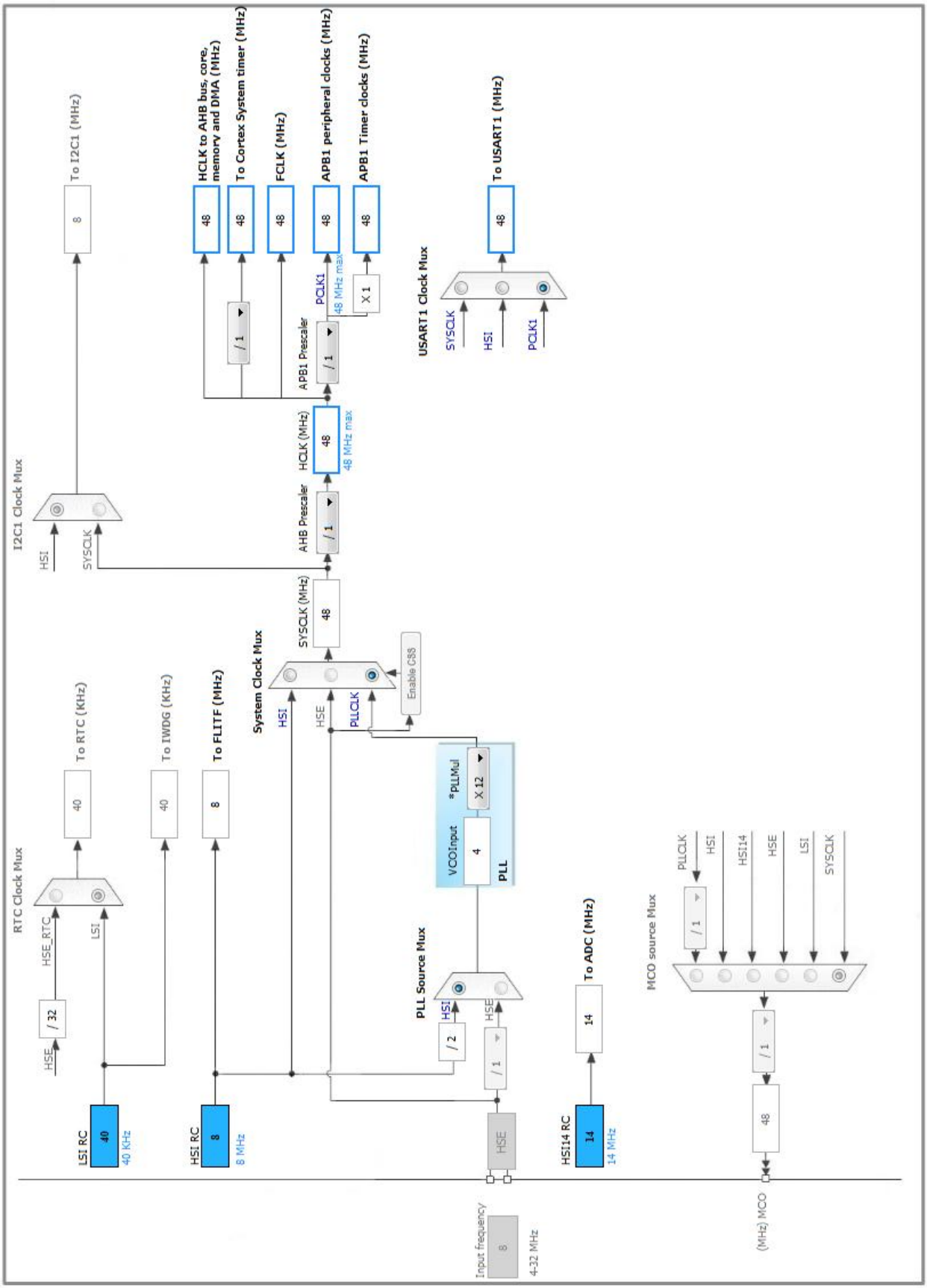


3. Pins Configuration

Pin Number LQFP32	Pin Name (function after reset)	Pin Type	Alternate Function(s)	Label
1	VDD	Power		
4	NRST	Reset		
5	VDDA	Power		
9	PA3 *	I/O	GPIO_Output	DEBUG_LED1
10	PA4 *	I/O	GPIO_Output	DEBUG_LED2
11	PA5 *	I/O	GPIO_Output	MOTOR_DIR
13	PA7	I/O	TIM3_CH2	PWM_SIGNAL
14	PB0	I/O	ADC_IN8	POTENTIOMETER
15	PB1	I/O	ADC_IN9	CURRENT
16	VSS	Power		
17	VDD	Power		
19	PA9	I/O	USART1_TX	
20	PA10	I/O	USART1_RX	
23	PA13	I/O	SYS_SWDIO	
24	PA14	I/O	SYS_SWCLK	
27	PB4	I/O	GPIO_EXTI4	ENCODER
28	PB5 *	I/O	GPIO_Input	DIR
31	BOOT0	Boot		
32	VSS	Power		

* The pin is affected with an I/O function

4. Clock Tree Configuration



5. IPs and Middleware Configuration

5.1. ADC

mode: IN8

mode: IN9

5.1.1. Parameter Settings:

ADC_Settings:	
Clock Prescaler	Asynchronous clock mode
Resolution	ADC 12-bit resolution
Data Alignment	Right alignment
Scan Conversion Mode	Forward
Continuous Conversion Mode	**Enabled ***
Discontinuous Conversion Mode	Disabled
DMA Continuous Requests	**Enabled ***
End Of Conversion Selection	End of single conversion
Overrun behaviour	Overrun data preserved
Low Power Auto Wait	Disabled
Low Power Auto Power Off	Disabled
ADC_Regular_ConversionMode:	
Sampling Time	**239.5 Cycles ***
External Trigger Conversion Source	Regular Conversion launched by software
External Trigger Conversion Edge	None
WatchDog:	
Enable Analog WatchDog Mode	false

5.2. SYS

mode: Debug Serial Wire

Timebase Source: SysTick

5.3. TIM3

Channel2: PWM Generation CH2

5.3.1. Parameter Settings:

Counter Settings:

Prescaler (PSC - 16 bits value)	**48 ***
Counter Mode	Up
Counter Period (AutoReload Register - 16 bits value)	**99 ***
Internal Clock Division (CKD)	No Division
auto-reload preload	Disable

Trigger Output (TRGO) Parameters:

Master/Slave Mode (MSM bit)	Disable (Trigger input effect not delayed)
Trigger Event Selection	Reset (UG bit from TIMx_EGR)

PWM Generation Channel 2:

Mode	PWM mode 1
Pulse (16 bits value)	0
Fast Mode	Disable
CH Polarity	High

5.4. TIM14

mode: Activated

5.4.1. Parameter Settings:

Counter Settings:

Prescaler (PSC - 16 bits value)	**192 ***
Counter Mode	Up
Counter Period (AutoReload Register - 16 bits value)	**2000 ***
Internal Clock Division (CKD)	No Division
auto-reload preload	Disable

5.5. TIM16

mode: Activated

5.5.1. Parameter Settings:

Counter Settings:

Prescaler (PSC - 16 bits value)	

	384 *
Counter Mode	Up
Counter Period (AutoReload Register - 16 bits value)	**2000 ***
Internal Clock Division (CKD)	No Division
Repetition Counter (RCR - 8 bits value)	0
auto-reload preload	Disable

5.6. USART1

Mode: Asynchronous

5.6.1. Parameter Settings:

Basic Parameters:

Baud Rate	**230400 ***
Word Length	8 Bits (including Parity)
Parity	None
Stop Bits	1

Advanced Parameters:

Data Direction	Receive and Transmit
Over Sampling	16 Samples
Single Sample	Disable

Advanced Features:

Auto Baudrate	Disable
TX Pin Active Level Inversion	Disable
RX Pin Active Level Inversion	Disable
Data Inversion	Disable
TX and RX Pins Swapping	Disable
Overrun	Enable
DMA on RX Error	Enable
MSB First	Disable

*** User modified value**

6. System Configuration

6.1. GPIO configuration

IP	Pin	Signal	GPIO mode	GPIO pull/up pull down	Max Speed	User Label
ADC	PB0	ADC_IN8	Analog mode	No pull-up and no pull-down	n/a	POTENTIOMETER
	PB1	ADC_IN9	Analog mode	No pull-up and no pull-down	n/a	CURRENT
SYS	PA13	SYS_SWDIO	n/a	n/a	n/a	
	PA14	SYS_SWCLK	n/a	n/a	n/a	
TIM3	PA7	TIM3_CH2	Alternate Function Push Pull	No pull-up and no pull-down	Low	PWM_SIGNAL
USART1	PA9	USART1_TX	Alternate Function Push Pull	No pull-up and no pull-down	**High ***	
	PA10	USART1_RX	Alternate Function Push Pull	No pull-up and no pull-down	**High ***	
GPIO	PA3	GPIO_Output	Output Push Pull	No pull-up and no pull-down	Low	DEBUG_LED1
	PA4	GPIO_Output	Output Push Pull	No pull-up and no pull-down	Low	DEBUG_LED2
	PA5	GPIO_Output	Output Push Pull	No pull-up and no pull-down	Low	MOTOR_DIR
	PB4	GPIO_EXTI4	**External Interrupt Mode with Rising/Falling edge**	No pull-up and no pull-down	n/a	ENCODER
	PB5	GPIO_Input	Input mode	No pull-up and no pull-down	n/a	DIR

6.2. DMA configuration

DMA request	Stream	Direction	Priority
USART1_TX	DMA1_Channel2	Memory To Peripheral	**High ***
ADC	DMA1_Channel1	Peripheral To Memory	**High ***

USART1_TX: DMA1_Channel2 DMA request Settings:

Mode:	Normal
Peripheral Increment:	Disable
Memory Increment:	**Enable ***
Peripheral Data Width:	Byte
Memory Data Width:	Byte

ADC: DMA1_Channel1 DMA request Settings:

Mode:	**Circular ***
Peripheral Increment:	Disable
Memory Increment:	**Enable ***
Peripheral Data Width:	**Word ***
Memory Data Width:	**Word ***

6.3. NVIC configuration

Interrupt Table	Enable	Preenmption Priority	SubPriority
Non maskable interrupt	true	0	0
Hard fault interrupt	true	0	0
System service call via SWI instruction	true	0	0
Pendable request for system service	true	0	0
System tick timer	true	0	0
EXTI line 4 to 15 interrupts	true	0	0
DMA1 channel 1 interrupt	true	1	0
DMA1 channel 2 and 3 interrupts	true	1	0
TIM3 global interrupt	true	1	0
TIM14 global interrupt	true	1	0
TIM16 global interrupt	true	0	0
USART1 global interrupt	true	0	0
Flash global interrupt	unused		
RCC global interrupt	unused		
ADC interrupt	unused		

*** User modified value**

7. Power Consumption Calculator report

7.1. Microcontroller Selection

Series	STM32F0
Line	STM32F0x0 Value Line
MCU	STM32F030K6Tx
Datasheet	024849_Rev2

7.2. Parameter Selection

Temperature	25
Vdd	3.6

8. Software Project

8.1. Project Settings

Name	Value
Project Name	acrobat_v4
Project Folder	C:\Users\Henry\Desktop\Skripsie\Feedback-Control-of-Robotic-Gymnast-
Toolchain / IDE	TrueSTUDIO
Firmware Package Name and Version	STM32Cube FW_F0 V1.9.0

8.2. Code Generation Settings

Name	Value
STM32Cube Firmware Library Package	Copy only the necessary library files
Generate peripheral initialization as a pair of '.c/.h' files	No
Backup previously generated files when re-generating	No
Delete previously generated files when not re-generated	Yes
Set all free pins as analog (to optimize the power consumption)	No

9. Software Pack Report

Command	Range	Reason for Implementation	
A	None	Testing of the UART circuit	Send the fol
B	0-100		
C	0-1	Testing of AND digital Circuit Directional Control of Motor	
D			

Table A.1: Summary of Communication Commands and their Effects

List of References

(2007 2). *5.0 A H-Bridge woth Load Current Feeback.* Freescale Semiconductor, 12th edn. Document Number: MC33887.

(2017). *How to get the best ADC accuracy in the STM32 microcontrollers.* ST Electronics.

Brown, S.C. and Passino, K.M. (1997). Intelligent Control for an Acrobot. *Journal of Intelligent and Robotic Systems*, vol. 18, no. 3, pp. 209–248. ISSN 09210296. Available at: `http://link.springer.com/10.1023/A:1007953809856`

Faulhaber (2011). *Encoders, Magnetic Encoder, Series IE2-16.* Faulhaber, 11th edn.

Gene F. Franklin, J. David Powell, e.a. (2015). *Feedback Control of Dynamic Systems.* 7th edn. Pearson Education Limited, Edingburgh Gate, Harlow, Essex CM20 2JE, England. ISBN 1-29-206890-6.

Inman, D.J. (2015). *Engineering Vibration.* Pearson Education Limited.

Instruments, N. (2006 9). *Magnetic Encoder Fundamentals.* 10. National Instruments, The address.

Lehl, M. (2012 10). *Control Of Planar Pendulum Systems.* Ph.D. thesis, Technische Universität München.

Murray, R.M. and Hauser, J. (1991). A Case Study in Approximate Linearization: The Acrobot Example. Tech. Rep..
Available at: `http://www.cds.caltech.edu/ murray/preprints/erl-M91-46.pdf`

Spong, M.W. (1995). The Swing Up Control Problem For The Acrobot. *IEEE Control Systems.* ISSN 1066033X.

Tedrake, R. (2009). Underactuated robotics: Algorithms for walking, running, swimming, flying, and manipulation (course notes for mit 6.832).
Available at: `http://underactuated.mit.edu/`