DEVELOPMENT OF A STAIR-CLIMBING ROBOT AND A HYBRID STABILIZATION SYSTEM FOR SELF-BALANCING ROBOTS

DÉVELOPPEMENT D'UN ROBOT MONTANT DES ESCALIERS ET D'UN SYSTÈME DE STABILISATION HYBRIDE POUR DES ROBOTS S'AUTO-BALANÇANT

A Thesis Submitted to

the Faculty of Graduate and Postdoctoral Studies

by

Dominic Robillard

In Partial Fulfillment of the Requirements for the Degree of

MASTERS OF APPLIED SCIENCE

in Mechanical Engineering

Ottawa-Carleton Institute for Mechanical and Aerospace Engineering

University of Ottawa

© Dominic Robillard, Ottawa, Canada, 2014

Acknowledgements

I would like to thank Dr. Atef Fahim, my former supervisor who is unfortunately not with us anymore. The accomplishment of this work could not have happened without his vision, support, wisdom, and sense of humour.

I would also like to give special thanks to Dr. Eric Lanteigne, my current supervisor who was an invaluable help with guiding me in the right direction, made the process enjoyable and encouraged me to pursue the project to its completion.

Last but certainly not least a warm thank you to my family, girlfriend, and friends whose amazing support and ability to cope with my difficulties have been vital in giving me the energy to complete my thesis.

Preface

This dissertation is an original intellectual product of the author, D. Robillard. All the theory and results covered in this thesis, unless otherwise cited, is original work. Development of the experimental robot and the hybrid inverted pendulum apparatus are also custom original work.

An article written by the author has been published on the stair-climbing robot at the *CCToMM Mechanisms, Machines, and Mechatronics (M3) Symposium, 2013*. A reproduction of the article is included in Appendix I.

Abstract

Robillard, Dominic. M.A.Sc. University of Ottawa, 2014. *Development of a stair-climbing robot and a hybrid stabilization system for self-balancing robots.* Supervised by Dr. E. Lanteigne, Ph.D.

Self-balancing robots are unique mobile platforms that stay upright on two wheels using a closed-loop control system. They can turn on the spot using differential steering and have compact form factors that limit their required floor space. However they have major limitations keeping them from being used in real world applications: they cannot stand-up on their own, climb stairs, or overcome obstacles. They can fall easily if hit or going onto a slippery surface because they rely on friction for balancing.

The first part of this research proposes a novel design to address the above mentioned issues related to stair-climbing, standing-up, and obstacles. A single revolute joint is added to the centre of a four-wheel drive robot onto which an arm is attached, allowing the robot to successfully climb stairs and stand-up on its own from a single motion. A model and simulation of the balancing and stair-climbing process are derived, and compared against experimental results with a custom robot prototype.

The second part, a control system for an inverted pendulum equipped with a gyroscopic mechanism, was investigated for integration into self-balancing robots. It improves disturbance rejection during balance, and keeps equilibrium on slippery surfaces. The model of a gyroscope mounted onto an actuated gimbal was derived and simulated. To prove the concept worked, a custom-built platform showed it is possible for a balancing robot to stay upright with zero traction under the wheels.

Keywords: Robot, Stair-climbing, Balancing, Inverted pendulum, Gyroscope, LQR

Résumé

Robillard, Dominic. M.A.Sc. Université d'Ottawa, 2014. *Développement d'un robot montant des escaliers et d'un système de stabilisation hybride pour des robots s'auto-balançant.* Thèse dirigée par Dr. E. Lanteigne, Ph.D.

Les robots qui s'auto-balancent sont des plateformes mobiles uniques. Ils gardent leur équilibre sur deux roues en utilisant un système de contrôle. Ils peuvent tourner sur place et occupent peu d'espace au sol. Ils ont cependant des limitations techniques restreignant leurs applications en dehors du laboratoire : ils ne peuvent pas se mettre debout par eux-mêmes, monter des escaliers, ou surmonter des obstacles. Le système de balancement nécessite une friction au sol, donc ces robots peuvent facilement tomber en cas d'impact ou en allant sur une surface glissante.

La première partie de cette thèse propose un nouveau concept de robot qui apporte une solution aux problèmes mentionnés ci-haut. Un joint est ajouté sur le corps d'un robot à quatre roues motrices, ce qui permet au robot de monter des escaliers et de se lever lui-même. Un modèle pour monter des escaliers et une simulation du balancement sont dérivés et comparés aux résultats expérimentaux.

La seconde partie décrit un système de stabilisation pour un pendule inversé utilisant un mécanisme gyroscopique. Le but étant d'intégrer ce système dans un robot qui se balance. Ce système améliore la résistance aux perturbations extèrieures pendant l'équilibre et maintient le robot stable sur les surfaces glissantes. Le modèle mathématique du système, qui consiste en un gyroscope monté sur un cardan, est dérivé, simulé et validé avec l'aide d'une plate-forme expérimentale.

Mots clés : Robot, Escaliers, Balance, Pendule Inversé, Gyroscope, LQR

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Self-balancing robots are a type of robotic platform that, like humans, stay in balance to keep an upright body. They achieve this by applying controlled torques on their wheels in order to keep the robot's centre of gravity aligned with the wheel's contact point. A popular example is the Segway personal transportation platform shown in Fig. 2.9. Having only two wheels provides these platforms the unique ability to steer with zero turning radius, and because they use wheels have the potential to travel efficiently at high speeds. Another advantage is their small footprint relative to their height making them better adapted for human-machine interaction purposes.

One may ask why we don't see more of these robots for real applications? They have limitations that greatly limit their usability in the field. First, they cannot stand-up on their own. This is problematic because to start their mission they require an external input to put them upright. Additionally, if a disturbance makes them fall down, they also require assistance to continue. This is important for a self-balancing robot because many real world situations could make the robot unstable and fall down; for example, a bump on the ground, or rolling on slippery surfaces. When balancing, an active control system always corrects the robot's equilibrium to keep it from falling: this results in a constant jitter that can be problematic if the robot is equipped with cameras as it will affect the image quality. Another limitation, inherent to many mobile robots, is the inability to climb stairs or overcome large obstacles, which again are essential capabilities in real world scenarios.

Up to now, no complete solution exists to solve these issues without sacrificing features such as speed, size, simplicity, and cost. Some solutions were developed to give self-balancing robots the ability to stand-up automatically using a gyroscopic flywheel, but are very specific and would be difficult to integrate into a useful machine because of the ideal conditions it requires to operate (Fig. 2.13). Other options that were considered was to add a kickstand that would deploy before the robot stops balancing and retracts when the robot is ready to balance again (Fig. 2.10). Current stair-climbing robots are inefficient tracked vehicles that have minimum size requirements to be capable of climbing stairs, or have complex mechanisms, making them large and heavy (Figs. 2.19 to 2.22(b)).

This two part research proposes a unique type of self-balancing robot, and presents the design of a novel hybrid stabilization system. Together the robot design combined with the hybrid stabilization solves all the aforementioned issues. The robot is self-balancing on two wheels, has the ability to travel at high speed, and can stand-up with no assistance. It is able to climb stairs and other obstacles, all with a single revolute arm. With the hybrid system, it can travel on slippery ground, and improves its balancing robustness and smoothness.

The thesis begins with a literature review in Chapter 2 on robotic locomotion types, existing balancing robots, stair-climbing methods, and examples of gyroscopic stabilization techniques. Chapter 3 on the stair-climbing robot first establishes a conceptual design to obtain the ideal layout for the robot. This is followed by the fabrication and assembly, the modelling and simulation, and finally the experimental field testing results. The hybrid stabilization system is covered in Chapter 4, where we first focus on the theory and application of the mathematical model, the design of a control system, and simulation. Then we study the design, fabrication and results of an experimental platform proving the theory. We will conclude with a discussion on implementation issues and future work with emphasis on applications to self-balancing robots.

Chapter 2

Literature review

Mobile robots navigating on a 2D plane are driven with a minimum of two actuators, one for each degree-of-freedom. The number and arrangement of degrees-of-freedom affect the robot's motion capabilities, planning, control and other navigation aspects. This survey examines the most common drive configurations for mobile robots, and outlines the capabilities, limitations and design considerations of each configurations.

2.1 Single wheel drive

In this configuration, a single wheel is used for both propulsion and steering (dark wheel in Fig. 2.1), and two casters are used to stabilize the robot on the ground. Analytically, it is the simplest non-holonomic configuration since the linear velocity and angular velocity are decoupled. As such, this type of mobile platform can be programmed to follow a path with little difficulty.

2.2 Ackermann steering

Ackermann steering (Fig. 2.2) is the same configuration we find in cars since its behavior is stable at high speed. The Ackermann steering has a minimum turning radius,

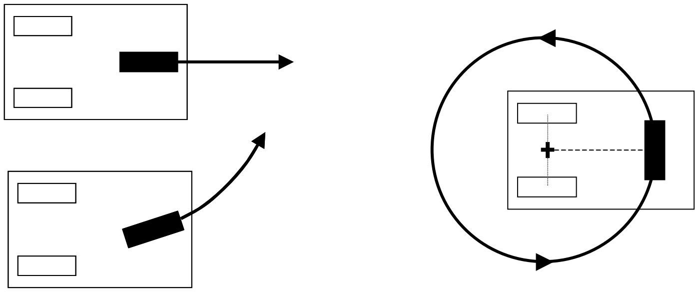


Figure 2.1: Driving and rotation of single wheel drive [1].

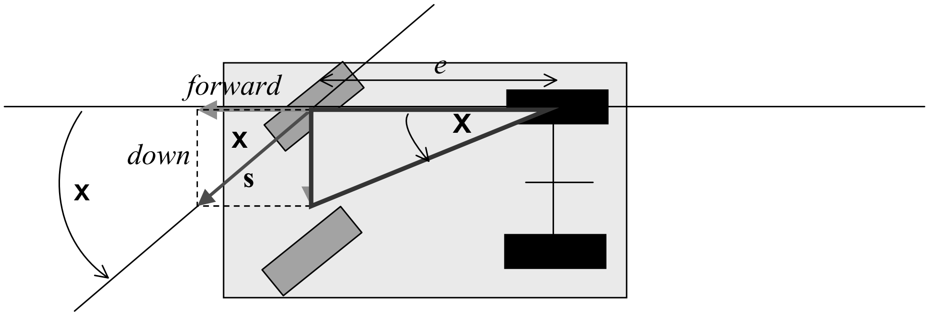


Figure 2.2: Motion of vehicle with Ackermann steering [1].

and typically requires higher level planning to navigate tight spots. Consequently, it's usability in obstacle avoidance applications or exploring unknown terrain is limited.

2.3 Synchro-drive

The synchro-drive is another configuration better suited for area coverage applications such as robotic vacuums. It is similar to the single wheel drive in the sense that it can move in any direction but this arrangement only turns the three wheels together (see Fig. 2.3). The three wheels are forced to turn together by either gears or chains. Only one wheel needs to be driven so this type of robot is still two degrees-of-freedom. The body of the robot always points in the same orientation it was pointing when the robot was first positioned on the floor. This has the advantage of keeping all the sensors pointing in the same direction at all times, which makes the platform a good candidate for mapping applications. Because the robot has to stop and realign its wheel to change its direction of motion, it is not considered a fully holonomic drive. It can however travel along a curve while in motion.

2.4 Omni-directional drive

The omni-directional drive is achieved using Mecanum wheels (Fig. 2.4). These wheels are made of a set of rollers 45° from the wheel's axis allowing the wheels to roll in any directions based on the rotation of the wheel, and the rotation of the other wheels of the robot. Because of their operation, each wheel must be actuated. They are however the only known wheel configuration to be truly holonomic. Since the robot

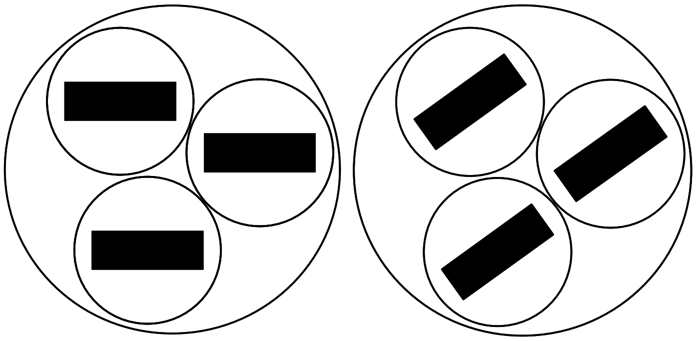


Figure 2.3: Schematic diagram of the synchro-drive principle [1].

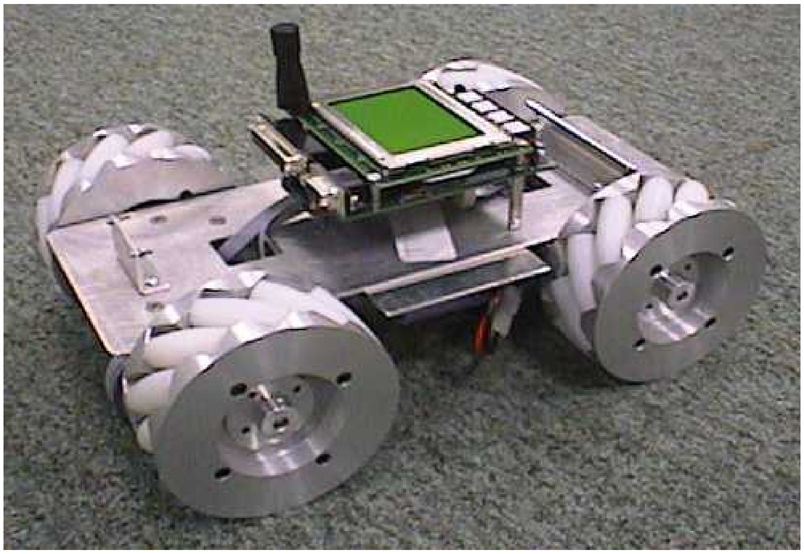


Figure 2.4: A Mecanum wheel used on a 4-wheeled omni-directional robot [1].

• Driving forward:	all four wheels forward	
• Driving backward:	all four wheels backward	
• Sliding left:	1, 4: backward;	2, 3: forward
• Sliding right:	1, 4: forward;	2. 3: backward
• Turning clockwise on the spot:	1, 3: forward;	2, 4: backward
• Turning counter-clockwise:	1, 3: backward;	2, 4: forward

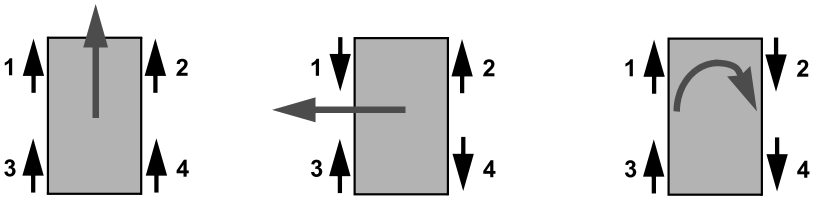


Figure 2.5: Kinematics of omni-directional robot [1].

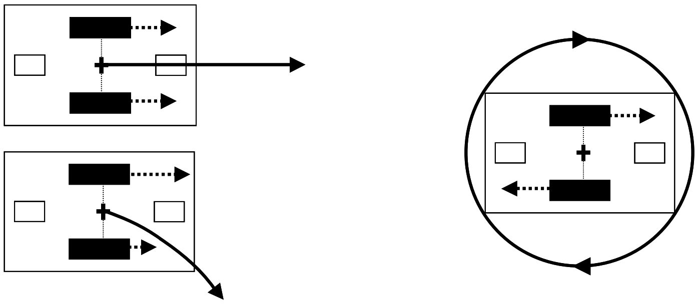


Figure 2.6: Driving and rotation of differential drive [1].

can move in any direction, anytime, the programming of path planning algorithms is greatly simplified. Figure 2.5 illustrates how a four-wheeled Mecanum robot can move forwards, sideways, and rotate. By mixing these motions by different amounts, the robot can move in any direction.

Their use is mostly limited to research applications since, by design, they only work on flat grounds, at low speed and must maintain traction. As soon as the rollers cannot turn properly (carpet, gravel, or slippery surfaces), the robot loses its ability to drive sideways.

2.5 Dual drive robotic platforms

2.5.1 Differential drive

The differential drive configuration uses two independent motors to rotate two wheels parallel to each other as shown in Fig. 2.6. The dark wheels are driven and the white wheels are casters. Like the single wheel drive configuration, this platform can turn on the spot at null speed and turn while moving forward. It is however mechanically simpler since the drive motors are fixed to the platform. On the other hand, this type of drive is more complex to control. Moving while turning requires careful coordination of the wheel speeds.

Self-balancing robots fall into this category but the casters (white squares on Fig. 2.6) are eliminated and replaced by a dynamic balance control system.

Figure 2.7: The EyeTrack robot, a tracked drive mobile platform [1].

2.5.2 Tracked drive

Tracked drives are a variation of the differential drive, where the powered wheels are replaced by tracks as shown in Fig. 2.7. This increases the robot's ability to overcome obstacles and navigate rough terrain. If the tracks are sufficiently long it can also climb stairs. However this added maneuverability comes at the price of mechanical complexity, drive train inefficiency, and increased effort while steering. Some designs include an articulated segment with tracks on it, allowing the robot to easily climb stairs. The Urban II platform [2] shown in Figure 2.8 is one example, but at more than half a meter of length and at 20kg, the robot is large.

2.5.3 Self-balancing inverted pendulum robots

Self-balancing robots have two wheels in contact with the ground and use a gyroscopic sensor coupled with a dynamic control loop to keep their body upright. Perhaps the best example of a commercially successful self-balancing platform is the Segway vehicle (Fig. 2.9(a)).

The concept of self-balancing robots is not new (1986 [5]), but the recent popularity of the Segway, the availability of low cost gyroscopic sensors, and powerful low-cost microcontrollers have inspired many versions of self-balancing robots both

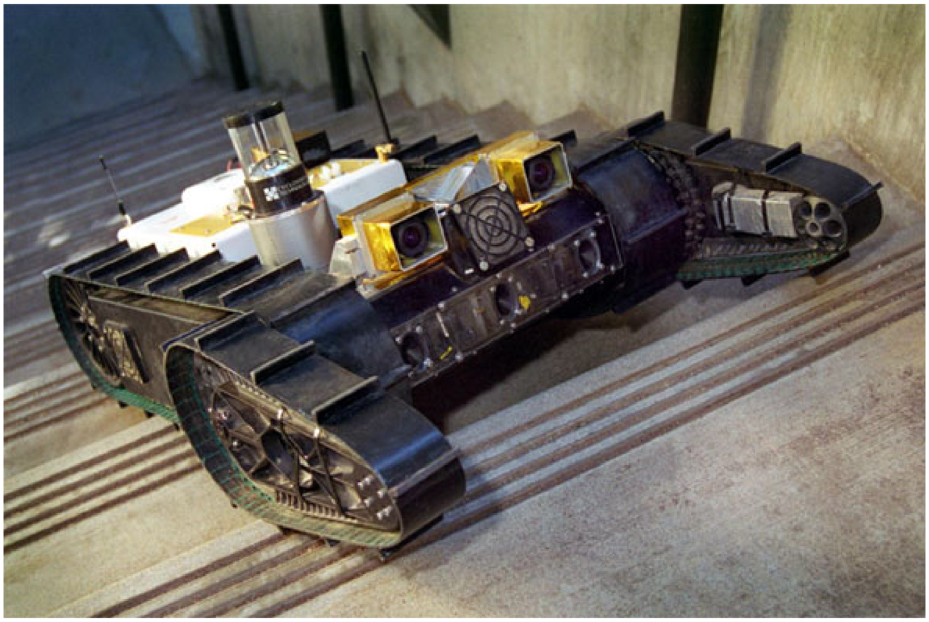
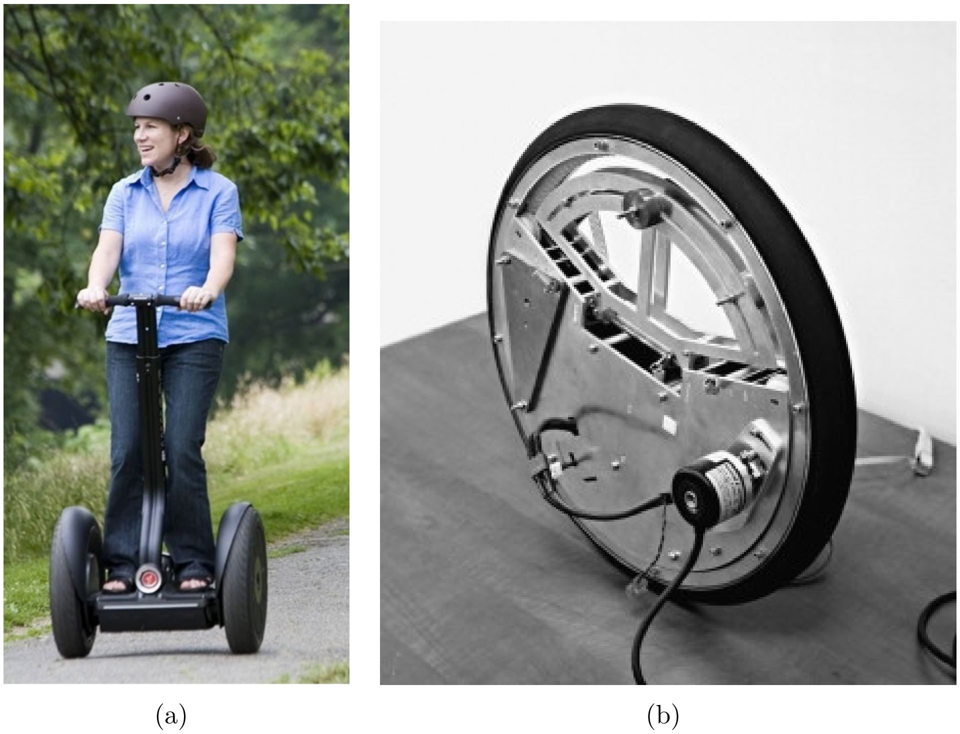


Figure 2.8: The Urban II with an articulated tracked drive [2].



(a)

(b)

Figure 2.9: (a): Tourist riding the Segway electric balancing scooter [3]. (b): The mono-wheel robot prototype during the self-stabilisation phase [4].

from research environments and amateurs.

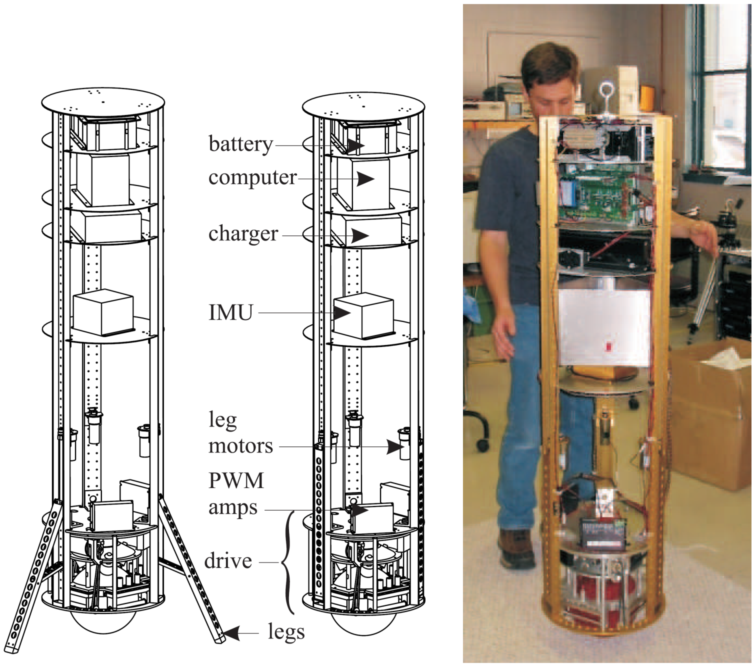
In [6], the authors look at how a self-balancing robot with two wheels can navigate along a predefined path. A six states state-space model is derived which includes the robot's steering mechanism and forward/backward motion of the robot as well as the rod's angle. They used a Linear Quadratic Regulator (LQR) controller to obtain the optimal gains and showed with simulations and a prototype robot that the resulting motion was smooth and stable for following a predefined path.

In [4], the authors demonstrated the capabilities of a mono-wheel robot - a compact robot shaped into a wheel, with all the mechanics and electronics hidden inside (Fig. 2.9(b)). For balancing the robot uses a mass pendulum actuated by a servomotor inside the wheel. The authors proved the robot is stable even when the wheel is not spinning (due to the internal pendulum) and that it gains stability as it goes faster (due to the gyroscopic effect from the wheel itself).

Self-balancing robots are very sensitive to impacts. In [7], the authors investigated methods for increasing the impact tolerance of soccer ball robots. Kicking a ball with a self-balancing robot can be a difficult task because not only the robot must keep balance, it must also stay in balance after the impact, which acts as a disturbance on the robot. In this article, the authors successfully developed a controller that can predict the result of the impact and tilt the robot just before impact so that after the impact the robot is in a stable angle. This allows the robot to kick footballs with higher speeds.

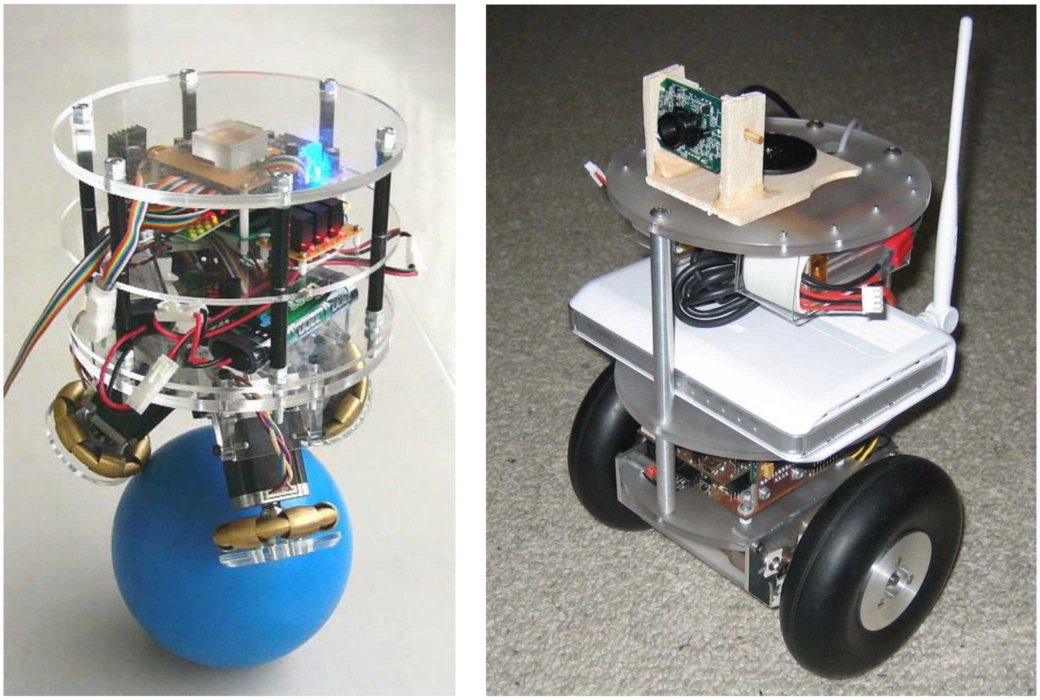
The robot shown in Fig. 2.10 is a self-balancing robot that balances on a sphere. Contrary to most self-balancing robots that balance on two wheels, this robot needs to actively stabilize two axes. To drive the spherical wheel, they used a mechanism similar to the one found in tracked PC mice where two perpendicular rollers each with their own motor forces the ball to rotate. The authors used a LQR controller to balance the robot. This concept allows the robot to move in an holonomic fashion. A similar robot is shown in Fig. 2.11(a). This design features three omni-wheels to control the ball (instead of rollers) [8].

Other examples of two-wheel self-balancing robots include Fig. 2.11(b) where the



(*a*) (*b*) (*c*)

Figure 2.10: Ballbot design and realization: (a) with three legs deployed, (b) with legs retracted into body, (c) balancing and station keeping [9].



(a) (b)

Figure 2.11: (a): Robot balancing on a ball using omni-wheel actuators [8]. (b): The balancing platform used for the obstacle avoidance experiment [10].

Figure 2.12: Murata Boy, self-balancing bicycle riding robot. [14].

authors developed a platform to navigate around tight obstacles using image processing techniques. The nBot [11] was developed in 2003 and shows that a self-balancing can be robust on rough and uneven terrain at moderate speeds. In [12] the authors added a retractable kick-stand on the robot to stop and resume balancing at any time. In [13] the authors study the effect of adding a payload (baggage) on the robot and how it affects its navigation capabilities.

It is also possible to balance a bicycle on two wheels. For example the Murata Boy shown in Fig. 2.12 uses a reaction wheel inside the robot's body as a torque generator to actively balance the bicycle. Its speed is increased or decreased to generate reaction torque about the spin axis. Advantage of such system is low cost, simplicity and ability to balance with no ground motion. The disadvantage is that it consumes more energy and it is not capable of producing large reaction torques.

Little work has been done on giving a self-balancing robot the ability to stand-up on its own. There is a lot of research done in the stand-up routine of humanoid robots [15] but they are not applicable to wheeled robots since these platforms have legs with

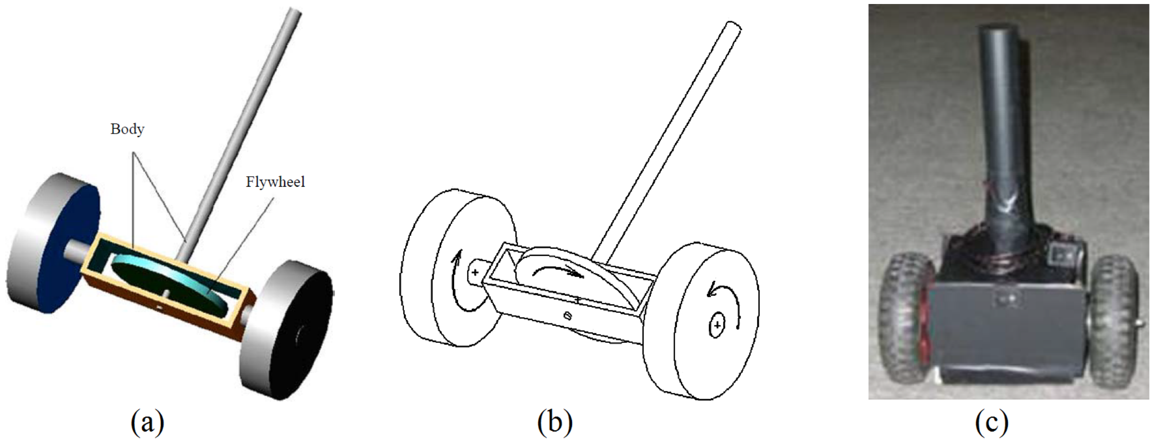


Figure 2.13: The self tilt-up balancing robot. (a): Structure; (b): motion made by the wheels to raise the pendulum; (c): prototype of the robot [16].

many degrees-of-freedom. Self-erecting an inverted pendulum has been examined, but the methods used (swing the pendulum back and forth from the hanging position until enough energy is provided to raise high) cannot be applied to wheeled robots.

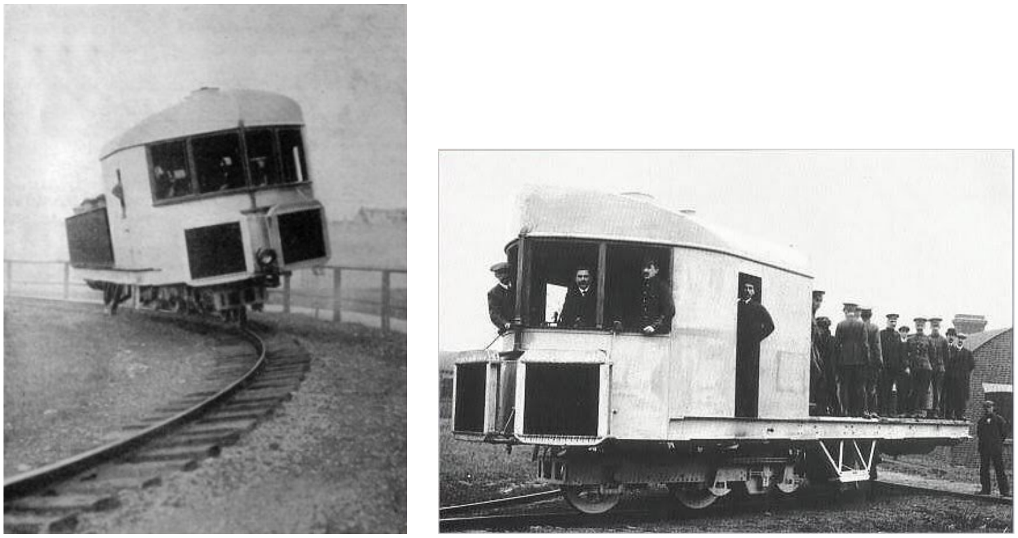
In [16], the authors show an interesting method to erect the robot without assistance by mounting a flywheel (mechanical gyroscope) inside the robot's body. The process shown in Fig. 2.13 works by rotating the drive wheels in opposite directions causing the flywheel and body to move around a fixed point. As a result of the precessional motion on the flywheel, a moment is synchronously generated to tilt the body upright. This robot can also lie down slowly without impacting the ground, which reduces strain on its internal components. The main issue with this method is if the wheels slip under the load it will not work.

For an overview on operation of self-balancing robots, and to learn about the author's previous self-balancing robot, refer to Appendix B.

2.6 Mechanical gyroscopic stabilization

2.6.1 Active stabilization

A mechanical gyroscope is a rotating mass that is spinning fast enough to store a large amount of momentum. Because of this momentum, changing the axis of rotation (tilting the gyroscope) creates a large moment resistance against the tilting motion.



(a) (b)

Figure 2.14: Brennan's gyro monorail (1909) [17].

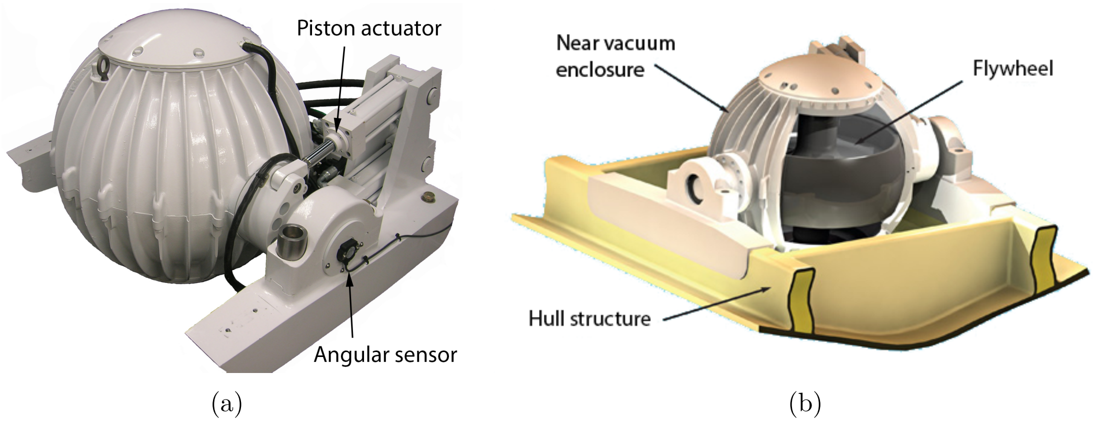
As the body is forced to rotate, it will respond by also rotating in another axis which is called the precession axis. This precession can be used effectively to control the moment on the other axis. The force from mechanical gyroscopes can be used to actively stabilize some systems that cannot rely on external forces (stabilization in space).

An example is the Brennan Gyro Monorail shown in Fig. 2.14. The 22 tonne train is balanced on a single track and is robust to disturbances such as people on the train or weights moving. The train also banked in turns automatically to avoid falling sideways.

Two counter rotating mechanical gyroscopes coupled with an hydraulic actuator kept the train balanced. Using the actuator, the gyroscope could be forced to precess and generate the necessary torque to regain balance.

Mechanical gyroscopes are also used to stabilize various vehicles. For example, Seakeeper (Fig. 2.15) manufactures gyroscopes mounted in a gimbal to stabilize the roll of the boats. To maximize efficiency, the gyroscopic disk is enclosed inside a near vacuum that reduces air friction and there is an hydraulic piston to control the angle of the gimbal.

Another example is the self-balancing bicycle that uses a control moment gyro-



(a) (b)

Figure 2.15: (a): A Seakeeper gyroscope used to stabilize the roll of boats [18]. (b): Section view of the Seakeeper stabilizer showing the internal gyroscope [18].

scope similar to the Gyro monorail. As seen in Fig. 2.16, the bicycle has a spinning gyroscope coupled to an actuator to generate balancing torques.

Often, the limiting factor in high speed gyroscopes is the strength of materials, where a high rotational velocity will burst the wheel due to the centrifugal forces. A method consisting of layering different composite materials was found [19] to maximize the momentum of a gyroscope for a given diameter (maximizing the product of the moment of inertia and the maximum velocity).

2.6.2 Gyroscopic dampener

The Gyro Monorail seen previously needed an active hydraulic actuator to compensate for the loss of potential energy as the car tilted sideways. In some applications it is possible to use a gyroscope coupled with a spring (passive stabilization). Its purpose is to ensure the gimbal will not drift out of its central position. Because there are no actuators, the gyroscope absorbs energy from the system. This is better referred to as a gyroscopic dampener.

In [20], the authors use a gyroscopic dampener to avoid the hunting oscillations that can happen with trains travelling at high speed. Hunting oscillations happen as the axle repositions itself on the track due to conical shape of the wheel's surface. The degrees-of-freedom (Fig. 2.17) on which the axle can move relative to the track creates a self-exciting vibration that gains intensity as speed increases. They built a

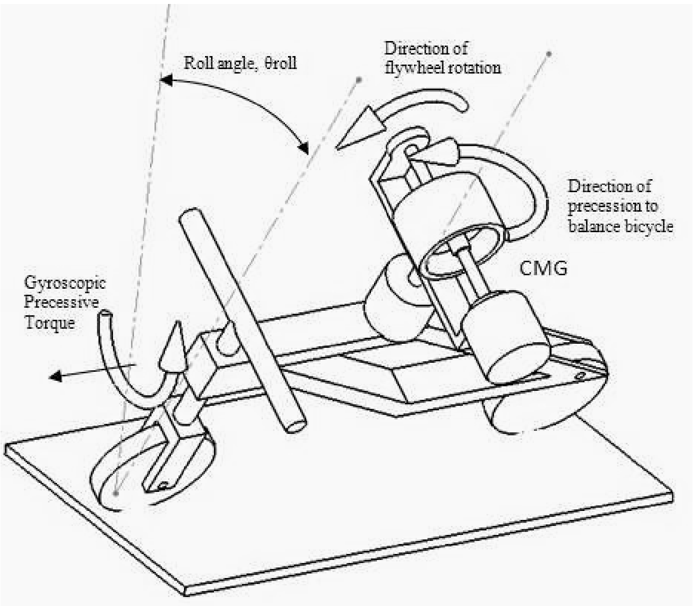


Figure 2.16: Balancing of bicycle using gyroscopic precession torque generated by a control moment gyro[14]

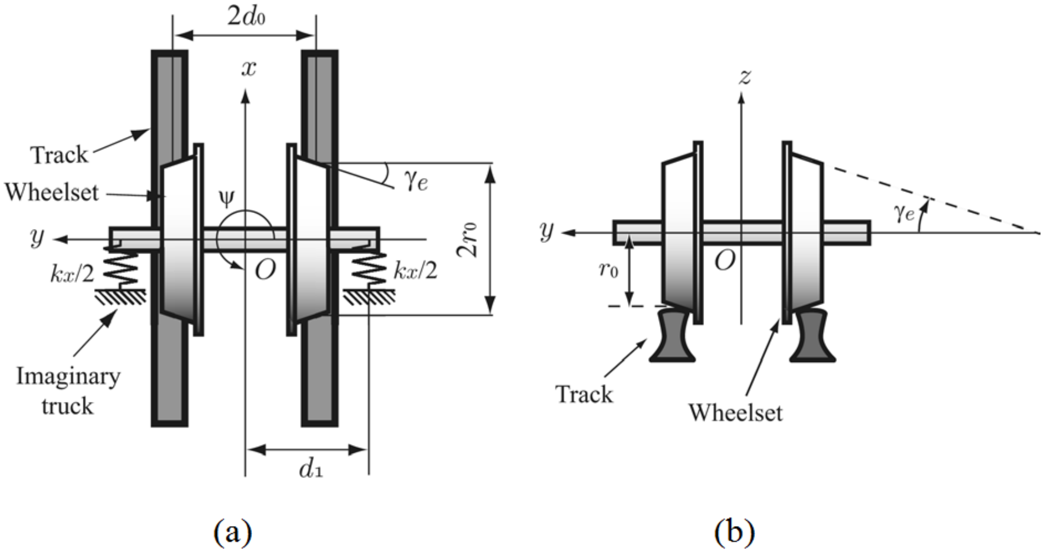


Figure 2.17: The hunting motion that occurs on railways can be modelled as an oscillation that involves motion in both ϕ and α_e. [20]

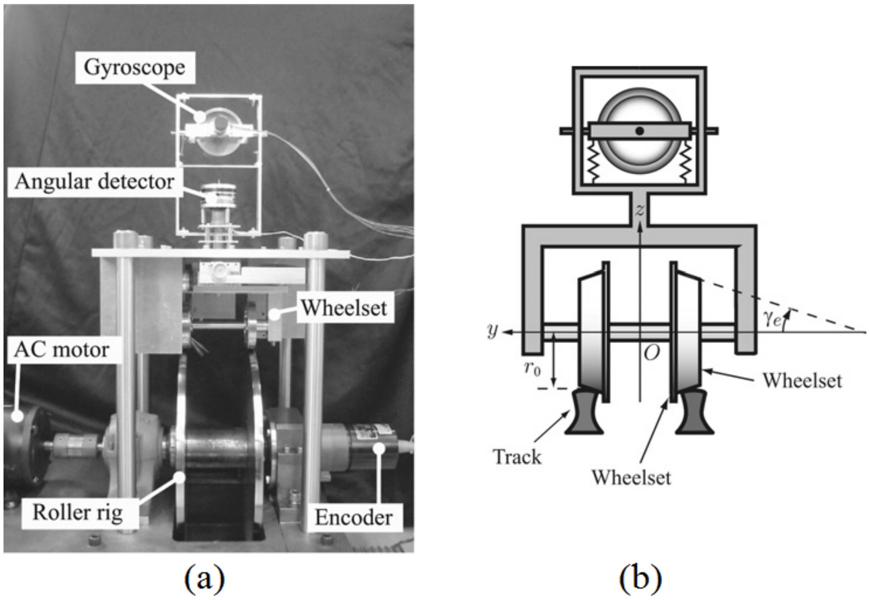


Figure 2.18: Apparatus to show the effect of the gyroscopic damper on the hunting oscillations [20].

single axle apparatus that recreates the conditions where hunting motion will occur shown in Fig. 2.18. By adding a gyroscopic damper, they were able to increase the critical speed at which the oscillations occur.

2.7 Stair and obstacle climbing robots

We saw in the previous section how balancing can be achieved with gyroscopes. To develop a stair-climbing and self-balancing robot, a review of existing robots that can climb stairs and other obstacles must be done.

The most common configuration of stair-climbing robot is the tracked drive. Providing the threads are sufficiently long to overlap at least two steps at a time, the robot sees the staircase as a slope and can easily climb stairs. However the difficulty is in transitioning from a flat surface onto the first step. For this, several methods have been proposed. An example is the MACbot four-tracked robot shown in Fig. 2.19 can move its front and rear tracks relative to each other to effectively climb the first step and then re-align its tracks and climb the remaining steps [21].

Another method is to use a deformable/reconfigurable track such as the LMA

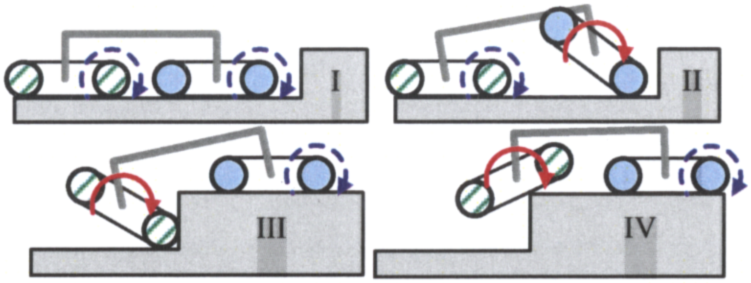
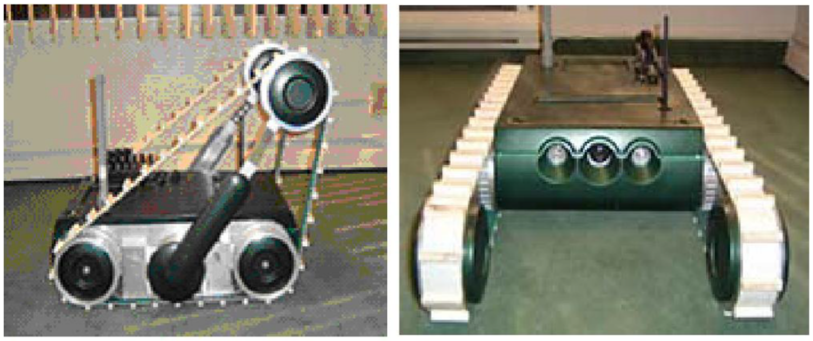


Figure 2.19: Climbing operation of the MACbot [21].



(a) Side view (b) Front view

Figure 2.20: Side and front view of the LMA tracked robot [22].

robot shown in Figs. 2.20 and 2.21.

The Whegs robot shown in Fig. 2.22(a) is a good example of a biologically-inspired robot that is small, compact, and able to climb obstacles such as steps. However, these types of robots may have trouble climbing a flight of stairs (more than one step). Also, because they do not have wheels, they are less efficient on flat ground.

The StairBot [24] in Fig. 2.22(b) is another example of stair-climbing robot. It uses two large wheels mounted on a prismatic joint, along with support members, to lift itself on every step. This robot works but is large and complicated. It also seems it is not very adaptable to other types of obstacles and will not be very efficient on flat terrain.

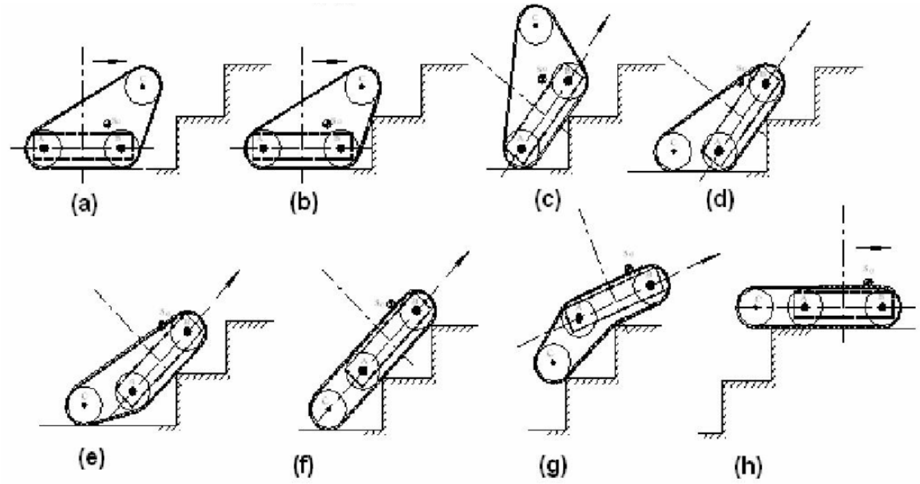
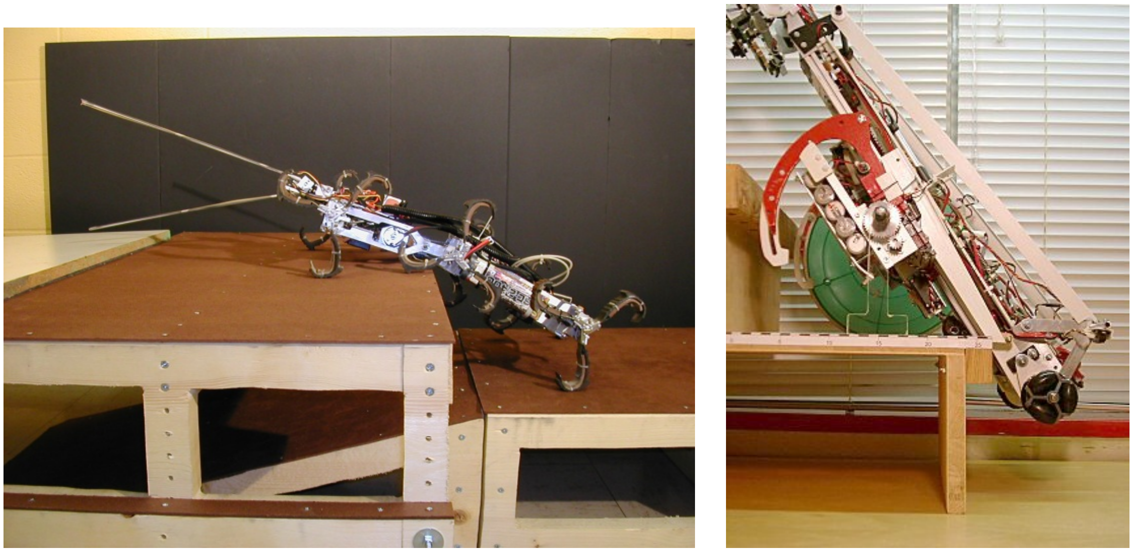


Figure 2.21: Climbing operation of the LMA robot [22].



(a) (b)

Figure 2.22: (a): The Whegs robot climbing a step [23]. (b): Picture of the stairBOT climbing a step [24].

Chapter 3

Development of the stair-climbing, self-balancing robot

The objective of this research was to develop a simple and effective method to solve the inability to stand-up and climb obstacles for self-balancing robots. The robot proposed in this chapter achieves both tasks with a simple design. An article written by the author has been published on the stair-climbing robot at the *CCToMM Mechanisms, Machines, and Mechatronics (M3) Symposium, 2013*. A reproduction of the article is included in Appendix I.

3.1 Design overview

Since the goal is to develop a robot that can both climb stair and balance on two wheels only the configurations that gives both capabilities to a robot were considered. To limit the project in scope, several constraints were imposed. Because many stairs in real life do not have vertical faces, any concept that would rely on this face to successfully climb has been discarded. Also, robots that do not use round wheels for navigation were excluded such as robots moving on a single sphere, or with legs for locomotion, and robots with special types of wheels (using prongs, tri-star wheels, omni-wheels, and others).

A total of six configurations were considered. The decision analysis for selecting the optimal configuration is outlined in Appendix C.

The proposed solution is to have a four-wheel drive differential steering mobile platform as shown in Fig. 3.1. The left and right sides of the robot each have their own drive motors and both wheels on the same side are coupled. This platform provides good mobility on flat ground and is suitable for a self-balancing robot since the opposite wheel provides protection when it robot falls from the upright position.

This platform is augmented with an arm attached at the middle of the body. This arm is driven by a powerful actuator and can indefinitely rotate around its joint. It is

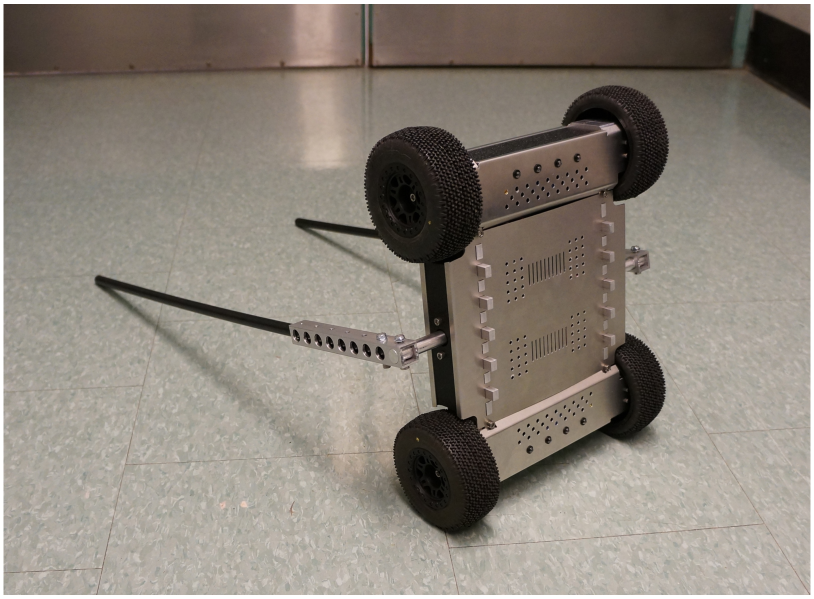


Figure 3.1: Photograph of the robot developed for climbing stairs and self-balancing. Including its wheels, the standing robot is 14 inches high. The arm is 26 inches long.

about twice as long as the body. The body carries all the heavy components while the arm is made as light as possible to minimize the centre of gravity positional changes.

To stand-up from a resting position (four-wheels in contact with floor), the arm rotates and lifts the body of the robot until it stands upright. At which point the balancing can start.

To climb stairs the robot aligns itself in four-wheel mode with the first step and the arm starts to rotate against the floor on the opposite side from the stairs. Once the arm touches the floor, the whole robot's body starts to turn. After a 1/4 turn, the body falls on the nose of the first step while the arm keeps rotating.

This process goes on for every step while the whole body of the robot continuously rotates. This is a simple effective method. It does have the drawback of making it more difficult to use sensors and cameras while the robot is climbing stairs. However because of the long arm towards the bottom of the stairs the climbing sequence is reliable. If the robot misses a step, it is almost impossible for the robot to tumble all the way down. The robot does not rely on the balancing ability while climbing steps.

3.2 Design analysis

3.2.1 Design constraints

The following constraints were applied to our design:

- Travel speed while balancing of 10 km/h or 2.8 m/s.
- Ability to roll over obstacles 1/3 of the wheel radius.
- Recovers from 20° disturbances in less than a second.
- Can climb all the stairs compliant to the building code shown in Appendix A.
- Arm powerful enough to climb two steps per second.
- Can drive a 45° incline.
- Can last about one hour on a single battery charge.

The overall size of the robot will affect the range of stair dimensions the robot can climb. As mentioned previously, at a minimum the robot should be able to handle all the stairs within the construction code and the ones we have found as examples. Figure 3.2(a) shows the nomenclature for the basic dimensions considered in the conceptual stage.

There are several choices for the wheel diameter, all of which give a suitable final result. However, there are some rules to follow:

a) The wheel should not be too large as to not fit nicely into a step. Essentially, the wheel radius should not be more than the thread of a step.

b) The wheel should be at least large enough to grab the nose of a step with a vertical reaction force. This includes steps that have a fillet or chamfer to them. This minimum value is directly related to a number of parameters such as the robot's body length, the robot's body width, and the rise of a step. See Fig. 3.2(b).

There is a chance that the robot falls down between two steps if the risers are open as shown in Fig. 3.3. After consideration, it was decided that there are not enough arguments to increase the wheel size above four inches in diameter even if there is a risk of falling trough open risers.

Another aspect to consider is the presence of grousers on the body of the robot. These will avoid the slippage of the body as it grabs the edge of a step as shown in Fig. 3.4.

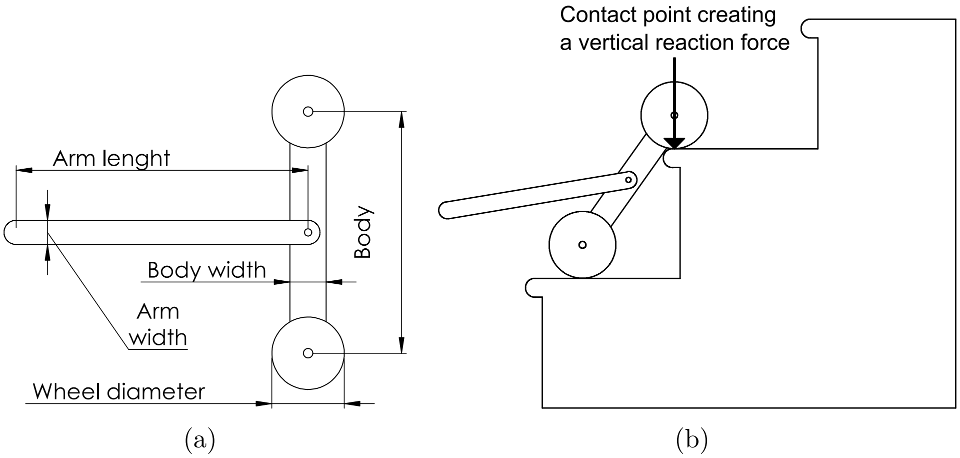


Figure 3.2: (a): External dimensions that are considered in the conceptual design. (b): Minimum wheel size relative to other dimensions

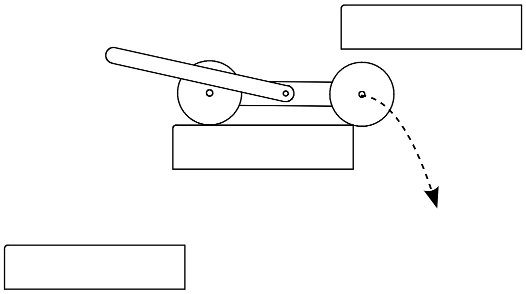


Figure 3.3: Showing how the robot can fall with open riser steps.

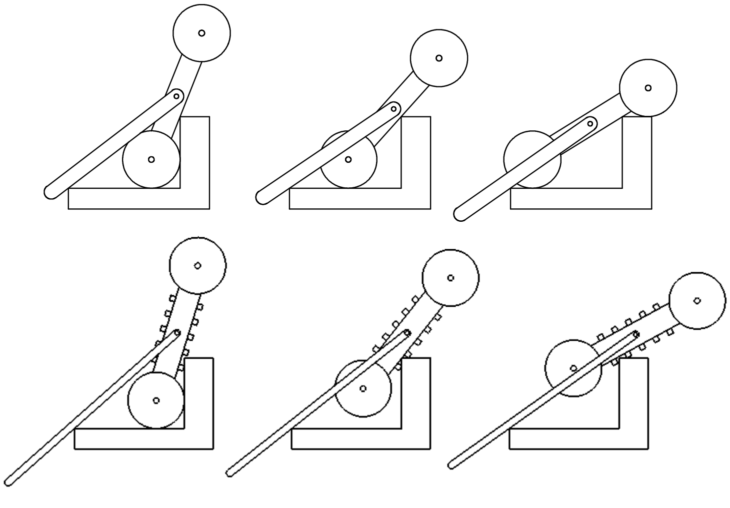


Figure 3.4: Showing the use of grousers on the robot's body to prevent slippage.

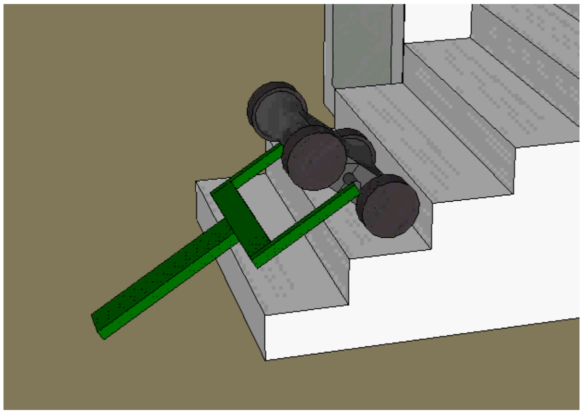


Figure 3.5: View of the conceptual robot simulation during a stair-climbing sequence.

3.2.2 Proof-of-concept simulation

Before going into a detailed analysis, a proof-of-concept simulation was performed on the design. This simulation was built in order to ensure the idea was feasible and would give information on the potential issues during actual use. The simulation was developed in Google Sketchup with the SketchyPhysics add-on which internally uses the Newton SDK, a fast and accurate physics library [25]. Figure 3.5 shows a view of the simulator in action. The robot was controlled in its virtual environment using a joystick. A video of the simulation is available in [26].

3.2.3 Modelling

Because the design was an iterative process the model values below were first estimated before fabricating the robot, and they were updated to reflect the real robot after its completion. The robot parameters used in the models were measured from the prototype and are given by:

- $M = 5.012$kg, total mass of robot.
- $r = 0.050$ m, wheel radius.
- $l = l_b = 0.127$ m, half of the distance wheel-to-wheel.
- $I = 0.056$kg·m^2, robot's moment of inertia.
- $I_r = 0.001$kg·m^2, rotational component's moment of inertia.

In the following two sections the models for balancing and for stair-climbing are derived. These two models are unrelated as the robot is not balancing while climbing

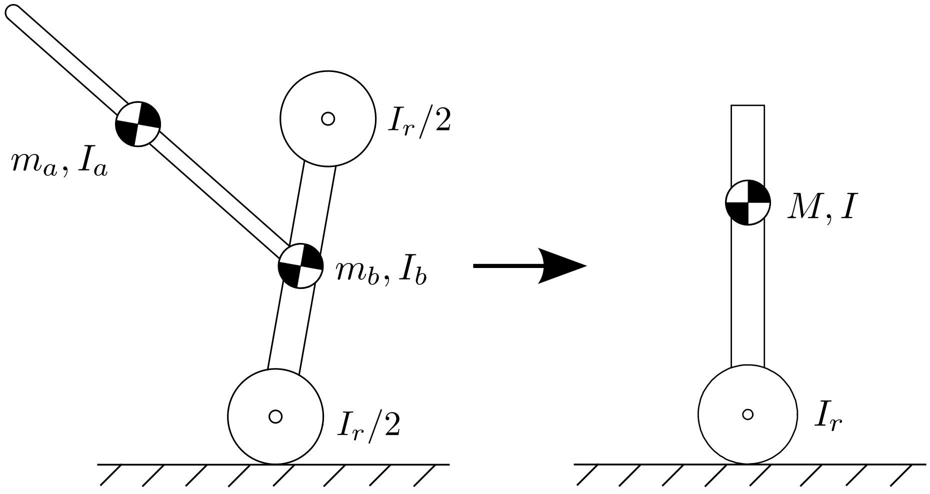


Figure 3.6: The arm and the body are both combined into a single mass for the balance analysis (given that the arm is not rotating).

stairs.

3.2.3.1 Model for Balancing and Controller Design

For the analysis of balancing, it is assumed that the arm is lumped with the body. As long as the arm does not rotate while balancing, the effect will be the same. If the arm is moving, it is simply considered as a disturbance. For the purpose of analyzing the balancing, the assumption is valid because the arm will not be moving. See Fig. 3.6. The top wheels are mechanically coupled to the bottom ones so their moment of inertia must also be taken into account. Additionally, we assume a no-slip condition, where the friction of the wheel on the ground is sufficiently high to prevent slipping.

The analysis will be made following the coordinate system shown in Fig. 3.7. The robot is broken down into two parts: the body and the wheel.

The position of the wheel is x. In the following equations the wheel is considered mass-less as its weight is included as part of the body weight M but it still has a mass moment of inertia I_r. This simplification is valid because the wheel is always at the same position relative to the body, but it can rotate relative to it. Applying Newton's second law of motion along x and y coordinates, and with moments along z on Fig. 3.7 we obtain:

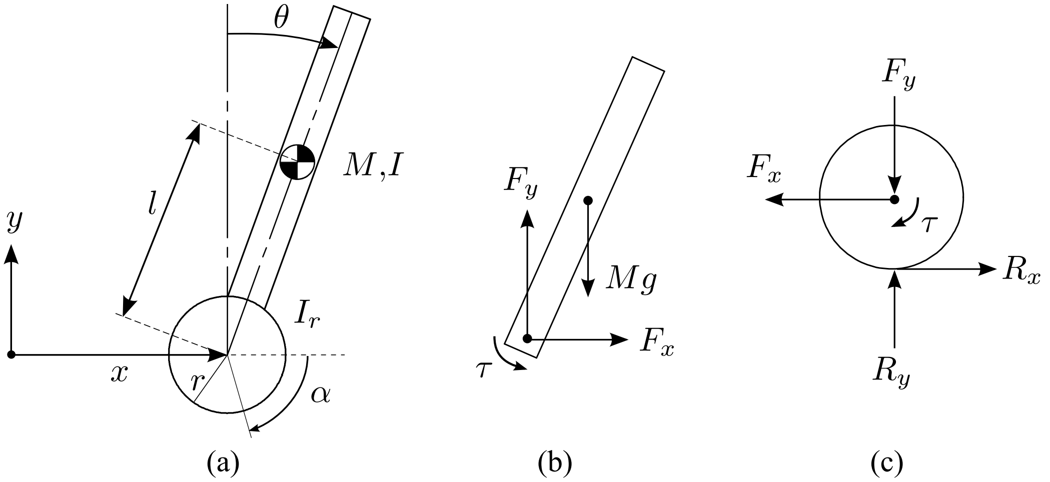


Figure 3.7: (a) Nomenclature of the robot balancing; (b) forces acting on the robot's body; (c) forces acting on the robot's wheel. τ is the control input (torque between wheel and body).

$$I_r\ddot{\alpha} - \tau + Mr(\ddot{\alpha}r + \ddot{\theta}l\cos\theta - \dot{\theta}^2 l\sin\theta) = 0 \tag{3.1}$$

$$Mgl\sin\theta - \tau = I\ddot{\theta} + Ml^2\ddot{\theta} + Ml^2\dot{\theta}^2\sin^2\theta + Mlr\ddot{\alpha}\cos\theta - Ml^2\dot{\theta}\cos\theta\sin\theta \tag{3.2}$$

Because the robot is balancing, we assume its body angle θ is stable and is close to zero. Applying small angle approximations $\sin\theta \approx \theta$, $\cos\theta \approx 1$, and $\theta\dot{\theta}^2 \approx 0$ and re-arranging the equations we have:

$$C\ddot{\alpha} = (I + Ml^2 + Mlr)\tau - M^2gl^2r\theta \tag{3.3}$$

$$C\ddot{\theta} = Mgl(I_r + Mr^2)\theta - (I_r + Mlr + Mr^2)\tau \tag{3.4}$$

with: $C = IMr^2 + I_rMl^2 + II_r$

From where we write the state space representation as:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u \tag{3.5}$$

with:

$$\mathbf{x} = \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{Bmatrix} = \begin{Bmatrix} \theta \\ \dot{\theta} \\ \alpha \\ \dot{\alpha} \end{Bmatrix}, \quad \mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \dfrac{Mgl(I_r + Mr^2)}{C} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -\dfrac{M^2gl^2r}{C} & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 \\ -\dfrac{I_r + Mlr + Mr^2}{C} \\ 0 \\ \dfrac{I + Ml^2 + Mlr}{C} \end{bmatrix}, \quad u = \tau$$

The control law for the robot's balancing is defined as:

$$u = -\mathbf{K}\mathbf{x} \tag{3.6}$$

where **K** needs to be determined. Here we will use the Linear Quadratic Regulator (LQR) method to design an optimal controller [27]. To do so, we define the performance index to be minimized as:

$$J = \int_0^\infty (\mathbf{x}\mathbf{Q}\mathbf{x} + Ru^2)dt \tag{3.7}$$

with **Q** and R being optimization parameters that need to be chosen. These values need to found by trial and error in the simulation, although as starting point, the Bryson's rule can be applied [28]. To minimize J we use the MATLAB® "lqr" function that takes as parameters the state space model and the **Q** and R constants. The "lqr" function returns **K** for the control law such that J is minimized. After several trials, the **Q** and R parameters that gave a satisfactory response were:

$$\mathbf{Q} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 20 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \text{ and } R = 0.5 \tag{3.8}$$

for which the "lqr" function returns:

$$\mathbf{K} = \begin{bmatrix} -101.2 & -15.11 & -6.325 & -3.343 \end{bmatrix} \tag{3.9}$$

Since the robot is controlled with a microcontroller, the model needs to be discretized. The continuous state space model was converted to a discrete state space model using the MATLAB® "c2d" function and a new discrete $\mathbf{K}_d$ was computed using the "dlqr" function. We obtained for a sample rate of 50Hz:

$$\mathbf{K}_d = \begin{bmatrix} -11.07 & -1.473 & -0.5405 & -0.2960 \end{bmatrix} \tag{3.10}$$

It was also desirable for testing to have the nonlinear model of the robot. Some control systems will work with a linear model but are unstable with the nonlinear version. The control system cannot easily be designed from the nonlinear version,

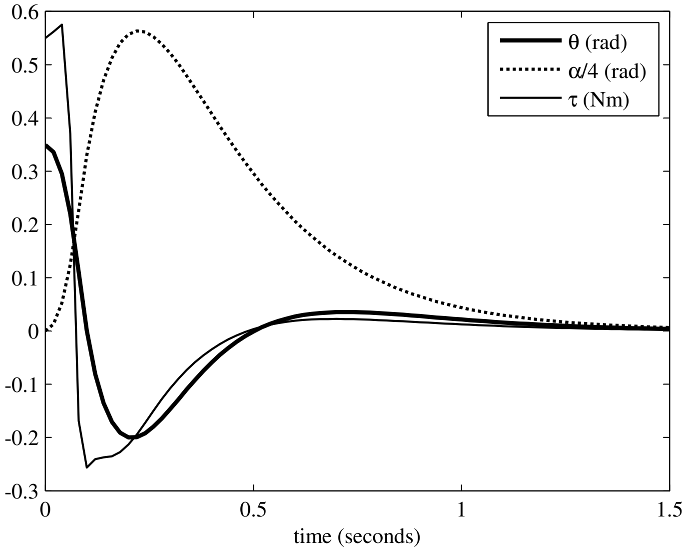


Figure 3.8: Simulation of the robot recovering from a 20° disturbance while balancing. The simulation is run with the discrete controller and the nonlinear model of the robot. Sampling rate is 50Hz.

but offers a good validation tool. Re-arranging Eqs. (3.1) and (3.2) by extracting $\ddot{\alpha}$ and $\ddot{\theta}$ we obtain the following two nonlinear equations:

$$\ddot{\alpha} = \frac{\begin{array}{c}\sin\theta(rM^2l^3\dot{\theta}(\dot{\theta}\cos\theta\sin\theta + \dot{\theta} - \cos^2\theta) - grM^2l^2\cos\theta + lrMl\dot{\theta}^2)\\ +(Ml^2 + rMl\cos\theta + I)\tau\end{array}}{M^2l^2r^2\sin^2\theta + I_rMl^2 + lMr^2 + II_r} \tag{3.11}$$

$$\ddot{\theta} = \frac{\begin{array}{c}\sin\theta(M^2lr^2\dot{\theta}(l\dot{\theta}\sin\theta - g - l\dot{\theta}\cos\theta + l\cos\theta) + MlI_r(l\dot{\theta}^2\sin\theta + l\dot{\theta}\cos\theta - g))\\ +(Mr^2 - Mlr\cos\theta + I_r)\tau\end{array}}{M^2l^2r^2\sin^2\theta + I_rMl^2 + lMr^2 + II_r} \tag{3.12}$$

A discrete model was written and the nonlinear model was augmented with the Pittman GM9236 motor model. It ensures that the current control law is compatible with the real torque limited motors. Figure 3.8 shows the simulation results for the pendulum subjected to a 20° disturbance. The system successfully recovers in about one second. The settling time could be shorter but it was observed that harmonic vibrations would appear on the real model as a result of changing the gain.

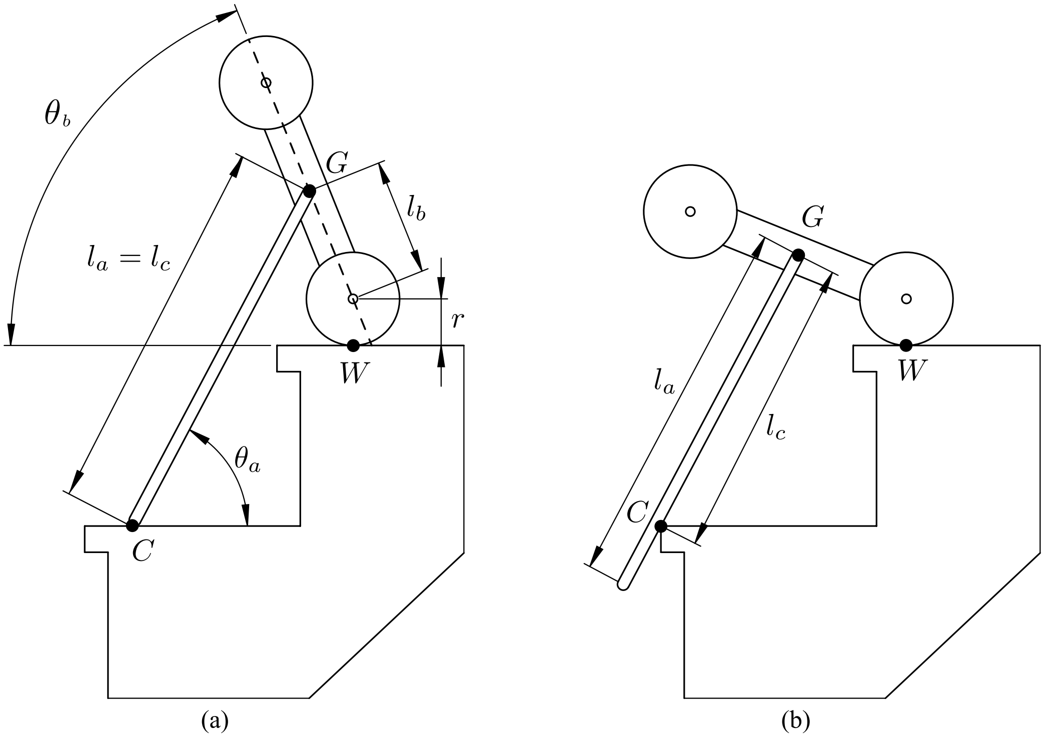
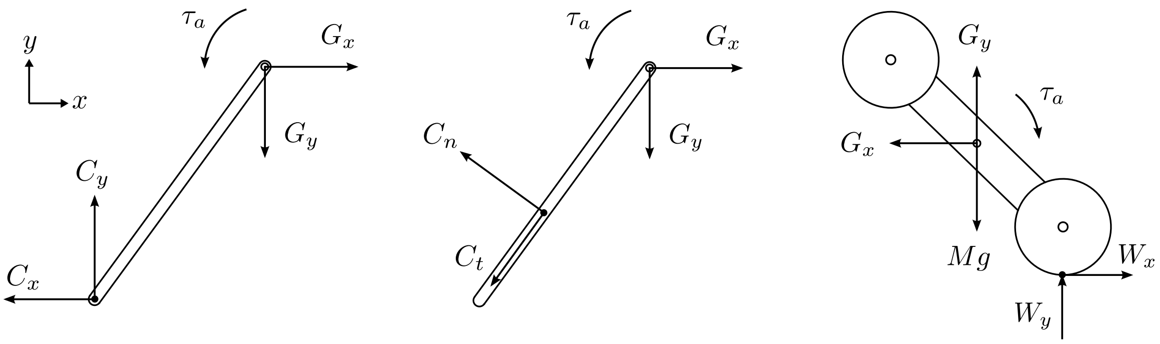


Figure 3.9: Schematic of the two cases encountered during the stair-climbing process. (a) Case A: the tip of the arm is in contact with a horizontal surface; (b) case B: the arm is in contact with the nose of a step.

3.2.3.2 Model for stair-climbing

In this section, the model for climbing stairs is developed. During the stair-climbing sequence, the robot relies on the friction of the wheel to overcome the normal forces generated by the arm against the steps, and the sliding friction force of the arm. The stair-climbing process can be simplified to have only two different configurations that we call case A and case B. This is a simplification as there exists other configurations (such as when the body of the robot is touching the step instead of the wheels), however these cases are not considered critical due to lower forces involved. Figure 3.9 shows the two configurations that were analyzed. Figure 3.10 shows the forces for both cases on the arm and the robot's body.

In this case, it is assumed that the robot is going up the stairs. Although there could be a sliding motion of the arm's contact point, the wheels of the robot are not slipping. A slipping condition would occur whenever force W_x is higher than the



(a) (b) (c)

Figure 3.10: Free body diagrams of the arm and the robot's body. (a) Forces applied on the arm under case A; (b) forces applied on the arm under case B; (c) forces applied on the robot's body for both case A and case B.

friction force the wheel can provide. The arm length was determined such that this never occurs in real stair conditions with a safety factor of at least two. The condition is:

$$W_x < \mu_w W_y \tag{3.13}$$

where μ_w is the coefficient of friction between the wheel and the step surface. Also for case A:

$$C_x = \mu_c C_y \tag{3.14}$$

where μ_c is the coefficient of friction of the arm at its contact point. And for case B:

$$C_t = \mu_c C_n \tag{3.15}$$

Case B can be re-written to be compatible with the equations of case A by transforming C_n and C_t into C_x and C_y. Then we have:

$$C_x = \begin{cases} \mu_x C_y & \text{if case A} \\ C_n(\sin\theta_a + \mu_c \cos\theta_a) & \text{if case B} \end{cases} \tag{3.16}$$

$$C_y = \begin{cases} C_y & \text{if case A} \\ C_n(\cos\theta_a - \mu_c \sin\theta_a) & \text{if case B} \end{cases} \tag{3.17}$$

Applying sum of forces on the arm and body, we obtain a set of equations that can be solved to obtain the arm and wheel torques (the wheel torque is defined as $\tau_w = rW_x$), as a function of the arm angle θ_a, body angle θ_b, and contact length l_c. For case A we obtain:

$$\tau_a = \frac{Mgl_b l_c \cos\theta_b(\cos\theta_a + \mu_c \sin\theta_a)}{D} \tag{3.18}$$

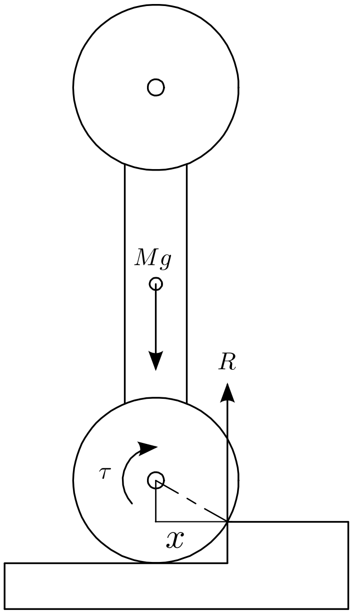


Figure 3.11: Diagram of the robot climbing an obstacle.

$$\tau_w = \frac{Mgl_b\mu_c r\cos\theta_b}{D} \tag{3.19}$$

with: $D = l_b\cos\theta_b - \mu_c r + l_c\cos\theta_A - l_b\mu_c\sin\theta_b + l_c\mu_c\sin\theta_a$.

For case B we obtain:

$$\tau_a = \frac{Mgl_bl_c\cos\theta_b}{E} \tag{3.20}$$

$$\tau_w = \frac{Mgl_br(\sin(\theta_a-\theta_b)+\mu_c\cos(\theta_a-\theta_b)+\sin(\theta_a+\theta_b)+\mu_c\cos(\theta_a+\theta_b))}{2E} \tag{3.21}$$

with: $E = l_b\cos(\theta_a+\theta_b) - l_b\mu_c\sin(\theta_a+\theta_b) - r\sin\theta_a + l_c - \mu_c r\cos\theta_a$

3.2.4 Mechanical design

3.2.4.1 Climbing an obstacle

One of the design constraints is that the motor is strong enough to climb an obstacle 1/3 of the wheel radius. Figure 3.11 shows an illustration of the scenario.

Using the Figure, it can be shown that:

$$x = r\sqrt{4/9}$$

Just as the obstacle climb begins, the wheel does not touch the ground anymore. From where we write:

$$R = Mg$$

And taking the sum of moments, we obtain:

$$\tau - xR = 0$$

$$\tau = Mgr\sqrt{4/9}$$

Using $M = 5\text{kg}$, and $r = 50.8\text{mm}$, we have:

$$\tau = 5 \cdot 9.81 \cdot 0.0508 \cdot \sqrt{4/9}$$

$$\tau = 1.66\text{Nm}$$

This is the minimum torque to guarantee the robot will climb the obstacle. Since there are two motors, they need to provide 0.83Nm each at a minimum.

We choose the Pitmann GM9236 gearmotor (datasheet provided in [26]). The stall torque of the motor is 0.436Nm, and the motor is fitted with a 5.9:1 gearbox. Thus the stall torque of the gearmotor is technically 2.57Nm, however, the gearbox itself is designed to handle a maximum torque of 2.12Nm. Using this motor would satisfy the obstacle climbing requirement. We now define:

$$\tau_{stall} = 2.12\text{Nm}$$

3.2.4.2 Belt selection

The belt carries the motion from the motors to the opposite wheels. At worst case, the full motor's torque is carried by the belt. This is an unlikely scenario because the wheel would slip many times before the motor will stall. The axial force carried by the belt is:

$$F = \frac{2 \cdot \tau_{stall}}{p}$$

where p is the pitch diameter of the pulley. Assuming a pitch diameter of 31mm, we have:

$$F = \frac{2 \cdot \tau_{stall}}{p} = 136.8\text{N}$$

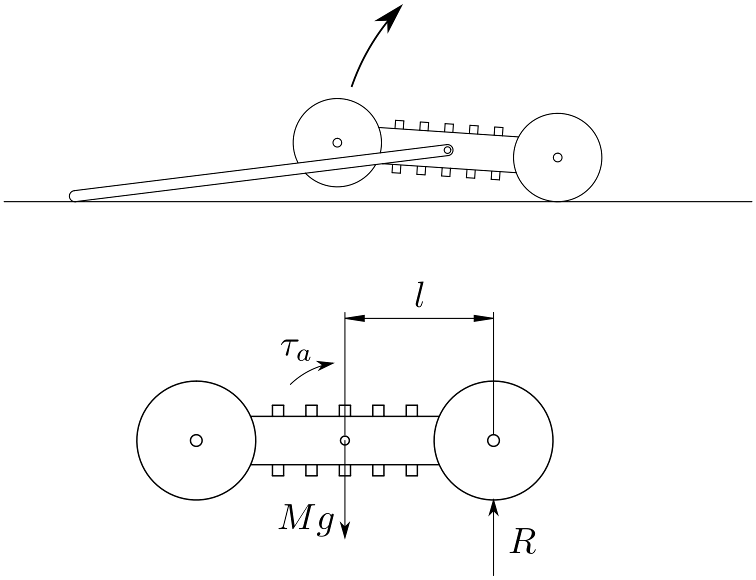


Figure 3.12: Diagram of the forces on the robot standing up from horizontal.

According to SDP-SI [29], the breaking strength of the TruMotion Timing belts is 158N for each millimetre of width.

Choosing a 6.35mm belt, we thus have a safety factor of more 7.3 against breaking from the maximum motor torque. Noting that the maximum stall torque of the motor exceeds the gearbox's own specification torque, this will likely never occur since it is electronically limited.

3.2.4.3 Arm gear motor selection

The largest arm torque occurs when the robot tries to lift itself from an horizontal position, as shown in Fig. 3.12.

From the Figure we extract:

$$R = Mg$$

$$\tau_a = R \cdot l = 5 \cdot 9.81 \cdot 0.127 = 6.23\text{Nm}$$

This torque is the minimum that is needed to rotate the arm under the worst case scenario. Ideally the arm is able to climb the robot at a rate of two steps per second. Since every complete body rotation makes the robot climb two steps, the arm must

Model	RS540	RS545	RS555
Operating v	4.5-12	5-15	5-15
Nominal v	12	12	12
No Load rpm	16800	16800	7750
No Load A	1	0.9	0.4
Stall Torque mN-m	278.8	166.7	205.9
Stall Current	42	21	15
Kt mN-m/A	6.6	7.9	13.7
Kv rpm/V	1400	1400	646
Peak efficiency %	71	79.8	68.5
Rpm peak eff.	14525	13920	6660
Torque peak eff. mN-m	43.8	34.5	33.6
Current peak eff. A	6.6	4.4	2.5
Weight g	153	176	213
Lengthmm	50	50.5	57
Diametermm	35.8	38	38

Table 3.1: Motors compatible with the P60 gearboxes.

be able to rotate at around one rotation per second under load τ_a. So at the rated torque, the motor must spin at 60rpm.

Also, during the selection of the motor and the gearbox, the diameter must not exceed 38.1mm [1.5in] including mounts otherwise they would not fit within the robot's body.

Amongst many motors found, very few had the right combination to match the above requirements. After research, a good option with many customizations is to use the Banebots P60 series of gearboxes available from [30]. Banebots actually have a number of configurations available. For all the gearboxes, three motors can be used. The motors specifications are outlined in Table 3.1.

These three motors can be matched with any of the gearbox ratios available. Since the gearbox is made by layering multiple planetary gear sets, the higher the ratio, the more weight and length the gearbox has. The gearbox options are outlined in Table 3.2.

After analyzing the motor properties with the working conditions (Figure 3.13), it was determined that the RS555 motor coupled with the 131.94 ratio gearbox would provide the best efficiency given the wide range of speeds and torque that the arm

Ratio 1:X	Weight g	Lengthmm
5	184	38.1
16	227	48.3
20	227	48.3
25.92	227	48.3
64	273	58.4
81.45	273	58.4
103.67	273	58.4
131.94	273	58.4
256	323	68.6

Table 3.2: P60 gearbox combinations available.

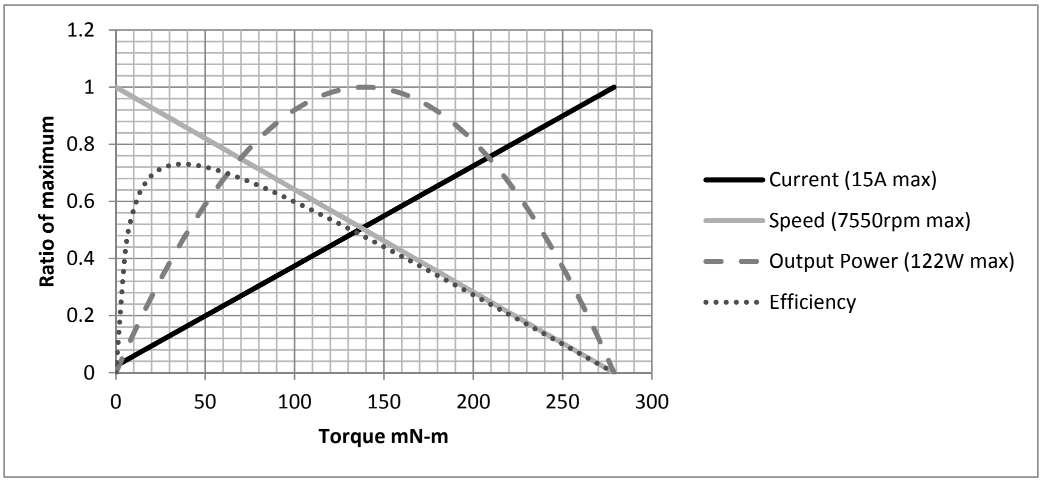


Figure 3.13: Torque, speed and efficiency curve of the RS555 motor (see efficiency data for all considered arm motors in [26]).

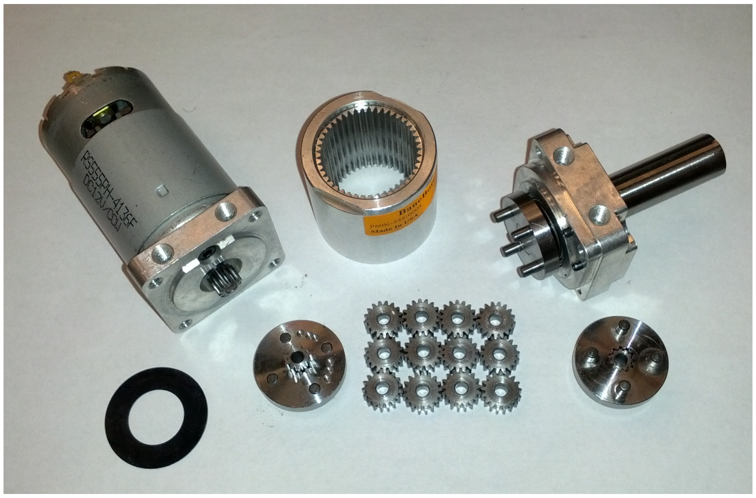


Figure 3.14: Planetary gearmotor assembly for the arm mechanism.

will use. Figure 3.14 shows the disassembled gearmotor.

Using this motor and gearbox, the motor will operate closest to its highest efficiency region most of the time, therefore increasing the battery life and reducing the motor heating.

It is important to note that the stall torque is much higher than the operating torque and tremendous torques can be generated. It will be needed to electronically limit the torque to avoid damaging the gearbox and other components.

3.2.4.4 Determination of the arm length

As the robot climbs a step, its body angle changes and as it progresses, the configuration of contact points changes. As seen in Section 3.2.3.2, there are two principal configurations that the arm can be while climbing steps.

The choice of arm length depends on the various stair dimensions. These dimensions were verified against the model for stair-climbing (Section 3.2.3.2).

In the end, an arm of 66cm (26in.) was selected. The arm design is made so that it can be extended by 5 additional inches if the need arise.

3.2.4.5 Other design considerations

Several other design considerations, such as grouser, arm drivetrain, and arm rod material selection can be found in Appendix D.

Please refer to Appendix E for the complete parts and assembly drawings for the robot.

3.2.5 Electrical design

3.2.5.1 Circuit overview

This section is an overview of the architecture of the circuit used in the robot. Figure 3.15 shows the wiring used to distribute the power to the motors and the PCB circuit. To avoid ground loops and current surges to sensitive components, the wiring is arranged into a star configuration where the junction is the first element after the battery fuse. The battery voltage is directly fed into the motor driver circuits and into the 5V voltage regulator for the low voltage components of the circuit. There is a battery connection to the main PCB but is used only to measure the voltage of the battery.

Figure 3.16 presents an overview of the PCB components and interface with external components. All the components of the PCB are interfaced with the central microcontroller that handles all measurements and control function. The complete circuit schematic is shown in Appendix I

The following sections cover some of the modules of the circuit. For more information please refer to the circuit diagram of the circuit and/or the components datasheets.

3.2.5.2 Battery selection

The batteries must be able to supply the necessary power for all the motors and the circuitry without being overloaded and without voltage drops. Here we look at the minimum specifications the battery should have. The wheel motors have a stall

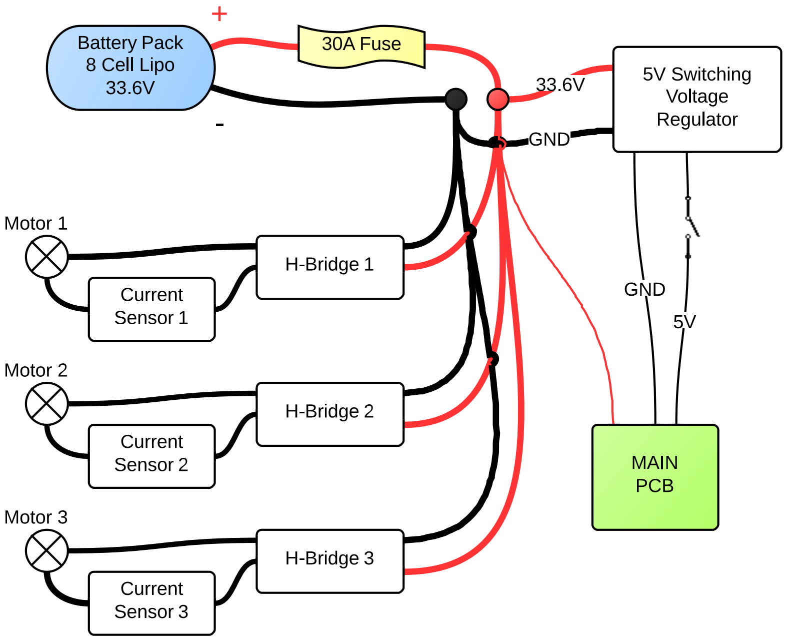


Figure 3.15: Overview of the robot's internal wiring and power distribution.

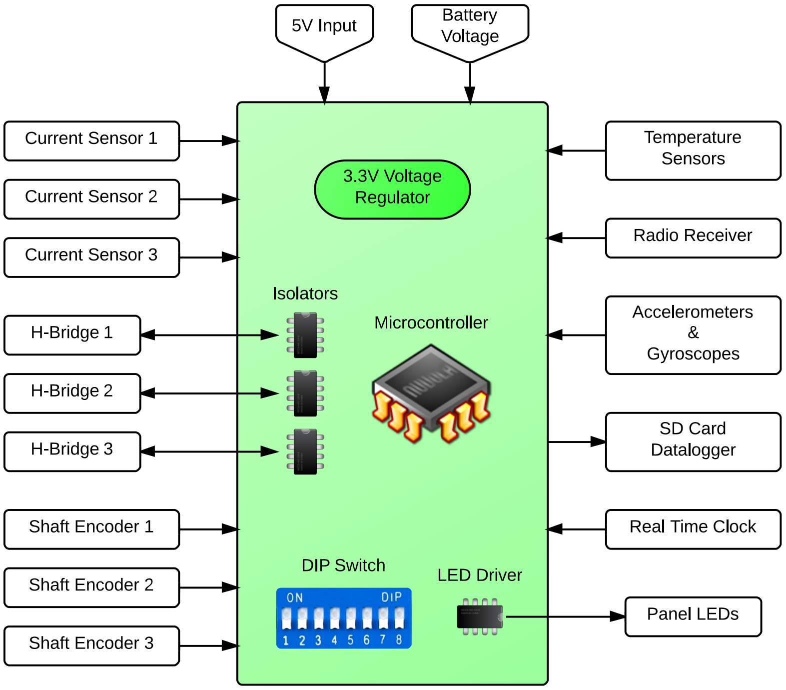


Figure 3.16: Overview of the PCB board components and connections to external modules.

current of 7.74A each. But this is an intermittent condition. Thus we could assume 10A peak for drive motors combined. Also the arm motor can have a stall current of 15A, but will never reach this condition on start up since the chain would break. Additionally, it should never happen that all three motors are functioning at their peak torque at the same time. When the arm is used to raise the robot, it will not be balancing, and while the robot is balancing, even if the arm is rotating there is no applied torque. All the other electronics are low power.

The battery voltage should be sufficient to control the drive motors, which have a nominal voltage of 30.3V. Based on the level of charge, lithium polymer batteries have cell voltages between 3.0V and 4.2V with a nominal value of 3.7V. Placing 8 battery cells in series would provide 29.6V nominal, 33.6V fully charged, and 24V fully discharged.

Assuming the whole system draws 2A continuously on average, for the robot to last one hour, the battery must have a 2000mAh capacity. Since 8 cells battery packs tend to be expensive and large, it is proposed to connect 4 packs of two cells in series to obtain the equivalent of an 8 cell pack. Each pack of two cells should each be 2000mAh for the total pack to be this capacity (since they are wired in series).

A 20A peak demand corresponds to a 10C (battery discharge unit expressed as a function of its total charge capacity) discharge capability of the battery and should not be a problem to satisfy.

The protection fuse for the circuit will be selected to trigger at 30A. To ensure battery protection even during a short circuit, a battery with a peak rating of more than 30A was chosen.

3.2.5.3 Inertial sensor requirements

- Pitch gyroscope: for balancing the robot needs a gyroscopic sensor on the pitch axis of the robot. Based on the MATLAB simulation, the maximum rate the robot rotates when recovering from a 20° disturbance is less than 8 rad/s for both the linear and nonlinear models. This corresponds to a rotational speed of 458°/s. The selection of a gyroscope sensor with a full range of ±500°/s would be desirable. Higher range sensors would reduce the resolution and would affect steady state balancing.

- Yaw gyroscope: to travel in a straight line without constantly requiring user corrections, a yaw gyroscope was also needed.
- Roll gyroscope: optionally another roll gyroscope can also be used to help the robot keep its body from tipping sideways. This stabilization was not implemented in this design.
- Accelerometers: Depending on how the control loop is implemented, the control system will need to estimate the angle of the robot. One way is to integrate the gyroscope readings, but this presents the problem of drift. To compensate this, accelerometers are usually combined with the gyroscope. Even if the control system does not use the accelerometer (estimates the angle from an observer), it is good to know the state of the robot's orientation to detect if it has fallen or to calculate its centre of gravity. Unless only the tilt angle is desired, a 3-axis accelerometer can be used.

For this robot we use the MPU-6050 from InvenSense. It is a single chip that includes a 3-axis accelerometer and gyroscope with a user selectable range.

3.2.5.4 Monitoring components

Because the robot is used for research purposes, a number of components were required to monitor its status:

- Temperature sensor: the design of the circuit is ready to accommodate DS18B20 temperature sensors. They are currently not installed.
- Current sensors: the stall current of the arm motor is 15A, and the stall current of the drive motors is 7.74A. It is expected that the electronics draw about 1A. Since we want to measure the motor current's direction current as well, a bidirectional sensor was used. For the circuits, to obtain a decent resolution

measure, a low current sensor is needed. Hall effect sensors are perfect for current measurement because they do not add a resistance in the path of the wire. This is better for high current loads because it does not add inefficiency to the circuit. The ACS711 and ACS714 sensors sold by Polulu were used.

- Voltage measurement: to measure the voltage of the battery, a simple voltage divider circuit was used. Two precision resistors were used to scale the voltage range of the battery to what the microcontroller can measure using its analog to digital converter. To protect the microcontroller, a safety factor is applied while calculating the resistor values to compensate any voltage spikes.

3.2.6 Software design

The code handles the commands from the remote control, and proper software has been written to transition from 4-wheel mode to balancing mode. A built-in memory records all sensor readings and system state for later analysis. The complete firmware mostly runs on interrupts. This enables a fast performance with consistent timing.

A special routine was also coded to allow automatic stand-up and stand-down sequence of the robot. It works by monitoring the current in the arm motor to detect when the arm is pushing against the floor or not, and uses the accelerometer to know the orientation of the body. During stand-up, a drop in the arm current means the robot is upright and is about to pass its equilibrium position. At this point the wheel motors give a short jolt of power to raise the arm away from the floor quickly and the balance control is activated. To stand down smoothly, a similar sequence is performed, but the balancing will deactivate when the arm has started touching the floor. The difficulty to stand-down smoothly lies with the fact that if the balancing routine stops too early the robot may fall on the other side. Because of this, the balancing continues until the robot actually starts pushing against the floor with the arm.

For details about the implementation please refer to the firmware code [26].

3.3 Robot fabrication and assembly

Many of the robot components had to be custom fabricated. For example, the wheel bearing blocks shown in Fig. 3.17.

Figure 3.18 shows all the robot parts prior to assembly. The body panels and some structural members were laser-cut and machined, the PCB was assembled, and some components had to be modified to fit the design.

The assembly started by combining the drivetrain components with the frame. As seen in Fig. 3.19 one motor powers the two wheels on the same side. The power is transferred to the other wheel using a belt.

The next step was to add the arm mechanism. It consists of a central shaft that extends to both sides of the robot and the shaft is driven by the arm's gearmotor trough a chain. Then all the electronics, wires, and batteries were added. Figure 3.20 shows the completed robot without its panels. The completed robot was already shown in Fig. 3.1.

3.4 Results

3.4.1 Balancing results

Figure 3.21 show the results for the robot keeping balance. The control loop is able to keep the robot upright within a $\pm 5^\circ$ error. This is very low and the balancing motion is smooth and barely visible (see videos of the robot while balancing in [26]).

3.4.2 Stair-climbing results

To validate the theory of Section 3.2.3.2, the experimental results of the robot climbing a complete step were obtained. Figure 3.22 shows a sequence of photographs taken during the experiment.

To avoid dynamic effects, the experiment was conducted at a slow rate where the arm was rotating at a speed of about one degree per second. While the experiment was performed, the robot's internal circuitry recorded a number of parameters into

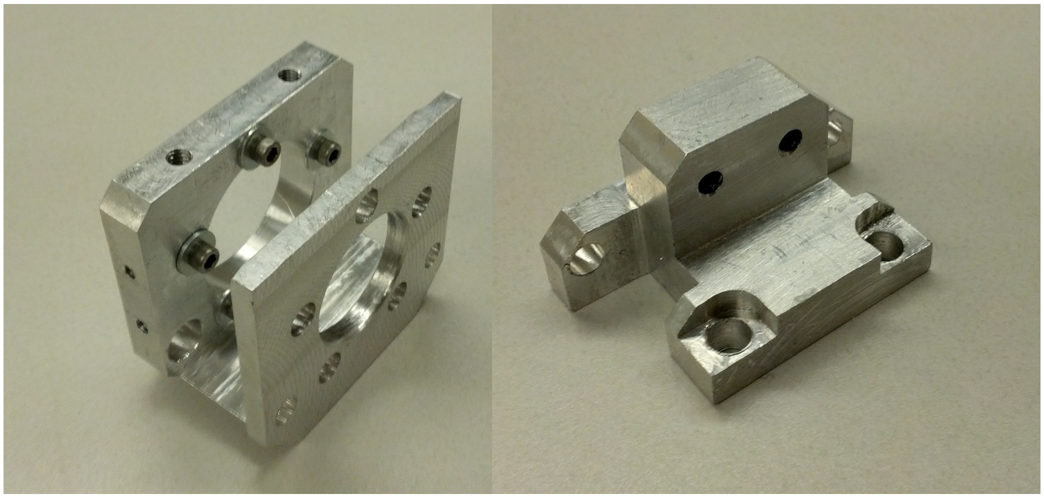


Figure 3.17: Custom bearing blocks for the robot.

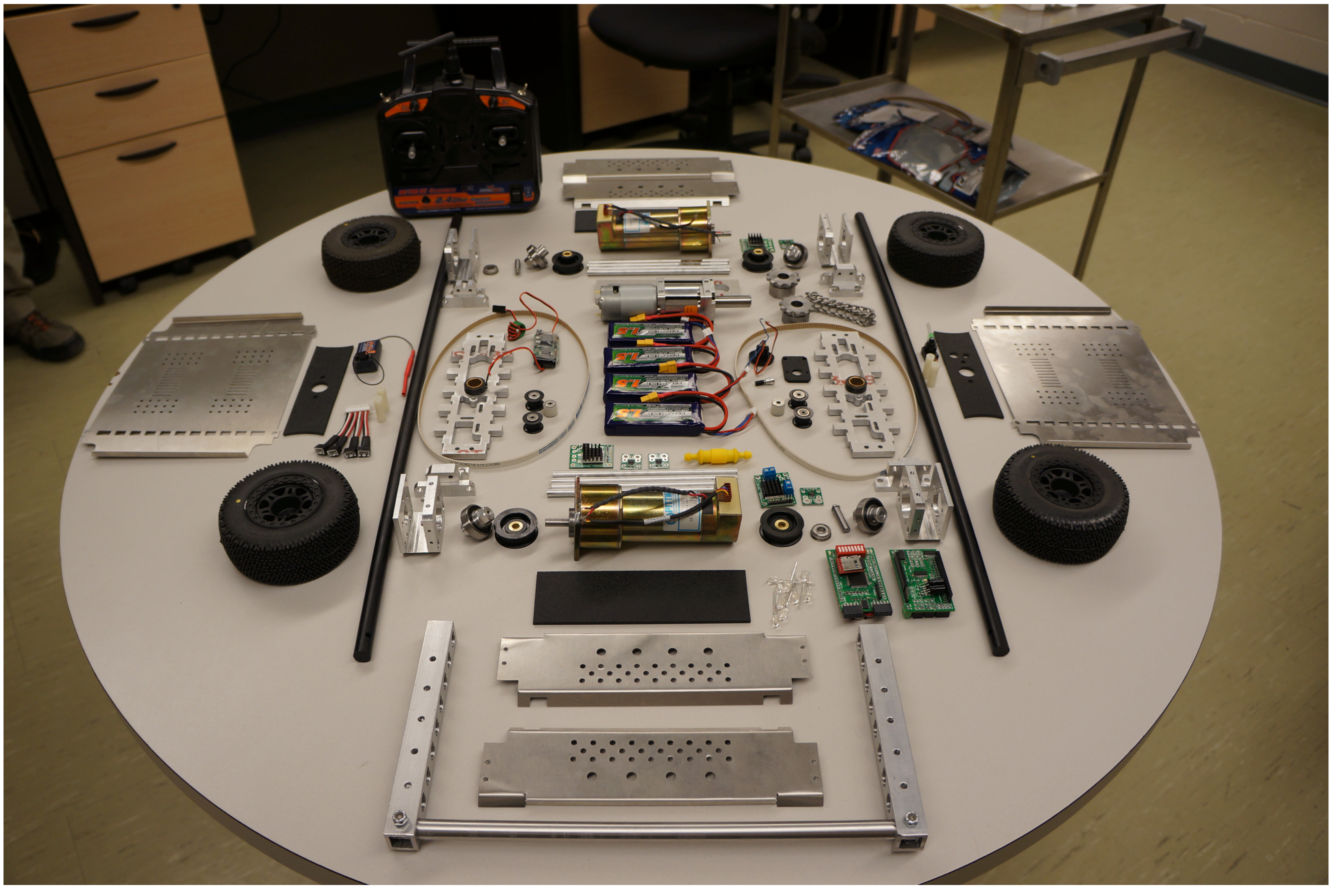


Figure 3.18: Exploded view of the robot parts after fabrication.

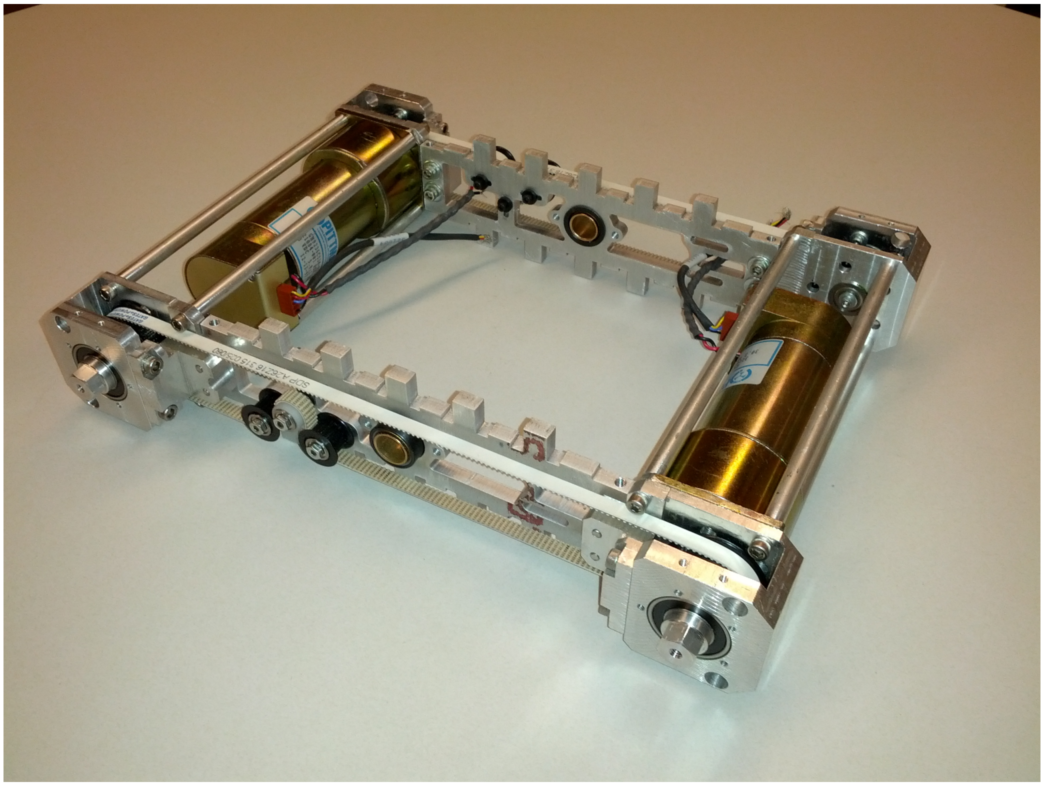


Figure 3.19: View of the frame assembly.

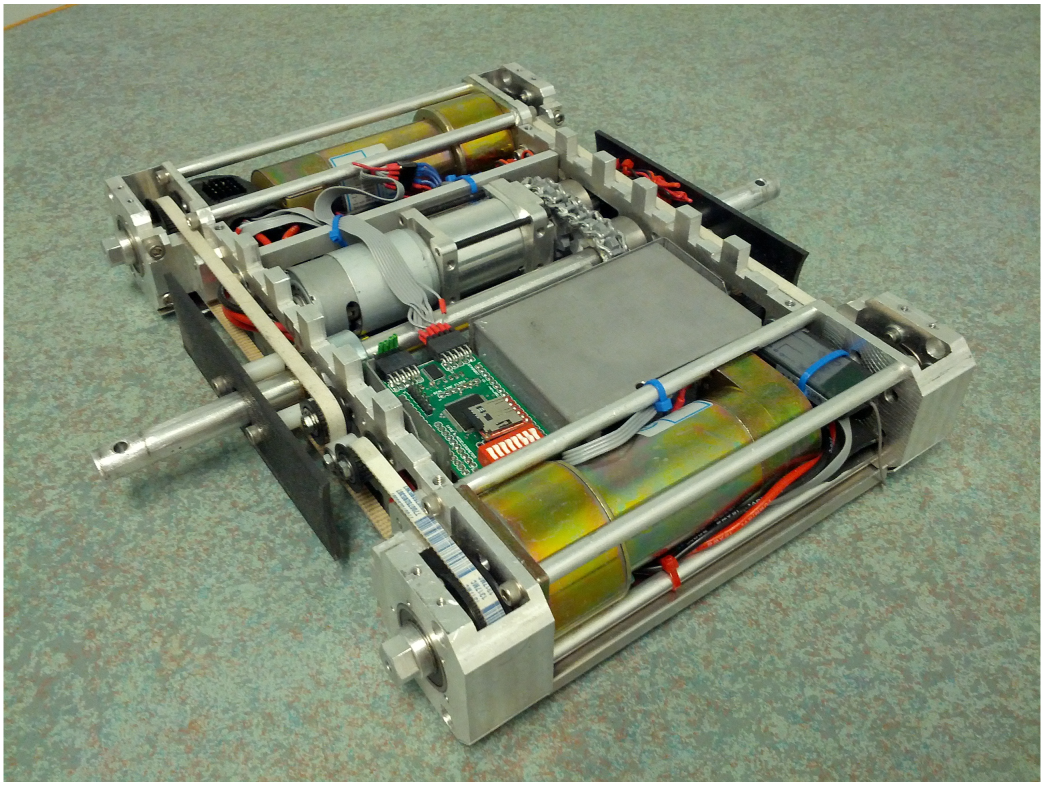


Figure 3.20: View of the complete robot internal assembly.

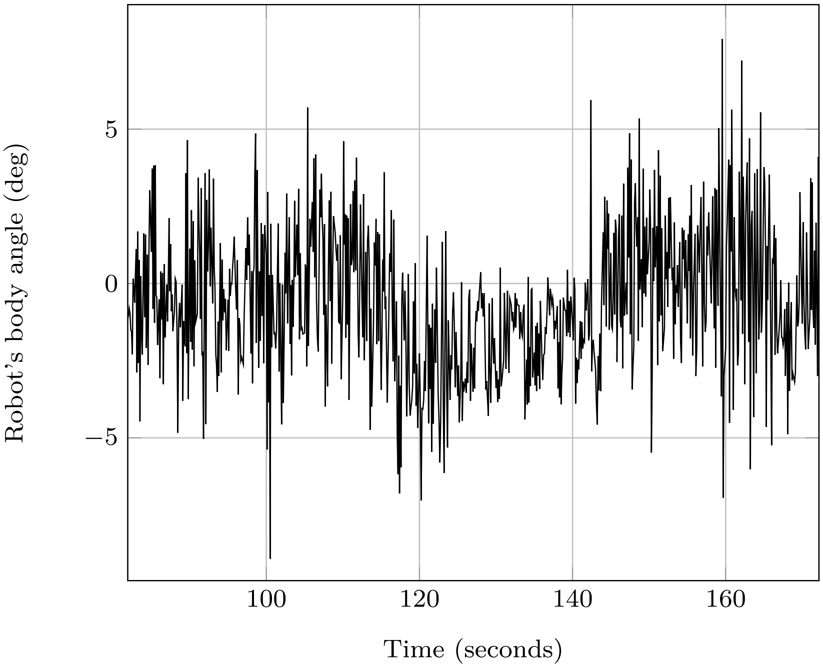


Figure 3.21: Steady robot balancing error.

a memory card which could then be analyzed offline. For this experiment, the robot logged the following parameters:

- Current time: time was synchronized with the camera's internal time. This allows matching the photographs with the readings in the memory card.
- Inertial measurements: the gyroscope and accelerometer readings were recorded to ensure the events on the pictures were synchronized.
- Motor currents: drive motors and arm motor.
- Battery voltage.

The arm angle θ_a, body angle θ_b, and arm contact length $l_c$were measured using pixel coordinates from the photographs. Whether the robot was in case A or case B was also determined from the picture. The arm an body angles could have also been determined by the logged measurements in the memory card, but were less accurate then measuring on the picture. Referring to Fig. 3.22, the contact mode is as follow:

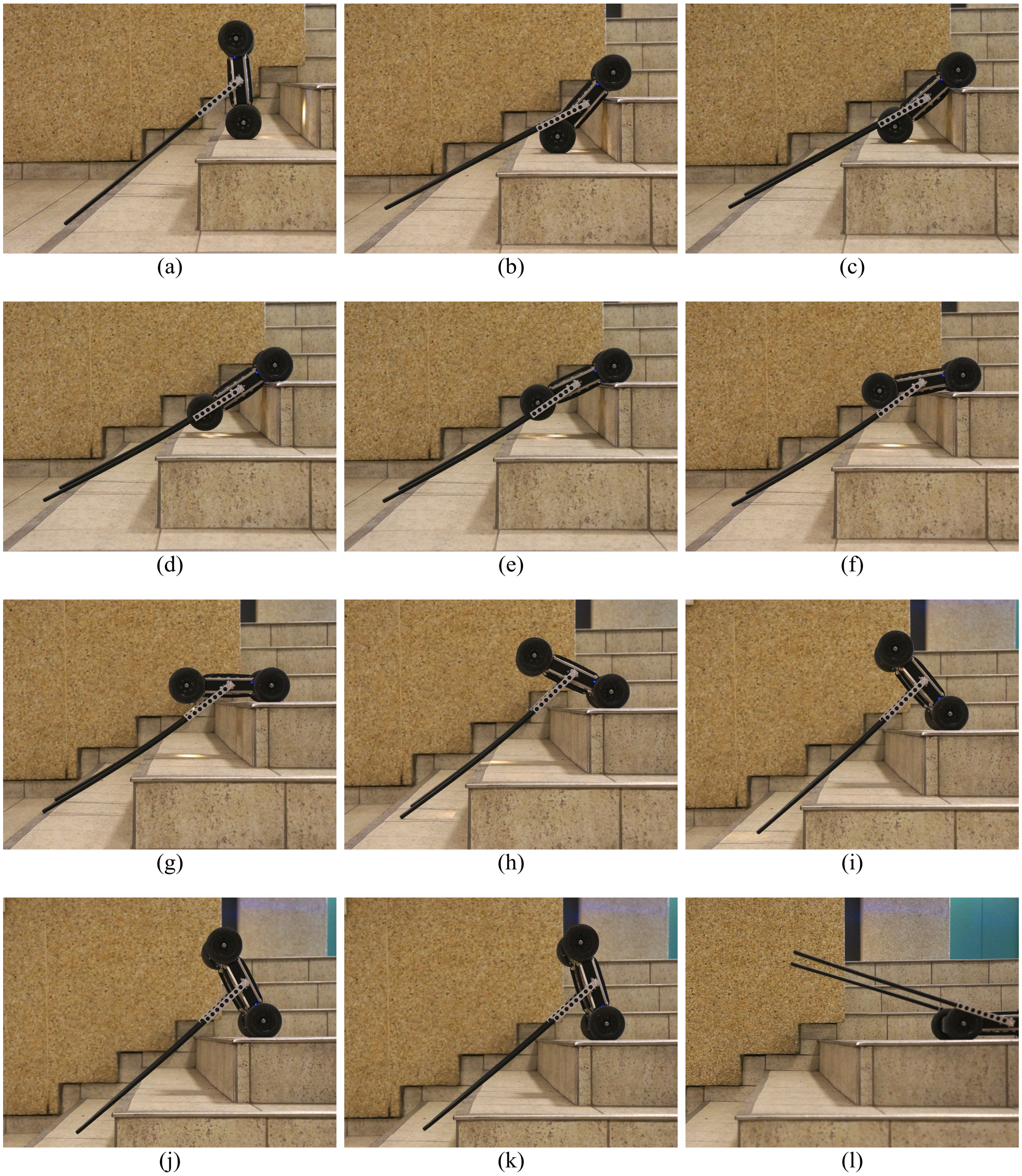

(a) (b) (c) (d) (e) (f) (g) (h) (i) (j) (k) (l)

Figure 3.22: Sequenced photographs of the robot climbing a single step. The robot starts with its body in a vertical position (a) and finishes at the same position (upside down) on the next step after (k). The same sequence would be repeated for more steps.

- (a): Just before being vertical, the robot is finishing its previous cycle.
- (b): The robot has fallen down with body against step. The arm is raised.
- (c): Arm pushes lightly on the bottom step. Body slips until the wheel grabs.
- (d) to (f): Case B: the bottom wheel has lifted from the step.
- (g) to (h): Case A: the arm is now pushing against the run of the step
- (i) to (k): Case B.
- (l): The robot passes its equilibrium point and falls (ready for another step).

Once θ_a, θ_b, l_c, and the mode were determined using the photographs, the theoretical torque values were calculated using Eqs. (3.18–3.21). The arm coefficient of friction μ_a was assumed to be 0.25 (arm is made of acetal). The logged motor currents were converted to torques using the motor's models. For the arm drivetrain, an estimated efficiency of 0.8 was assumed to account for the triple-stage greased planetary gearbox and the chain drive in the arm mechanism. Figures 3.23 and 3.24 show the results for the arm and wheel torques. The annotations refer to the pictures in Fig. 3.22.

For both figures, we see that the trend is closely followed between both theoretical and experimental data. The torque values are about zero when the body is vertical. This is expected because the centre of gravity of the robot is directly above its wheels and the arm does not need to apply any torque. There is discrepancy between the model and the real data for point (b) for the arm torque. It is because the model assumes the bottom wheel is not touching the ground while in reality it is still in contact with it, thus the lower arm torque required. For the wheel torque, point (c) is peaking to almost 1Nm, which can happen if the robot tries to move forward but is blocked by the body in contact with the riser (see Fig. 3.22 (c)).

There are a number of sources of errors. For one, the arm gearbox efficiency and the arm coefficient of friction were assumed based on literature examples but were not explicitly available. The experiment was conducted at very slow speed thus dynamic effects were neglected. Angle and distance measurements on the photographs are subjected to resolution, visual, and lens distortion errors. The current sensors have an error of ±1.5% plus the error due to the digital-to-analog converter resolution and noise. There are also unmodeled physical properties such as friction, arm

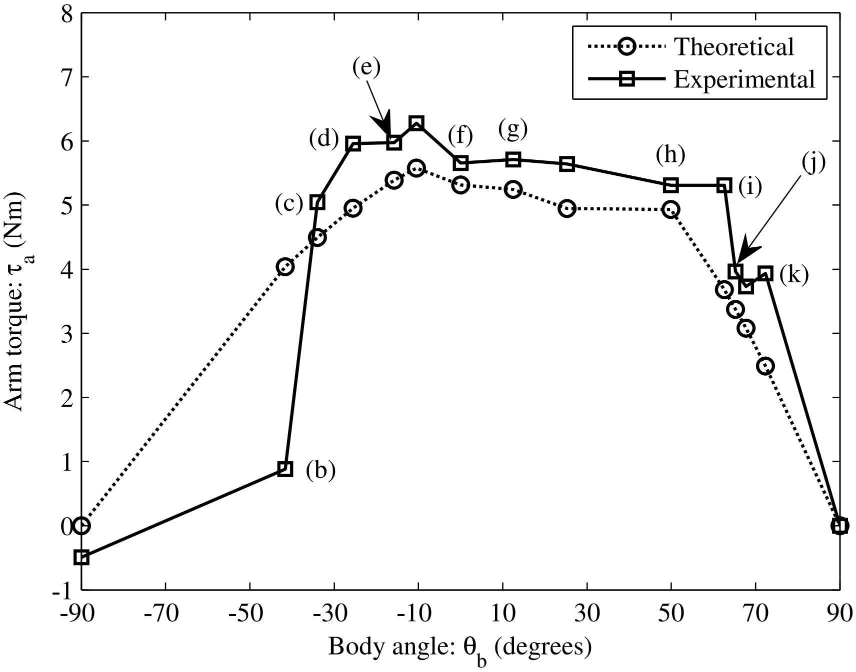


Figure 3.23: Torque exerted on the arm of the robot during the climbing sequence shown in Fig. 3.22.

bending, wheel deformation, motor dead-band, and backlash. Nonetheless with all these sources of error, it is clear that the model closely followed the experimental data.

3.4.3 Obstacle climbing and other manoeuvres

In addition to stairs, we found that this robot was capable of traversing other obstacles. The field tests that were performed show the robot is in fact very versatile and can navigate above obstacles that only larger robots can usually do.

The robot can navigate on slopes and over random obstacles. Several examples are shown in Fig. 3.25. A video of the robot in action is available in [26].

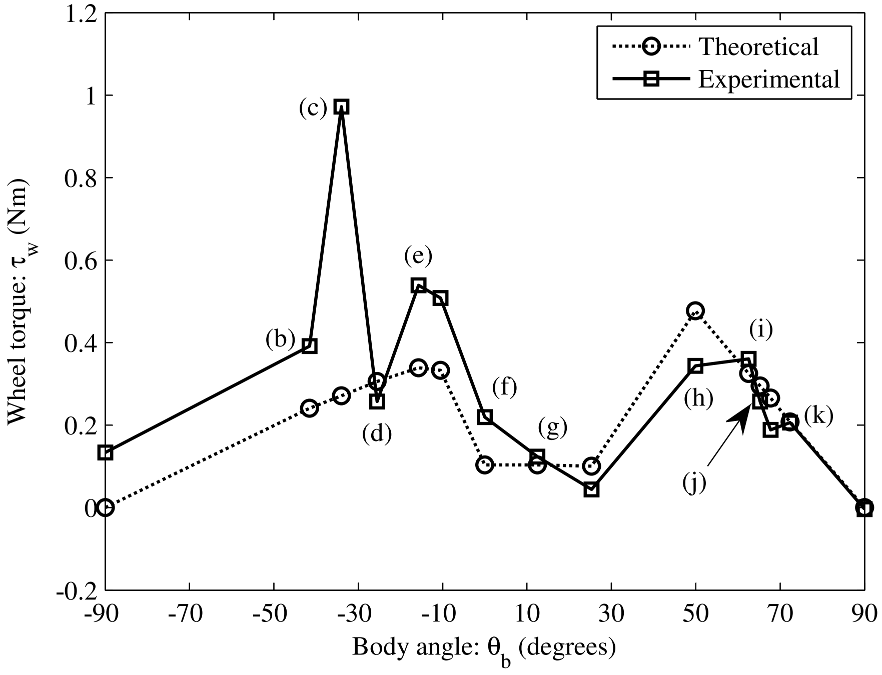


Figure 3.24: Torque exerted on the wheels of the robot during the climbing sequence shown in Fig. 3.22.

(a)

(b)

(c)

(d)

Figure 3.25: (a): Rolling on rough ground. (b): Climbing a large number of concrete bricks. (c): Balancing on a speed bump. (d): Climbing onto an elevated wood crate.

3.5 Discussion

3.5.1 Results summary

The design proved very successful at stair-climbing and self-balancing.

The robot was able to navigate at high velocity on four wheels. By navigating on two wheels the robot could also travel very efficiently by steering without friction. Since a dynamic control loop was needed to balance, the robot could not travel as quickly compared to navigating on four wheels, but it still had a decent speed of motion (see videos in [26]). A great advantage of two-wheel travel is the robot is taller (useful if adding a camera) and can also steer very easily on the spot.

It was shown that a very simple joint mechanism with a carefully designed layout gives the robot a lot of field capability. The robot can transition automatically from 4-wheel to balancing modes just using the arms, and can also use its arms to climb stairs. It was found during field trials that the robot was able to overcome many types of obstacles that would be very difficult even for larger robots designed for the task.

The current prototype has limitations. First, it is fairly heavy for its size (5kg) . The frame was made very strong in case the robot fell down stairs while testing. The wheel hubs are constructed with solid aluminium blocks that are held in an all aluminium frame. If the robot was to fall from a height the parts that would most likely break are the plastic wheels. Another element contributing to the weight are the large DC drive motors which could be replaced by smaller, lighter, and more powerful brushless motors.

Another limitation is that the focus of this work was mostly done on the mechanical aspect and unique capabilities of the novel layout. The robot is controlled with a remote control and does not have any type of autonomous functions (apart from balancing automatically and the stand-up sequence).

3.5.2 Future work

Being a research project, the robot is far from being complete to be sold as a commercial robot or to be useful for surveillance/tactical operations. At this point, the robot is fully functional and would be ready for continuing research and development to add more capabilities to the robot. Future plans on the robot would be:

-Obtain more data and results on the following aspects: comparison of power consumption on 4-wheel and balancing modes, test battery life in various conditions, make more field tests on different grounds, make a deeper analysis of a complete stair-climbing sequence to design an autonomous climbing sequence.

-Add autonomous stair-climbing. This could be achieved by either proximity sensors or by using a laser line coupled with a camera, the latter having more chances of working with open riser steps (stairs that do not have vertical faces).

-Redesign the robot to be lighter by using smaller motors, replacing some metal parts by plastic 3D printed parts, and reducing the battery capacity. Most of the internal volume is occupied by batteries and the motors. Reducing their size would give additional space to carry a payload.

-Add cameras to see where the robot is going. More then one camera would need to be installed and custom software would send the correct camera images depending on the orientation of the body.

-Depending on the application, the robot could have a number of other additions, such as custom sensors, advanced wireless communication or autonomous navigation.

-Make the enclosure weatherproof. The current enclosure is mainly to protect the internal components from debris and external objects, but weatherproofing the complete robot would make it usable in real outdoor conditions. With minor modifications the robot could also float and propel itself across a mass of water.

Chapter 4

Hybrid gyroscopic stabilization system

From this point in the thesis we will analyze the effect of adding a mechanical gyroscope to an inverted pendulum, for use in self-balancing robots. This is a different project than the development of the stair-climbing robot covered in the previous chapters. A mechanical gyroscope generates precession moments when constrained to rotate in an axis perpendicular to its spinning axis. This precession moment is applied by the gyroscope's disk on a third axis, perpendicular to both the spinning and rotation axes. The gyroscopic disc is mounted to the rod of an inverted pendulum using a gimbal. This gimbal is free to rotate relative to an axis perpendicular to the rotation axis of the rod, allowing the creation of a precession moment that can influence the rod's angle.

The chapter is organized as follows. Section 4.1 covers the theory and simulation of the hybrid balancing system and Section 4.2 outlines the design, fabrication, and experimental testing of this device.

4.1 Theory and simulation

4.1.1 Modelling of an inverted pendulum

Although the following derivation is also available in [31], the complete derivation is shown here as it will help clarify the gyroscopic stabilization model developed in this thesis. Figure 4.1 shows the physical representation of the robot modelled as an inverted pendulum, where u is the control force applied by an external actuator to balance the robot. Figure 4.2 shows the free body diagrams of the rod and the cart that make the two components of the inverted pendulum.

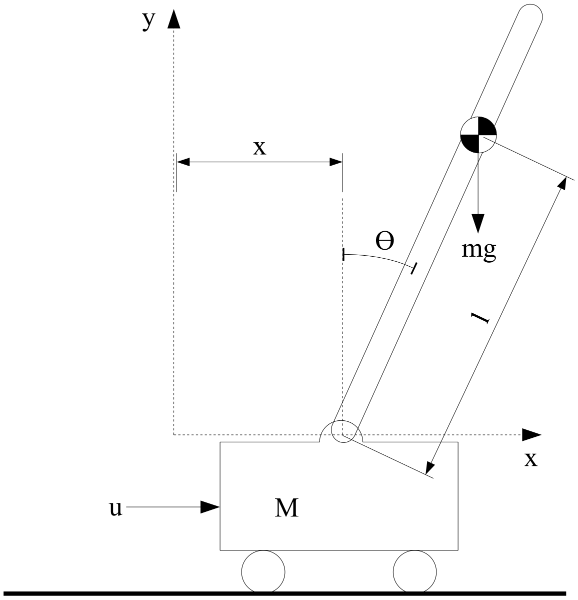


Figure 4.1: Model of the inverted pendulum

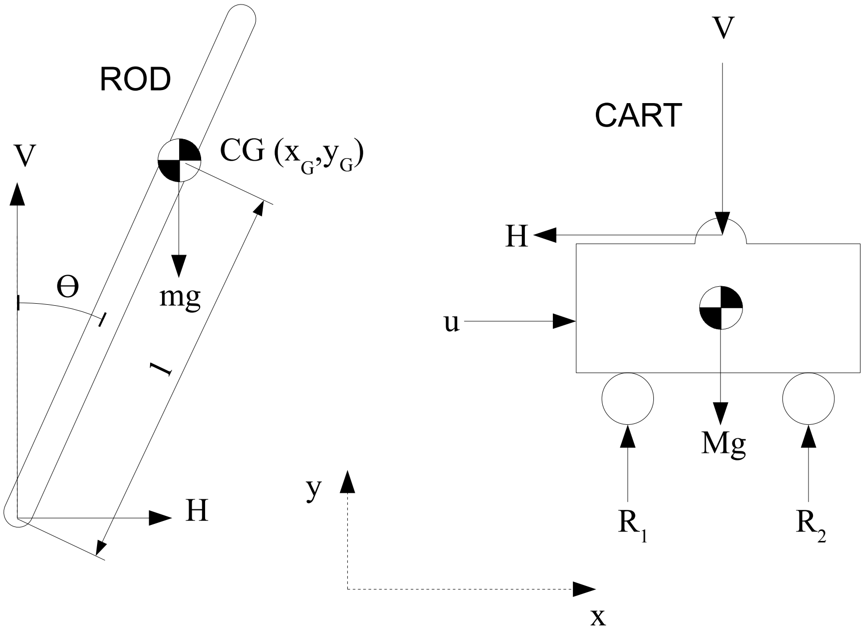


Figure 4.2: Free body diagram of the rod and cart of the inverted pendulum

Forces on the rod

We can express the coordinates of the centre of gravity of the rod relative to its pivot as follow:

$$x_G = x + l \sin\theta$$

$$y_G = l \cos\theta$$

By taking the sum of moments about the centre of gravity of the rod, we obtain:

$$\sum M_{CG} = I\ddot{\theta}$$

$$Vl \sin\theta - Hl \cos\theta = I\ddot{\theta} \tag{4.1}$$

and taking the sum of forces along x and y, we obtain:

$$\sum F_x = ma_{Gx} H = m\ddot{x}_G$$

$$H = m\ddot{x}_G$$

$$H = m\frac{d^2}{dt^2}(x + l\sin\theta) \tag{4.2}$$

$$\sum F_y = ma_{Gy} \tag{4.3}$$

$$V - mg = m\ddot{y}_G$$

$$V = \frac{d^2}{dt^2}(l\cos\theta) + mg$$

Since the pendulum should be in a balanced position, we linearize around equilibrium by assuming θ is close to zero. Thus the following assumptions:

$$\sin\theta \approx \theta$$

$$\cos\theta \approx 1$$

Eqs. (4.1), (4.2), and (4.3) can be simplified to:

$$H = m\ddot{x} + ml\ddot{\theta} \tag{4.4}$$

$$V = mg \tag{4.5}$$

$$Vl\theta = I\ddot{\theta} + Hl \tag{4.6}$$

Solving Eqs. (4.4), (4.5), and (4.6) for H and V, we obtain:

$$(I + ml^2)\ddot{\theta} + ml\ddot{x} = mgl\theta \tag{4.7}$$

Note that the mass of the cart M is not directly involved here. However to achieve a desired cart acceleration $\ddot{x}$, a certain amount of power is needed that is dependant on the mass of the cart.

In a self-balancing robot, the cart is replaced by a wheel. The effect is that the mass M would be the equivalent rotational mass of the motor, gears, and the wheel.

If all the mass of the rod would be concentrated on a single point, I would become null. The model of the system would still be second order linear equation.

Forces on the cart

Taking the sum of the forces on the cart along x, we have:

$$\begin{aligned} \sum F_x = Ma_x \Leftrightarrow u - H = M\ddot{x} \\ u = (M + m)\ddot{x} + ml\ddot{\theta} \end{aligned} \tag{4.8}$$

The force u is a linear force that is applied from an actuator. In a self-balancing mobile robot, this force originates from the wheels that are driven using an electric motor. During the simulation $\ddot{x}$ is calculated from the result of the control loop (desired acceleration) while $\ddot{\theta}$ is obtained from rate and acceleration sensors.

4.1.2 Modelling of the inverted pendulum with a gyroscopic gimbal

Figure 4.3 shows the representation of the inverted pendulum with a gyroscopic gimbal model. Table 4.1 shows a list of symbols and their values used in the simulations.

The system consists of an inverted pendulum configuration with the addition of a gyroscopic assembly mounted on top of the rod. The gyroscopic assembly is comprised of both a rotating disk and a gimbal, to hold the rotating disk. The gimbal is free to rotate with respect to an axis perpendicular to the rod's axis of rotation.

Table 4.1: Symbol definitions and simulation system parameters.

Symbol	Value	Unit	Description
b	0.8	$\mathrm{N\,m\,(rad\,s)^{-1}}$	Damping coefficient of the gimbal
F	N/A	N	Actuation force on the cart
F_{lim}	1	N	Actuator force saturation limit
g	9.81	$\mathrm{m\,s^{-2}}$	Gravitational force
J_{gX}	0.001	$\mathrm{kg\,m^2}$	Moment of inertia of the gimbal along X (Fig. 4.3)
J_{gY}	0.003	$\mathrm{kg\,m^2}$	Moment of inertia of the gimbal along Y (Fig. 4.3)
J_{gZ}	0.003	$\mathrm{kg\,m^2}$	Moment of inertia of the gimbal along Z (Fig. 4.3)
J_{rz}	0.001	$\mathrm{kg\,m^2}$	Moment of inertia of the rod along z (Fig. 4.3)
J_w	0.01	$\mathrm{kg\,m^2}$	Moment of inertia of the wheel along its rotation axis
J_w'	0.005	$\mathrm{kg\,m^2}$	Moment of inertia of the wheel perpendicular to its rotation axis
k	5	$\mathrm{N\,m\,rad^{-1}}$	Spring coefficient of the gimbal
l	0.6	m	Distance as shown in Fig. 4.6
m_{c}	0.1	kg	Mass of the cart
m_{g}	0.05	kg	Mass of the gimbal
m_{r}	0.2	kg	Mass of the rod
m_{w}	0.2	kg	Mass of the gyroscopic wheel
ϕ	N/A	rad	Angle of the gimbal
ψ	N/A	rad	Angle of the wheel
$\dot{\psi}_{rpm}$	2000	rpm	Rotation speed of the wheel
τ	N/A	N m	Actuation torque on the gimbal
τ_{lim}	2	N m	Actuator torque saturation limit
θ	N/A	rad	Angle of the rod
x	N/A	m	Position of the cart

Initial assumptions

To simplify the model, we assume that the static mass of the wheel, gimbal, and rod is lumped into the total mass of the rod, m. That means the gyroscopic assembly has zero mass. This assumption will not affect the results since the centre of gravity of the gyroscopic assembly with respect to the rod is always at the same location. The moments of inertia of the wheel, gimbal, and rod are however non-zero and are taken into account.

It is assumed the cart of the inverted pendulum is constrained to move only along the x axis and cannot rotate other then the angle θ as shown in Figure 4.3. The system is assumed to be without friction (no air friction on the wheel, and no bearing friction on the joints).

Coordinate system

Before calculating the forces for each component, we must derive a few relations for expressing the rotation speed of different elements of the system. The XYZ coordinate system is fixed orientation with respect to the gimbal, where Y is collinear with the wheel's axis of rotation.

The gyroscopic disc rotates at a speed of $\dot{\psi}$ as shown in Fig. 4.4. The absolute rotation speed of the wheel with respect to the stars must be calculated in order to derive the resulting gyroscopic moments. The absolute speed $\mathbf{\Omega}_{XYZ}$ of the XYZ coordinate system is given by:

$$\mathbf{\Omega}_{XYZ} = \dot{\phi}\vec{I} + \dot{\theta}\vec{k}$$

Thus we have the absolute rotation speed of the wheel as:

$$\boldsymbol{\omega}_{wheel} = \dot{\psi}\vec{J} + \dot{\phi}\vec{I} + \dot{\theta}\vec{k}$$

but $\vec{k}$ is not orthogonal to $\vec{I}$ and $\vec{J}$ and must be decomposed into (from Fig. 4.3):

$$\vec{k} = \cos\phi\vec{K} + \sin\phi\vec{J}$$

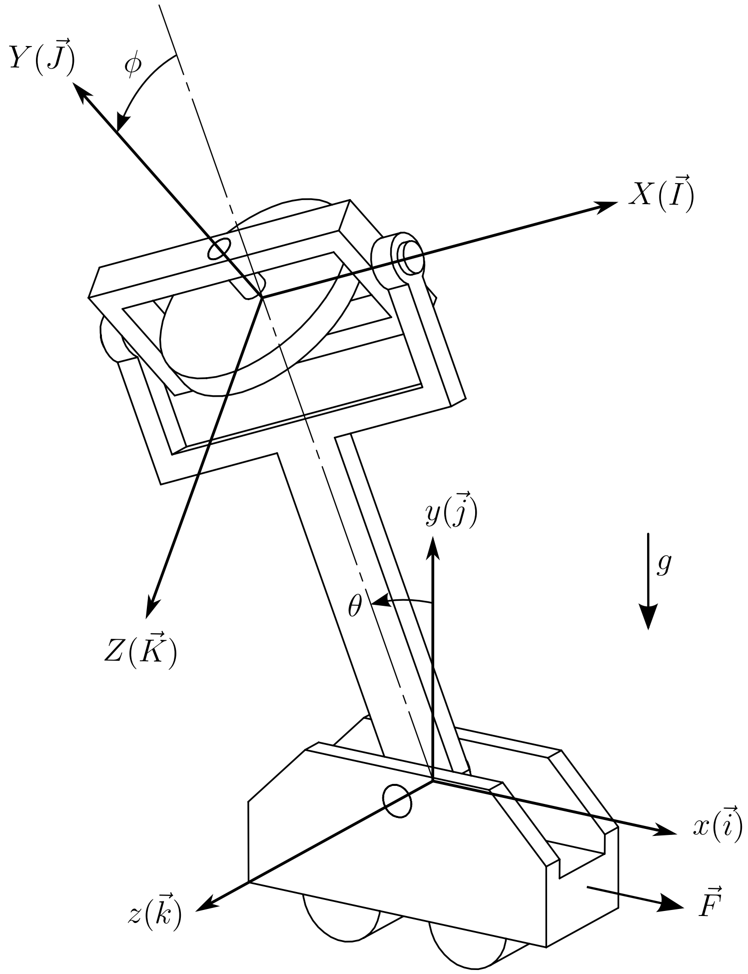


Figure 4.3: Coordinate systems and control force on the inverted pendulum equipped with a mechanical gyroscope

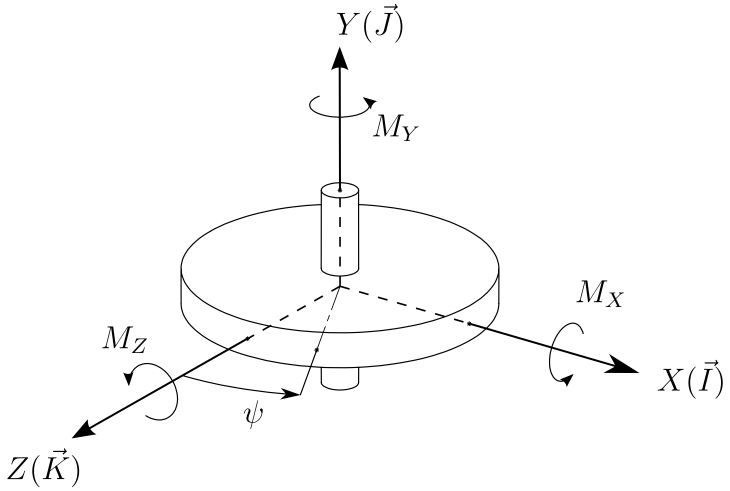


Figure 4.4: Moments acting on the gyroscopic wheel used to generate a gyroscopic precession moment

which can be substituted into the previous equations to obtain the absolute speed of the wheel expressed relative to the current position of XYZ:

$$\boldsymbol{\Omega}_{XYZ} = \dot{\phi}\vec{I} + \dot{\theta}\sin\phi\vec{J} + \dot{\theta}\cos\phi\vec{K} \tag{4.9}$$

$$\boldsymbol{\omega}_{wheel} = \dot{\phi}\vec{I} + (\dot{\psi} + \dot{\theta}\sin\phi)\vec{J} + \dot{\theta}\cos\phi\vec{K} \tag{4.10}$$

Gyroscopic wheel

We will begin by analyzing the moments acting on the gyroscopic wheel. Figure 4.4 shows the free body diagram of the rotating wheel.

By taking Eqs. (18.1 and 18.2) from [32] reproduced below for convenience:

$$\sum \boldsymbol{F} = m\mathbf{a} \tag{4.11}$$

$$\sum \boldsymbol{M}_G = \dot{\boldsymbol{H}}_G \tag{4.12}$$

where: $\boldsymbol{a}$ is the acceleration vector of the centre of gravity of the body,
m is the mass of the body,
$\boldsymbol{H}_G$ is the angular momentum stored in the object, and
$\dot{\boldsymbol{H}}_G$ is the change of momentum.

These equations state that the acceleration of the centre of gravity of the wheel is the sum of forces applied to the body and the change of angular momentum is the sum of moments applied to the body. Since we considered in the assumptions that the wheel has no mass, we can ignore Eq. (4.11). Because the momentum of the wheel is complex to express as a function of the xyz coordinate system, $\dot{\boldsymbol{H}}_G$ can be expressed from the rotating referential XYZ by using Eq. (18.22) of [32]:

$$\dot{\boldsymbol{H}}_G = (\dot{\boldsymbol{H}}_G)_{XYZ} + \boldsymbol{\Omega}_{XYZ} \times \boldsymbol{H}_G \tag{4.13}$$

where: $(\dot{\boldsymbol{H}})_{XYZ}$ is the rate of change of momentum of the wheel relative to the rotating coordinate system.

The use of Eq. (4.13) allows us to use only the principal moments of inertia of the wheel because XYZ coincides with them. The momentum stored in the wheel can be calculated following Eq. (18.10 [32]) as:

$$\boldsymbol{H}_G = J'_w\omega_{wheelX}\vec{I} + J_w\omega_{wheelY}\vec{J} + J'_w\omega_{wheelZ}\vec{K} \tag{4.14}$$

where: J_w is the moment of inertia along the axis of rotation of the wheel,
J'_w is the moment of inertia of the wheel relative to its transverse axes (orthogonal to the rotation axis),
ω_{wheelX}, ω_{wheelY}, and ω_{wheelZ} are the components of $\boldsymbol{\omega}_{wheel}$ along $\vec{I}$, $\vec{J}$, $\vec{K}$ respectively.

which gives,

$$\boldsymbol{H}_G = J'_w\dot{\phi}\vec{I} + J_w(\dot{\psi} + \dot{\theta}\sin\phi)\vec{J} + J'_w\dot{\theta}\cos\phi\vec{K} \tag{4.15}$$

Substituting Eqs.(4.9) and (4.15) into Eq. (4.13), then into Eq. (4.12), we obtain:

$$\begin{aligned}\sum \boldsymbol{M}_G =&[J'_w(\ddot{\phi} + \dot{\theta}^2\cos\phi\sin\phi) - J_w\dot{\theta}\cos\phi(\dot{\psi} + \dot{\theta}\sin\phi)]\vec{I}\\ &+[J_w(\ddot{\psi} + \ddot{\theta}\sin\phi + \dot{\phi}\dot{\theta}\cos\phi)]\vec{J}\\ &+[J'_w(\ddot{\theta}\cos\phi - 2\dot{\theta}\dot{\phi}\sin\phi) + J_w\dot{\phi}(\dot{\psi} + \dot{\theta}\sin\phi)]\vec{K}\end{aligned}$$

For the study of the balance control, the wheel speed can be assumed constant ($\ddot{\psi} = 0$) and we can write:

$$M_X = J'_w(\ddot{\phi} + \dot{\theta}^2\cos\phi\sin\phi) - H\dot{\theta}\cos\phi - J_w\dot{\theta}^2\cos\phi\sin\phi \tag{4.16}$$

$$M_Y = J_w(\ddot{\theta}\sin\phi + \dot{\phi}\dot{\theta}\cos\phi) \tag{4.17}$$

$$M_Z = J'_w(\ddot{\theta}\cos\phi - 2\dot{\theta}\dot{\phi}\sin\phi) + H\dot{\phi} + J_w\dot{\phi}\dot{\theta}\sin\phi \tag{4.18}$$

where: $H = J_w\dot{\psi}$ is the amount of momentum stored in the wheel that induces the gyroscopic effect.

This set of three differential equations represent the gyroscopic torques generated by the wheel when it is rotated. Even if considered constant in this analysis, it is worth noting that the magnitude and sign of the gyroscopic effect H can be varied by modifying the rotation speed and direction of the wheel. If $\dot{\psi}$ is specified in rpm, H is calculated as follow:

$$H = \frac{2\pi\dot{\psi}_{rpm}J_w}{60}$$

Gimbal

The gimbal (Figure 4.5) has the purpose to carry the gyroscopic wheel, and constrain its rotation to a specific axis.

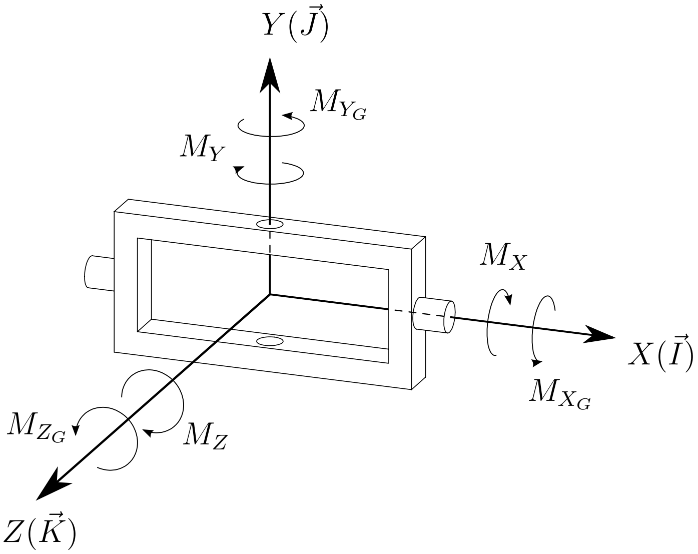


Figure 4.5: Moments acting on the gimbal holding the gyroscopic wheel.

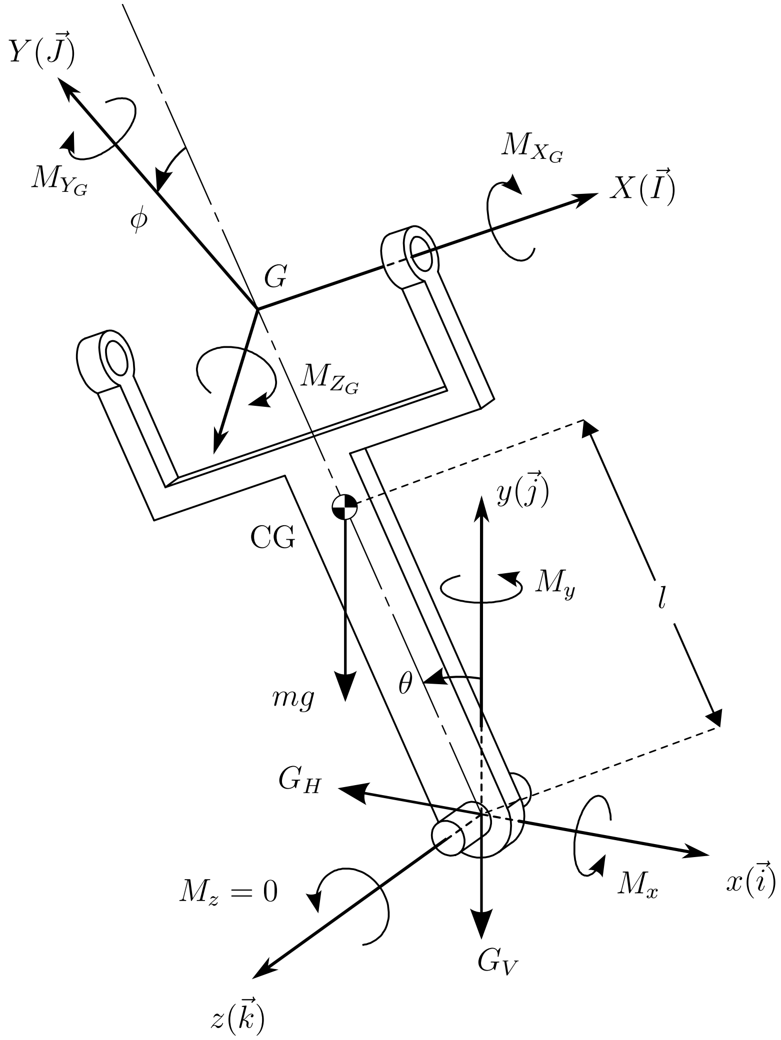


Figure 4.6: Forces acting on the rod

The joint along X, is the one that allows rotation of the gimbal relative to the rod. For the analysis, it is assumed that this joint is equipped with a spring, damper, and torque actuator in order to control the amount of torque on axis X between the gimbal and the rod. Thus we define:

$$M_{X_G} = \tau - k\phi - b\dot{\phi} \tag{4.19}$$

where: τ is the torque originating from an actuator between the gimbal and the rod,
k is the spring coefficient, and
b is the dampening coefficient.

The spring, damper and actuator can be excluded by setting their coefficients to 0. Ideally, the joint is free running for simplicity, but analysis will reveal that the system is not controllable with only the motion of the cart. A passive torque is needed (spring and/or damper), and if not sufficient, an actuator could be required. Applying Eq. (4.12) the same way as it was done with the wheel, we have:

$$\sum \boldsymbol{M}_G = \dot{\boldsymbol{H}}_G$$
$$(M_{X_G} - M_X)\vec{I} + (M_{Y_G} - M_Y)\vec{J} + (M_{Z_G} - M_Z)\vec{K} = (\dot{\boldsymbol{H}}_G)_{XYZ} + \boldsymbol{\Omega}_{XYZ} \times \boldsymbol{H}_G$$

with:

$$\boldsymbol{H}_G = J_{gX}\dot{\phi}\vec{I} + J_{gY}\dot{\theta}\sin\phi\vec{J} + J_{gZ}\dot{\theta}\cos\phi\vec{K}$$

where: J_{gX}, J_{gY}, J_{gZ} are the moments of inertia of the gimbal along XYZ. Note that the gimbal rotates with the XYZ coordinate system and its principal axes are along XYZ.

We obtain a set of three equations:

$$M_{X_G} = M_X + J_{gX}\ddot{\phi} + (J_{gZ} - J_{gY})\dot{\theta}^2 \sin\phi\cos\phi \tag{4.20}$$

$$M_{Y_G} = M_Y + J_{gY}(\ddot{\theta}\sin\phi + \dot{\theta}\dot{\phi}\cos\phi) + (J_{gX} - J_{gZ})\dot{\theta}\dot{\phi}\cos\phi \tag{4.21}$$

$$M_{Z_G} = M_Z + J_{gZ}(\ddot{\theta}\cos\phi + \dot{\theta}\dot{\phi}\sin\phi) + (J_{gY} - J_{gX})\dot{\theta}\dot{\phi}\sin\phi \tag{4.22}$$

Rod

Figure 4.6 shows the main pendulum component, the rod. As explained previously, it is assumed that the rod holds the mass of the gyroscopic assembly for simplification purposes since the centre of gravity of the wheel and the gimbal are fixed relative to the rod, regardless of θ, ϕ, and ψ. Thus we define:

$$m = m_{rod} + m_{gimbal} + m_{wheel}$$

From this point, we will use the xyz coordinate system (fixed orientation and located on the pivot of the rod) instead of the rotating XYZ coordinate. xyz only moves relative to x in a linear motion. A fixed orientation reference is simpler to use for the following derivations. However, a conversion from XYZ to xyz has to be made for the moments calculated previously. By observing Fig. 4.6, it can be shown that torques originating from the gimbal (holding the gyroscopic wheel) may be converted to the xyz reference as:

$$\begin{bmatrix} M_{x_G} \\ M_{y_G} \\ M_{z_G} \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta\cos\phi & \sin\theta\cos\theta \\ \sin\theta & \cos\theta\cos\phi & -\cos\theta\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} M_{X_G} \\ M_{Y_G} \\ M_{Z_G} \end{bmatrix} \tag{4.23}$$

Applying Eq. (4.11) to the rod ($\sum \boldsymbol{F} = m\boldsymbol{a}$), we obtain the following equations:

$$-G_H = m\ddot{x}_{CG}$$
$$-G_V - mg = m\ddot{y}_{CG}$$

or,

$$G_H = -m\frac{d^2}{dt^2}(x - l\sin\theta)$$
$$G_V = m\frac{d^2}{dt^2}(l\cos\theta) - mg$$

which simplifies to,

$$G_H = ml\ddot{\theta}(\cos\theta - \sin\theta) - m\ddot{x} \tag{4.24}$$

$$G_V = ml\ddot{\theta}(\sin\theta + \cos\theta) - mg \tag{4.25}$$

Using Eq. (4.12) ($\sum \boldsymbol{M}_G = \dot{\boldsymbol{H}}_G$) with:

$$\boldsymbol{H}_G = J_{rz}\dot{\theta}\vec{k}$$

where: J_{rz} is the moment of inertia of the rod around its centre of gravity along z, $\boldsymbol{H}_G$ is the angular momentum of the rod, which can only rotate about the z axis,

As it was assumed, the mass of the gimbal and the disc are lumped in the rod. When calculating J_y, include m_g and m_w as point masses at the origin of XYZ. We obtain:

$$\begin{aligned} M_x - M_{x_G} &= 0 \\ M_y - M_{y_G} &= 0 \\ M_z - M_{z_G} - G_V l \sin\theta - G_H l \cos\theta &= J_{rz}\ddot{\theta} \end{aligned}$$

We note that the rod's joint is free along the z axis, and consequently $M_z = 0$. Eliminating M_{x_G}, M_{y_G}, and M_{z_G} using Eq. (4.23) we obtain:

$$\begin{aligned} M_x &= M_{X_G}\cos\theta - M_{Y_G}\sin\theta\cos\phi + M_{Z_G}\sin\theta\cos\theta \\ M_y &= M_{X_G}\sin\theta + M_{Y_G}\cos\theta\cos\phi - M_{Z_G}\cos\theta\sin\phi \\ -M_{Y_G}\sin\phi - M_{Z_G}\cos\phi - G_V l \sin\theta - G_H l\cos\theta &= J_{rz}\ddot{\theta} \end{aligned} \tag{4.26}$$

M_x and M_y are useful to determine the reaction torques at the pivot joint. However, for the stability analysis these quantities are not needed. Substituting Eqs. (4.24) and (4.25) into Eq. (4.26), we eliminate G_H and G_V to obtain:

$$-M_{Y_G}\sin\phi - M_{Z_G}\cos\phi + mlg\sin\theta + ml\ddot{x}\cos\theta = (J_{rz} + ml^2)\ddot{\theta} \tag{4.27}$$

Eq. (4.27) is a differential equation that relates the angle of tilt of the pendulum with the gimbal rate of rotation and the cart's acceleration.

Cart

The cart (Figure 4.7) is a mobile block (of position x) that is constrained to move only along axis x. It is fixed orientation and has a mass m_{cart}. An actuator applies a force F along its motion axis in order to control the balance and position of the inverted pendulum. Note that force vectors (reaction forces) in the figure are included for completeness but are not all necessary in this section. They are mainly useful to

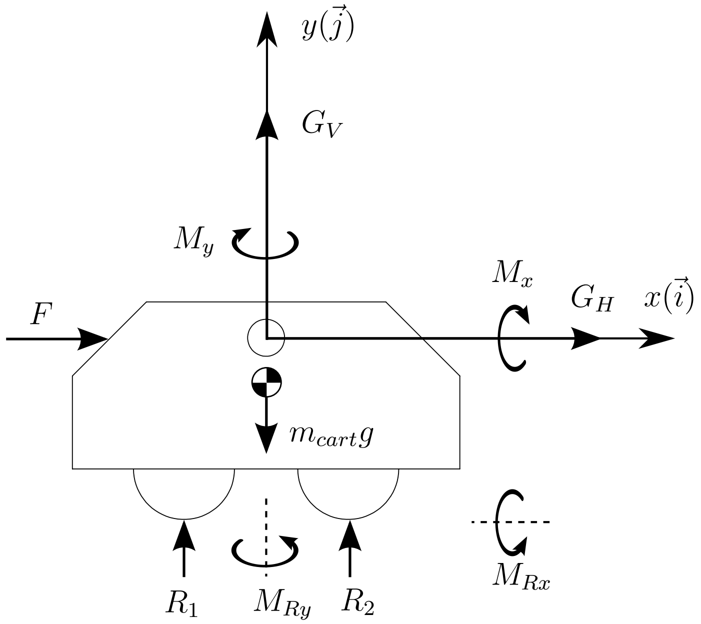


Figure 4.7: Forces acting on the cart.

calculate the reaction forces when designing the mechanical components of the system. Applying Newton's second law along x, we have:

$$\sum F_x = m_{cart}\ddot{x}$$

$$F + G_H = m_{cart}\ddot{x} \tag{4.28}$$

Equation (4.28) relates the force from the actuator to its effect on the motion of the pendulum.

4.1.2.1 Nonlinear equations of motion

Even if a linear state space controller is used, it is necessary to simulate the nonlinear system to ensure the control law will work with the physical apparatus. In this section the nonlinear equations will be derived. In order to simplify the expressions, the following constants were defined:

$$M = m + m_{cart}$$

$$J_X = J_{gX} + J'_w$$

$$J_Y = J_{gY} + J_w$$

$$J_Z = J_{gZ} + J'_w$$

For the gimbal, we use Eq. (4.20) with M_X and M_{X_G} substituted by Eqs. (4.16) and (4.19) respectively. We obtain:

$$J_X\ddot{\phi} + (J_Z - J_Y)\dot{\theta}^2 \cos\phi \sin\phi = \tau - k\phi - b\dot{\phi} + H\dot{\theta}\cos\phi$$

For the rod, we use Eq. (4.27) and substitute M_{Y_G} and M_{Z_G} using Eqs. (4.21) and (4.22). The resulting equation has variables M_Y and M_Z which we eliminate using Eqs. (4.17) and (4.18). We obtain:

$$\begin{aligned}&\ddot{\theta}(J_{rz} + J_Z\cos^2\phi + J_Y\sin^2\phi + ml^2)\\ &+\dot{\phi}\cos\phi(H + 2\dot{\theta}\sin\phi(J_Y - J'_w)) = m\ddot{x}l\cos\theta + mgl\sin\theta\end{aligned}$$

For the cart, we use Eq. (4.28) and substitute G_H from Eq. (4.24) which gives:

$$F + ml\ddot{\theta}(\cos\theta - \sin\theta) = M\ddot{x}$$

These three equations are sufficient to fully model the nonlinear system. To obtain the plant's transfer function in state space the last three equations are re-arranged to isolate $\ddot{\phi}$, $\ddot{\theta}$, and $\ddot{x}$. We obtain:

$$\ddot{\phi} = \frac{\tau - k\phi - b\dot{\phi} + H\dot{\theta}\cos\phi + (J_Y - J_Z)\dot{\theta}^2\cos\phi\sin\phi}{J_X} \tag{4.29}$$

$$\ddot{\theta} = \frac{Flm\cos\theta + MY}{Z} \tag{4.30}$$

$$\ddot{x} = \frac{FX + Yml(\cos\theta - \sin\theta)}{Z} \tag{4.31}$$

with: $Y = mgl\sin\theta - \dot{\phi}\cos\phi(H + 2\dot{\theta}\sin\phi(J_Y - J'_w))$

$X = J_{rz} + J_Z\cos^2\phi + J_Y\sin^2\phi + ml^2$

$Z = MX + m^2l^2(\sin\theta - \cos\theta)\cos\theta$

Equations (4.29) to (4.31) are the nonlinear equations of motion of the inverted pendulum with a gyroscope. These three equations are the basis of the nonlinear simulation shown in Section 4.1.4.1.

4.1.2.2 Linear equations of motion

Since the goal is to stabilize the system, we assume the angles θ and ϕ will be kept small. Thus we can linearize the equations of motion around the system's equilibrium point. At the equilibrium point, the control efforts F and τ are zero. Therefore to linearize the equations, we can use the first order multi-variable MacLaurin series expansion for θ, $\dot{\theta}$, ϕ, $\dot{\phi}$, F, and τ. For a given expression $f(\{\theta, \dot{\theta}, \phi, \dot{\phi}, F, \tau\})$, we calculate its linearization as:

$$f \approx f(\vec{0}) + \theta \left.\frac{\partial f}{\partial \theta}\right|_{\vec{0}} + \dot{\theta} \left.\frac{\partial f}{\partial \dot{\theta}}\right|_{\vec{0}} + \phi \left.\frac{\partial f}{\partial \phi}\right|_{\vec{0}} + \dot{\phi} \left.\frac{\partial f}{\partial \dot{\phi}}\right|_{\vec{0}} + F \left.\frac{\partial f}{\partial F}\right|_{\vec{0}} + \tau \left.\frac{\partial f}{\partial \tau}\right|_{\vec{0}} \tag{4.32}$$

Applying the above formula on Eqs. (4.29), (4.30), and (4.31) we obtained the linearized system as:

$$\ddot{\phi} = \frac{H\dot{\theta} - k\phi - b\dot{\phi} + \tau}{J_X} \tag{4.33}$$

$$\ddot{\theta} = \frac{Mmgl\theta - MH\dot{\phi} + mlF}{T} \tag{4.34}$$

$$\ddot{x} = \frac{m^2gl^2\theta - Hml\dot{\phi} + IF}{T} \tag{4.35}$$

with $I = J_Z + J_{rz} + ml^2$, the equivalent moment of inertia of the pendulum assembly around its pivot point, and $T = MI - m^2l^2$

4.1.3 Control System

4.1.3.1 State-space representation

The results from the previous analysis can be combined to obtain a model of the inverted pendulum with a gyroscope, that we call a plant. The plant's transfer function is to be expressed in the form of state-space.

We begin by selecting state variables that suitably represent the actual state of the system:

$$x_1 = \theta \qquad x_3 = x \qquad x_5 = \phi$$

$$x_2 = \dot{\theta} \qquad x_4 = \dot{x} \qquad x_6 = \dot{\phi}$$

These six states are sufficient to represent the complete model's state. The knowledge of these six variables and the knowledge of the control forces (τ and F) completely describe the balancing behaviour. Using Eqs. (4.33) to (4.35) we formulate the state space model:

$$\begin{aligned}
\dot{x_1} &= \dot{\theta} = x_2 \\
\dot{x_2} &= \ddot{\theta} = \frac{Mmgl}{T}x_1 - \frac{MH}{T}x_6 + \frac{ml}{T}F \\
\dot{x_3} &= \dot{x} = x_4 \\
\dot{x_4} &= \ddot{x} = \frac{gl^2m^2}{T}x_1 - \frac{Hml}{T}x_6 + \frac{I}{T}F \\
\dot{x_5} &= \dot{\phi} = x_6 \\
\dot{x_6} &= \ddot{\phi} = \frac{H}{J_X}x_2 - \frac{k}{J_X}x_5 - \frac{b}{J_X}x_6 + \frac{1}{J_X}\tau
\end{aligned}$$

Assuming the system has sensors for θ, x, and ϕ, we obtain the state-space representation of the system from the linear model as:

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} \tag{4.36}$$

$$\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{D}\boldsymbol{u} \tag{4.37}$$

with:

$$\boldsymbol{x} = \begin{bmatrix} \theta \\ \dot{\theta} \\ x \\ \dot{x} \\ \phi \\ \dot{\phi} \end{bmatrix}, \boldsymbol{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ \frac{Mmgl}{T} & 0 & 0 & 0 & 0 & \frac{-MH}{T} \\ 0 & 0 & 0 & 1 & 0 & 0 \\ \frac{gl^2m^2}{T} & 0 & 0 & 0 & 0 & \frac{-Hml}{T} \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & \frac{H}{J_X} & 0 & 0 & \frac{-k}{J_X} & \frac{-b}{J_X} \end{bmatrix},$$

$$\boldsymbol{B} = \begin{bmatrix} 0 & \frac{ml}{T} & 0 & \frac{I}{T} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{J_X} \end{bmatrix}^T,$$

$$\boldsymbol{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix},$$

$$\boldsymbol{D} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \text{ and } \boldsymbol{u} = \begin{bmatrix} F \\ \tau \end{bmatrix}$$

This state-space representation will be the basis of our control and stability analysis.

The mathematical model was obtained from a number of approximations in order to obtain linear equations. Since the equilibrium point is about the origin for θ and ϕ, we can assume that the system will be close to the real one as long as we remain in this region.

4.1.3.2 LQR controller design

As will be seen in the results of Section 4.1.4.2 the open loop system is unstable. To control the plant, a Linear Quadratic Regulator (LQR) was used to design an optimal controller [31]. It was chosen to use an optimal controller such as LQR because of the resulting smoothness of motion. There are a number of other possibilities that would all give suitable results (Fuzzy logic, neural networks, sliding mode controllers). The study of all possible controller types suitable for this system is outside the scope of this research. The system is controlled using

$$\boldsymbol{u} = -\boldsymbol{K}\boldsymbol{x} \tag{4.38}$$

where $\boldsymbol{K}$ needs to be determined using the LQR method. To do so, we define the performance index to be minimized as

$$J = \int_0^\infty (\boldsymbol{x}\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{u}\boldsymbol{R}\boldsymbol{u})\,\mathrm{d}t \tag{4.39}$$

with $\boldsymbol{Q}$ and $\boldsymbol{R}$ being symmetric and $\boldsymbol{Q}$ positive-semidefinite and $\boldsymbol{R}$ positive-definite. The initial values for $\boldsymbol{Q}$ and $\boldsymbol{R}$ were chosen according to Bryson's Rule where:

$$\boldsymbol{Q}_{ii} = \frac{1}{\text{maximum acceptable value of } [x_i]^2} \tag{4.40}$$

and

$$\boldsymbol{R}_{ii} = \frac{1}{\text{maximum acceptable value of } [u_i]^2} \tag{4.41}$$

and all other terms being 0. It is difficult to know all the system limits without an actual prototype, but some of them were programmed in the nonlinear model simulation. For example, the motor limits are according to Table 4.1: $F_{\text{lim}} = 1$ and $\tau_{\text{lim}} = 2$. From where we find $\boldsymbol{R}_{11} = 1$ and $\boldsymbol{R}_{22} = 0.25$. Also, assuming the cart has a maximum displacement of 0.2m and we limit its speed to 0.3m/s, we have $\boldsymbol{Q}_{33} = 25$ and $\boldsymbol{Q}_{44} = 11$. These values were chosen as a starting point, and were tuned by trial and error. The goal was to obtain a satisfying response for the position x with less importance being given to the other states. In the end, the best results were obtained with:

$$\boldsymbol{Q} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 25 & 0 & 0 & 0 \\ 0 & 0 & 0 & 9 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.42}$$

$$\boldsymbol{R} = \begin{bmatrix} 1 & 0 \\ 0 & 0.5 \end{bmatrix} \tag{4.43}$$

The $\boldsymbol{K}$ coefficients are obtained using the MATLAB *lqr* command (after initializing $\boldsymbol{Q}$ and $\boldsymbol{R}$):

```
%Obtain the K values
[K,P,E]=lqr(A,B,Q,R);
```

where we obtained

$$\boldsymbol{K} = \begin{bmatrix} -2.8 & -1.65 & 4.68 & 3.78 & -1.23 & -0.04 \\ -8.93 & -0.73 & -2.50 & -0.68 & -9.68 & 0.30 \end{bmatrix} \tag{4.44}$$

The pole placement method to obtain $\boldsymbol{K}$ was also tried but didn't provide good results compared with the LQR method, unless the poles from the LQR method were used as a starting point. But it was difficult to further increase the performance. In many trials, the linear system was stable but the nonlinear system could not be controlled.

4.1.3.3 Observer design

Because we have 6 states in $\boldsymbol{x}$ but only 3 outputs $\boldsymbol{y}$ (3 sensor are available) as shown in Eqs. (4.36) and (4.37), we need to design an observer if we want to control the system using the full state feedback control law (4.38). The observer is defined as follow [33]:

$$\dot{\hat{\boldsymbol{x}}} = \boldsymbol{A}\hat{\boldsymbol{x}} + \boldsymbol{B}\boldsymbol{u} + \boldsymbol{L}(\boldsymbol{y} - \boldsymbol{C}\hat{\boldsymbol{x}}) \tag{4.45}$$

with $\hat{\boldsymbol{x}}$ being the observed/estimated state, and $\boldsymbol{L}$ a constant matrix that needs to be determined. Having this observer, we rewrite the control law to use the observed states rather than the actual states:

$$\boldsymbol{u} = -\boldsymbol{K}\hat{\boldsymbol{x}} \tag{4.46}$$

To obtain $\boldsymbol{L}$, we simply use the pole placement method to have the poles of the observer 5 times the poles of the closed loop system. Using MATLAB we find:

```
EstPole = 5*LQREig;
L=place(A′,C′,EstPole)′
```

$$\boldsymbol{L} = \begin{bmatrix} 216.27 & 128.95 & 545.59 \\ 19386 & 15511 & -18337 \\ 182.58 & 137.41 & 417.53 \\ 10188 & 8150.9 & -7739.1 \\ -299.34 & -249.16 & 477.66 \\ 108990 & 75516 & 129500 \end{bmatrix} \tag{4.47}$$

4.1.3.4 Reference Input

It is desirable to control the cart's position from an external reference value. This reference is integrated into the system control loop by rewriting the control law as:

$$\boldsymbol{u} = -\boldsymbol{K}\hat{\boldsymbol{x}} + \bar{\boldsymbol{N}}\boldsymbol{r} \tag{4.48}$$

where $\bar{\boldsymbol{N}}$ is a constant matrix that needs to be determined and $\boldsymbol{r}$ is the reference input vector. To obtain $\bar{\boldsymbol{N}}$, the formal method seen in [33] is applied. First, we construct a temporary matrix as:

$$\boldsymbol{N}_t = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{D} \end{bmatrix}^{-1} \tag{4.49}$$

But to be invertible a matrix needs to be square. Since $\boldsymbol{A}$ is 6 × 6, $\boldsymbol{B}$ is 6 × 2, and $\boldsymbol{C}$ is 3 × 6, we remove one row in $\boldsymbol{C}$ for $\boldsymbol{N}_t$ to become square. The most logical row to remove is the second row corresponding to θ because this angle is not controllable in steady state (it is physically impossible to keep the rod at an angle other than 0 in steady state). Once the above matrix is calculated, we write:

$$\begin{bmatrix} \boldsymbol{x}_{ss} \\ \boldsymbol{u}_{ss} \end{bmatrix} = \boldsymbol{N}_t \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{r}_{ss} \end{bmatrix} \tag{4.50}$$

from where we define:

$$\boldsymbol{x}_{ss} \equiv \boldsymbol{N}_{\boldsymbol{x}}\boldsymbol{r}_{ss} \tag{4.51}$$

$$\boldsymbol{u}_{ss} \equiv \boldsymbol{N}_{\boldsymbol{u}}\boldsymbol{r}_{ss} \tag{4.52}$$

from which matrices $\boldsymbol{N}_{\boldsymbol{x}}$ and $\boldsymbol{N}_{\boldsymbol{u}}$ can be extracted. Then we can calculate:

$$\bar{\boldsymbol{N}} = \boldsymbol{N}_{\boldsymbol{u}} + \boldsymbol{K}\boldsymbol{N}_{\boldsymbol{x}} \tag{4.53}$$

and substitute into (4.48) with:

$$\boldsymbol{r} = \begin{bmatrix} r_x \\ r_\phi \end{bmatrix} \tag{4.54}$$

Using the simulation parameters we obtain:

$$\bar{\boldsymbol{N}} = \begin{bmatrix} 4.6782 & -1.2478 \\ -2.4956 & -4.6782 \end{bmatrix} \tag{4.55}$$

4.1.4 Simulation and results

4.1.4.1 Simulation setup

The nonlinear and linear equations of motion for the system were implemented in Simulink. For the linear model, a combination of continuous objects were used, while for the nonlinear model, both continuous blocks and a MATLAB embedded function block were used. The embedded function block allows to directly write the nonlinear equations as MATLAB code which simplifies the simulation design process.

For the nonlinear model, the motors were also implemented with limits of how much force they could produce. These limits are denoted by the actuator limit values F_{lim} and τ_{lim} given in Table 4.1. The details of the implementation of the Simulink model are shown in Appendix H.

4.1.4.2 Open loop system

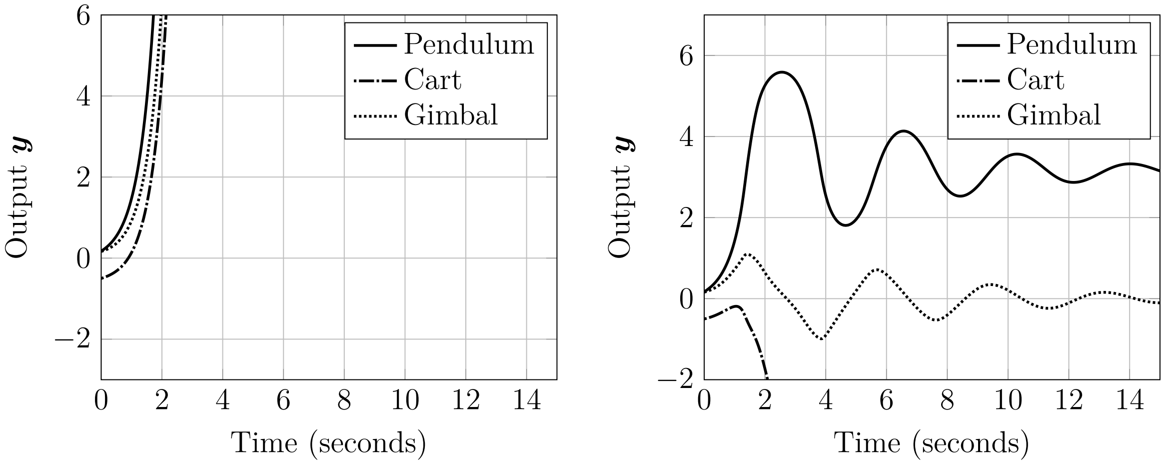
To test the simulation model, and to obtain a better understanding of the behaviour of the open loop system, the free response of both the nonlinear and linear models was simulated. The response to an initial condition of 10° for θ and ϕ and of 0.5m for x:

$$[\theta,\ x,\ \phi] = \left[\frac{10\pi}{180},\ -0.5,\ \frac{10\pi}{180}\right]$$

was simulated. Those initial conditions will be the same for every result in this report except in the last section about the reference input, where they will be null.

Figure 4.8(a) shows the response of the linear system while Fig. 4.8(b) is for the nonlinear system.

Both systems are unstable but differ by their response. Because of the linearization $\sin\theta \approx \theta$ the linear system exponentially rises while the nonlinear system stabilizes



(a) Response of the linear system.

(b) Response of the nonlinear system.

Figure 4.8: Open loop response of the linear system subjected to initial conditions.

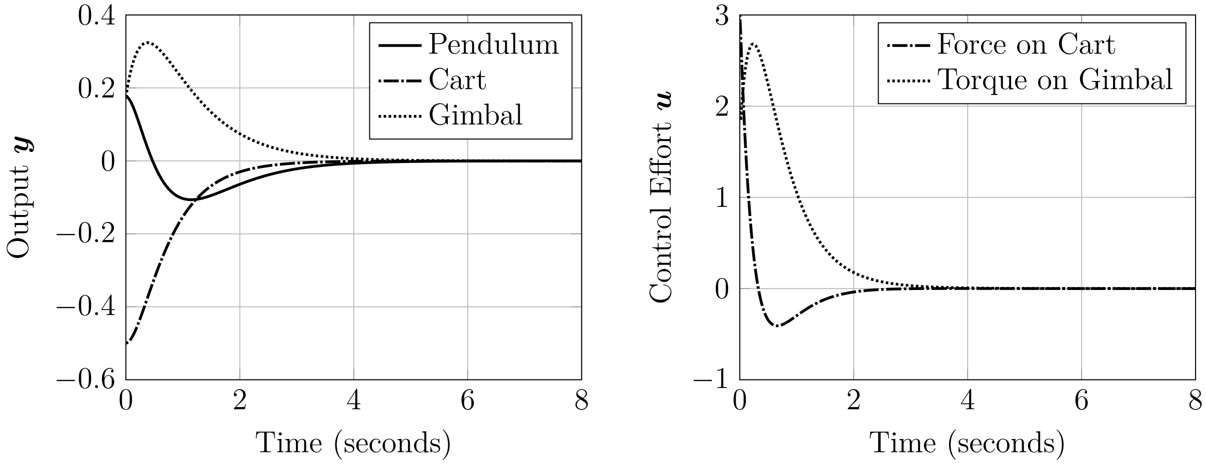
to equilibrium point at $\theta = \pi$. Although the linear and nonlinear open loop response differ, they still have the same tendency when the nonlinear system is close to its linear region.

4.1.4.3 LQR controller results

The simulation was run for the system response to initial conditions (4.1.4.2) using the feedback control law (4.38). The results are shown in Fig. 4.9(a) and Fig. 4.9(b) for the linear system, and Fig. 4.9(c) and Fig. 4.9(d) for the nonlinear system.

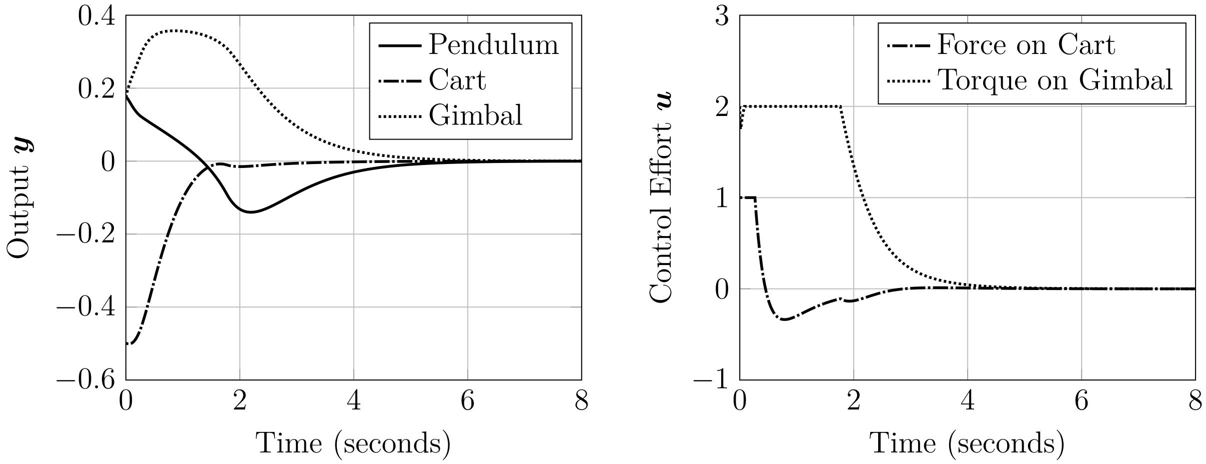
By observing the response of the linear system (Fig. 4.9(a)), we see that it is now stable with a settling time of about 2 seconds on the cart position. By comparison, the cart settling time for the nonlinear system (Fig. 4.9(c)) is about 1.5 seconds, however, the response for the rod's angle and the gimbal angle are more sluggish and seem to be in an uncorrected state for longer.

The main difference in response is due to the nonlinearity of the actuators. By comparing the control effort of the linear (Fig. 4.9(b)) and nonlinear (Fig. 4.9(d)) systems, we notice that the actuator's limits are reached in the nonlinear system.



(a) Response of the linear system.

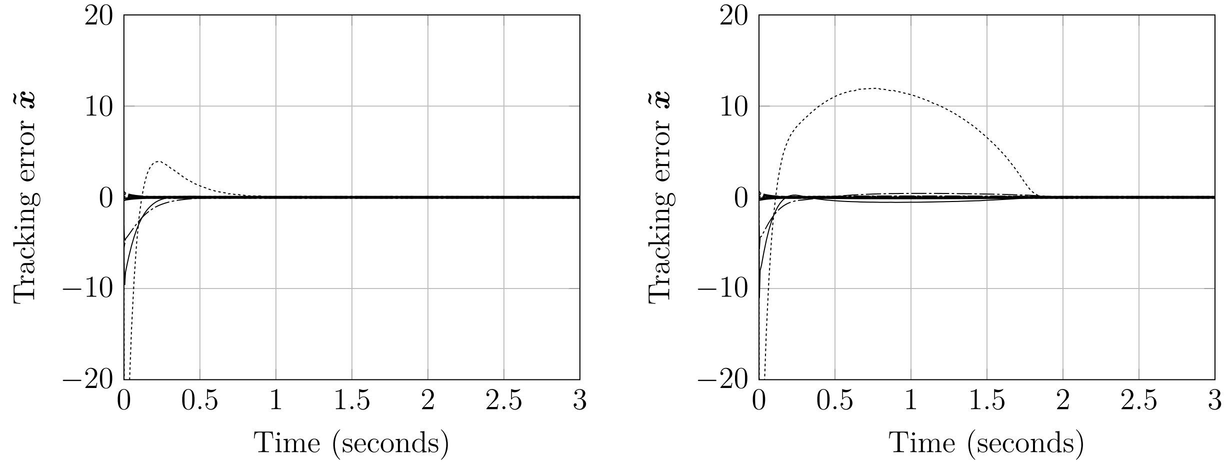
(b) Control effort of the linear system.



(c) Response of the nonlinear system.

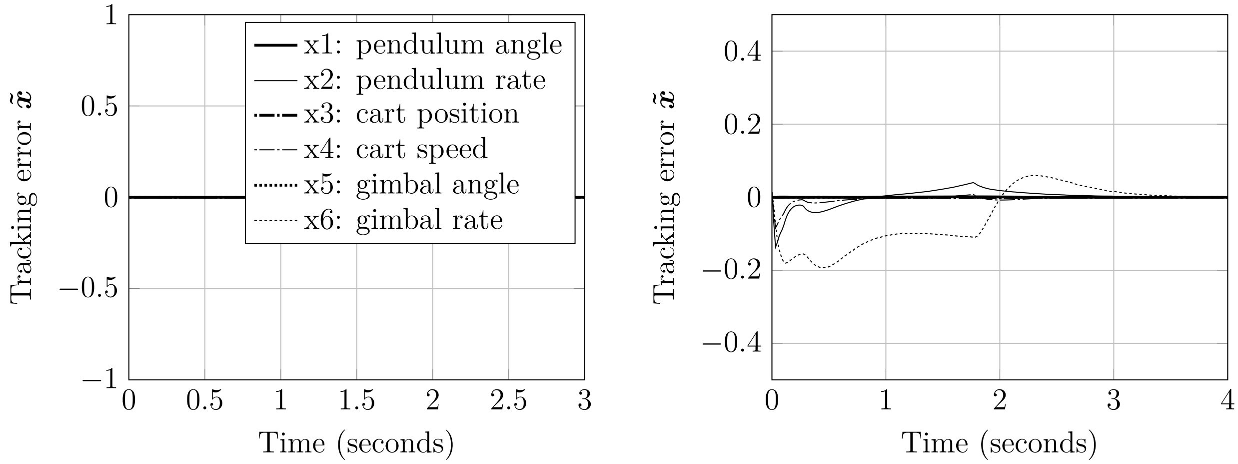
(d) Control effort of the nonlinear system.

Figure 4.9: Simulation results of the closed loop LQR controller.



(a) Initial linear system.

(b) Initial nonlinear system.



(c) Improved (with initial conditions) linear system. Error is exactly zero for all states.

(d) Improved (with initial conditions and actual $\boldsymbol{u}$) nonlinear system.

Figure 4.10: Observer tracking error results for both the initial and improved observer designs.

4.1.4.4 Observer results

The simulation is run again with the observer for both the linear and nonlinear models. To analyze the observer's performance we define the tracking error as the difference between the estimated states and the actual states:

$$\tilde{\boldsymbol{x}} = \hat{\boldsymbol{x}} - \boldsymbol{x} \tag{4.56}$$

Fig. 4.10(a) and Fig. 4.10(b) show the tracking error for the linear and nonlinear systems. The tracking error is fairly high at the first half second on the linear system and during the first two seconds on the nonlinear system. The largest error is found to be -46.1rad/s for the linear system and -38.3rad/s for the nonlinear system (both at time 0). The error becomes null very fast for most states and shows that the observer correctly estimates the actual states.

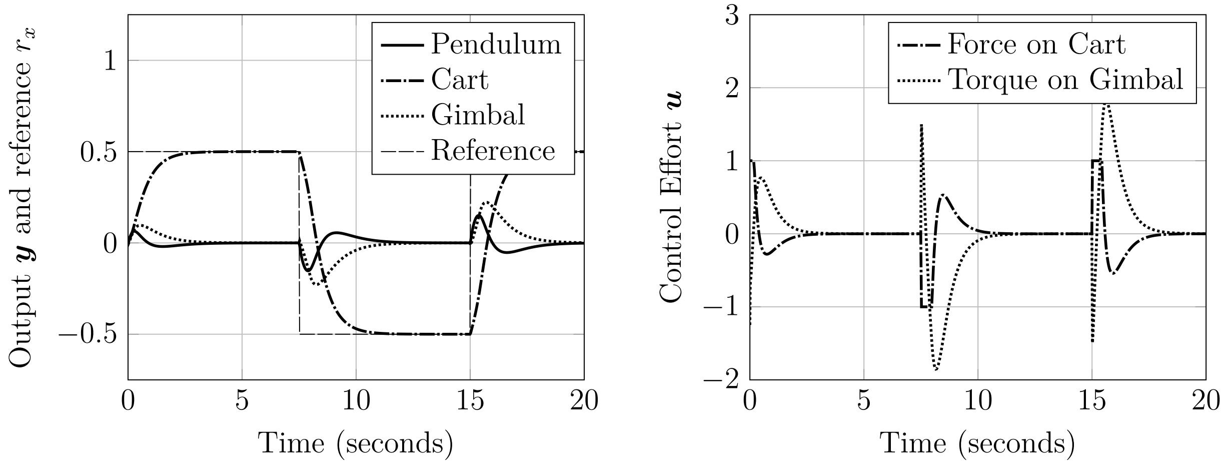
The high initial error can be a problem in some systems as there will be a jitter from the motors when the system starts. Small improvements were brought to the observer's simulation in an attempt to reduce this error. First, the observer's initial condition in $\boldsymbol{y}$ were set to match the initial sensor readings at time 0.

For the nonlinear system, another improvement was to make the observer aware of the motor limits. This was achieved by cropping the value of $\boldsymbol{u}$ whenever it exceeded the saturation limits F_{lim} and τ_{lim} when evaluating (4.45).

The tracking error of the improved observer is shown in Fig. 4.10(c) for the linear model and in Fig. 4.10(d) for the nonlinear model. For the linear system, the error has decreased to zero and to 0.2rad/s for the nonlinear system.

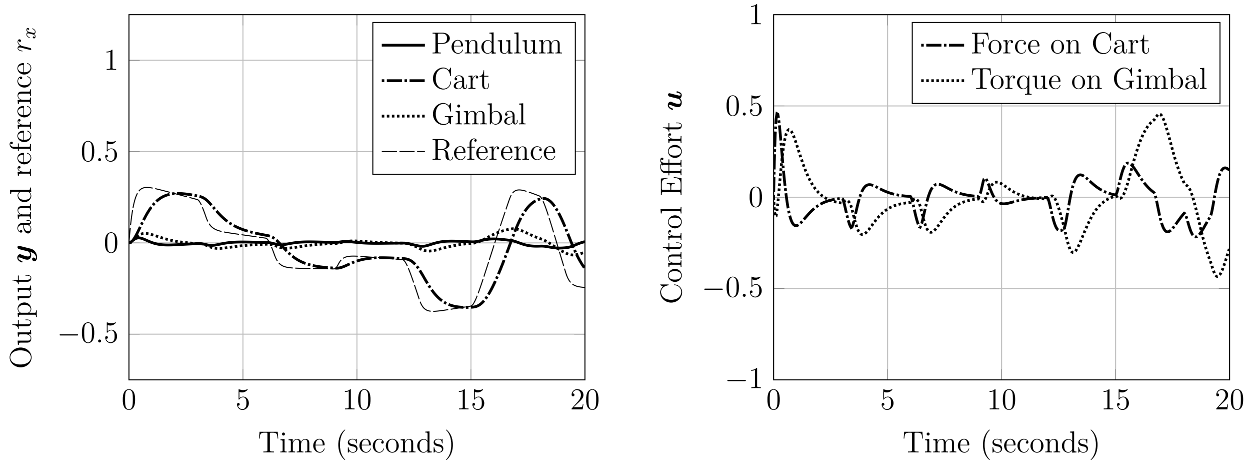
4.1.4.5 Path following

To test the reference input signal for the position, two scenarios were examined. The first scenario sent a square wave position signal of amplitude 0.5m with a frequency of 15 seconds. The results for the nonlinear system are shown in Fig. 4.11(a) and Fig. 4.11(b). The second was to send a random path signal. It was created using a combination of a random number generator with a low pass filter. The results for the nonlinear system are shown in Fig. 4.11(c) and Fig. 4.11(d). In both cases, the command was followed closely by the cart while the control system kept the rod and the gimbal within small values. There is a settling time of about 2.5 seconds for the square wave signal and there is a constant lag of less than a second for the reference path. If we observe the control effort for the square wave signal, the force on the cart saturates momentarily when there is a step on the reference, and the torque on the gimbal stays within the motor limits. For the reference path, it seems the control effort is kept relatively small throughout. This is where we see that it is better to have slowly varying reference inputs rather than step inputs. To increase



(a) Response to a square wave reference position.

(b) Control effort with a square wave reference position.



(c) Response to a random path reference position.

(d) Control effort with a random path signal reference position.

Figure 4.11: Nonlinear simulation following a square wave and random reference input.

the performance of this system to a reference input, an integrator would need to be added in the controller. The integrator would help reduce the lag, and would also reduce the steady state error should there be friction or external forces on the cart (pushing an object or climbing a hill).

4.1.4.6 Effect of gyroscope speed

Modifying the gyroscope speed changes the momentum H in Eq. 4.36. To show the effect of varying H, a number of nonlinear simulations for a step input and for a random path reference signal were performed. Figure 4.12(a) shows the step response of the system with gyroscope rotation speeds from 0 to 15000rpm. For every simulation, the LQR coefficients were kept constant, but the feedback gain $\boldsymbol{K}$ was recalculated due to state space model changing.

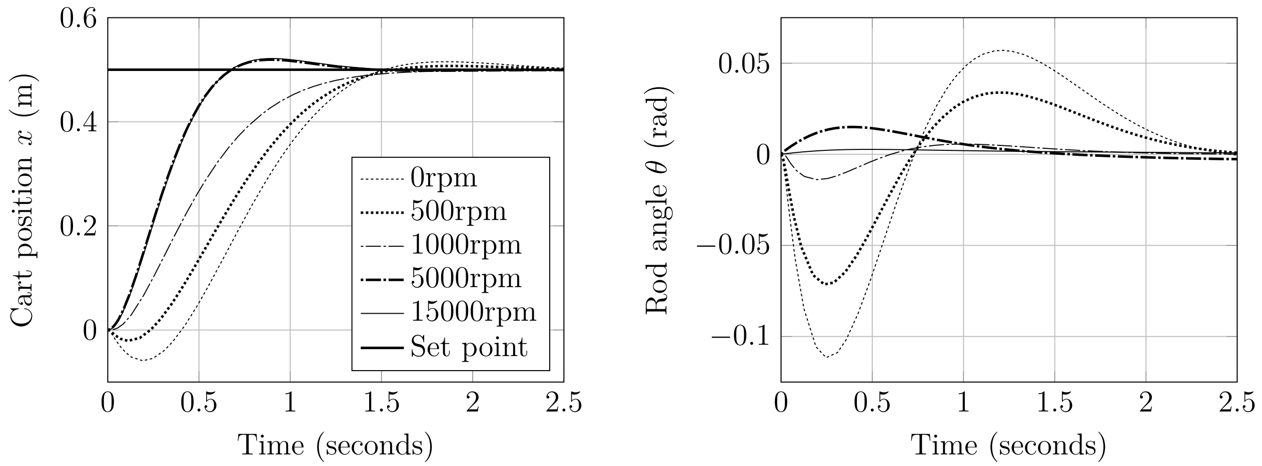
At 0rpm the system is equivalent to a classic inverted pendulum. In order to change position, the cart must step back by 0.06m to create a negative tilt angle seen on Fig. 4.12(b). This negative tilt angle is required to carry the rod centre of gravity forward. To settle at the set point, the cart must overshoot to create a positive rod angle that will allow it to stop. By adjusting the LQR coefficient it would be possible to obtain no overshoot, at the expense of settling time, but a step back motion would still be required. The 1% settling time for 0rpm is 2.4s. At 500rpm, the step back motion is reduced from 0.06m to 0.02m. The negative rod deflection angle is also reduced. At 1000rpm, the rod still shows a negative deflection angle but the step back motion is eliminated. The 1% settling time is reduced to 1.6s. At 5000rpm, the cart settling time is 1.3s but with a slight position overshoot. Observing Fig. 4.12(b) the rod deflection angle becomes positive. This means that between 1000 and 5000rpm, the rod stays completely vertical regardless of the motion of the cart. At this point, the cart motion would be critically damped with no overshoot. At 15000rpm there is no benefit for the cart position and settling time. This is because the cart's actuator saturates and cannot move the cart any faster. Increasing the gyroscope speed to 15000rpm increases momentum resulting in a lower rod and gimbal deflection compared to 5000rpm. This is clearly seen in

Fig. 4.12(c). The disturbance rejection of the system is improved before the gimbal reaches its saturation limit.

Figure 4.12(d) shows the system following a random reference path. As the gyroscope speed increases, the cart is able to follow the set point with less error. There is a constant error when the set point varies in a slope since the controller has no integrator. Also, no difference is seen between 5000 and 15000rpm because of the cart actuator saturation limits.

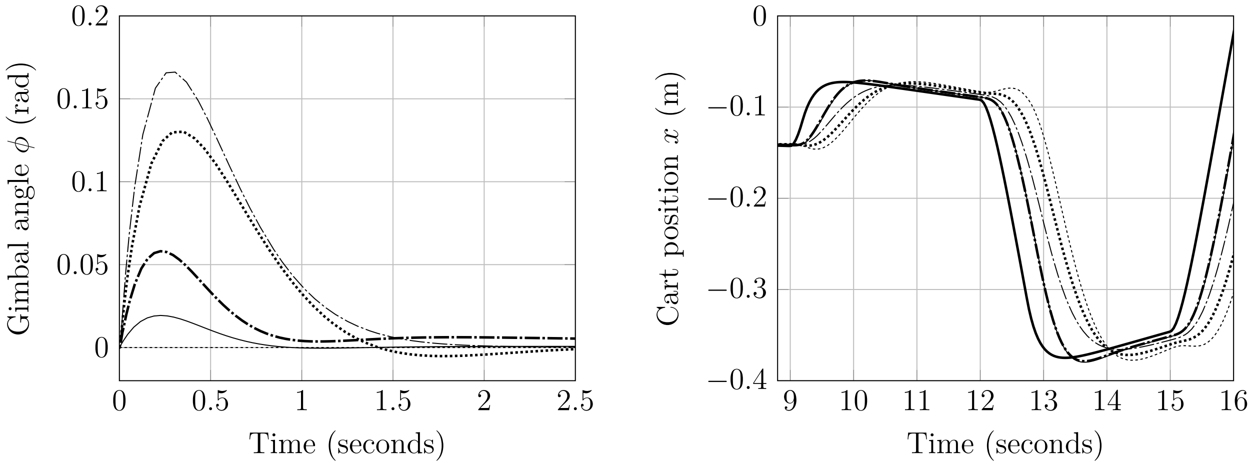
We mentioned there is a rotation speed at which the rod angle deflection is minimal (nonzero because of nonlinear effects). To find this speed, a large number of simulations were performed by varying the gyroscope speed from 0 to 10000rpm. Figure 4.13 shows a plot of the maximum rod deflection angle for a step input against the rotation speed. We see that as the speed increases, the rod deflection decreases, until a minimum at 1125rpm. From this point, the deflection has a maximum at 2325rpm. Further increasing the speed decreases the deflection to 0.

Adding the gyroscopic gimbal setup eliminates the step back effect and reduces the rod deflection angle. It also increases the disturbance rejection of the system. There are two speed at which the rod deflection is minimized: 1125rpm and infinity. The best speed for a given system depends on whether the goal is smooth motion or high disturbance rejection, and depending on the system limits.



(a) Cart position step response.

(b) Rod angle step response.



(c) Gimbal angle step response.

(d) Cart position random path reference.

Figure 4.12: Nonlinear simulation at various gyroscope speeds following a square wave and random reference input.

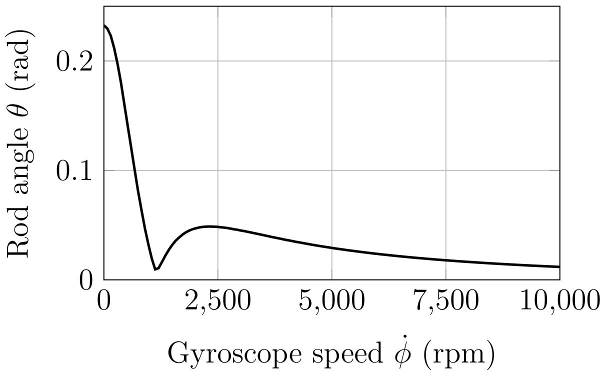


Figure 4.13: Effect on gyroscope speed on the rod deflection angle.

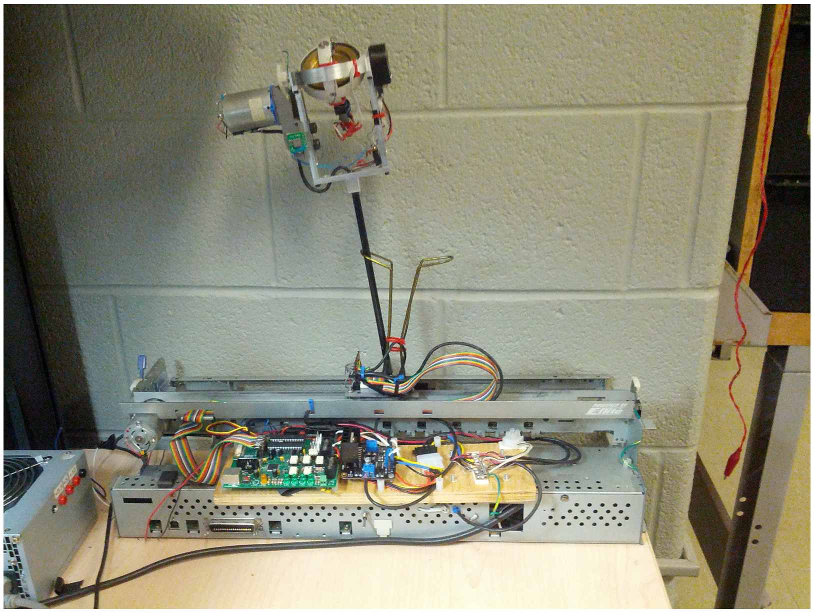


Figure 4.14: Complete hybrid balancing system apparatus.

4.2 Practical Implementation of the hybrid stabilization system

This section documents the development of a mechanism used to provide a proof-of-concept for the hybrid inverted pendulum stabilization system.

4.2.1 Development of the test platform

In order to validate the theoretical results of the hybrid stability system, a custom apparatus was developed. This section is divided in three parts. First we will look at the design of the cart and pendulum, then the gimbal assembly, and finally the control system and associated electronics.

4.2.1.1 Cart and pendulum design

To effectively replicate the conditions imposed in the simulation, the cart has to move with the following characteristics:

- High positional accuracy
- Low backlash
- Movement constrained along one axis
- Fast response to control commands (high torque and high speed motor)

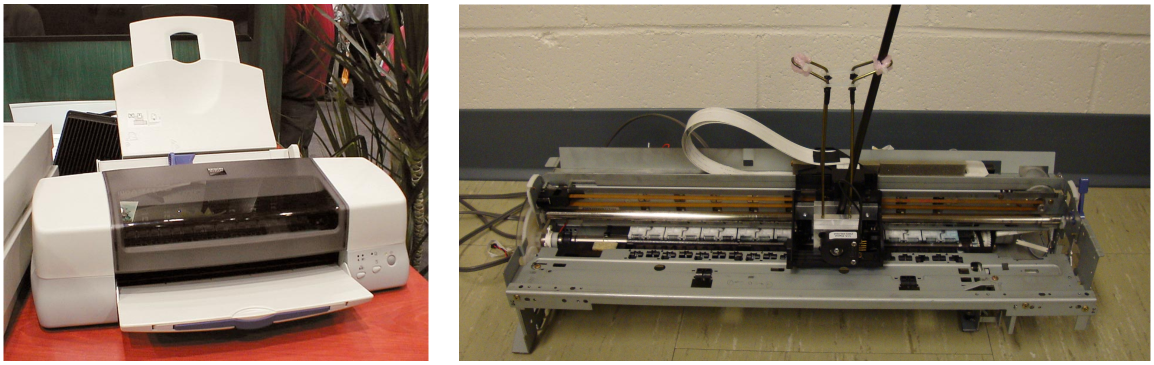

(a) (b)

Figure 4.15: (a): The Epson PM-3300C ink-jet printer. (b): Printer converted into an inverted pendulum apparatus.

- Low friction, low dampening
- Rugged and Durable

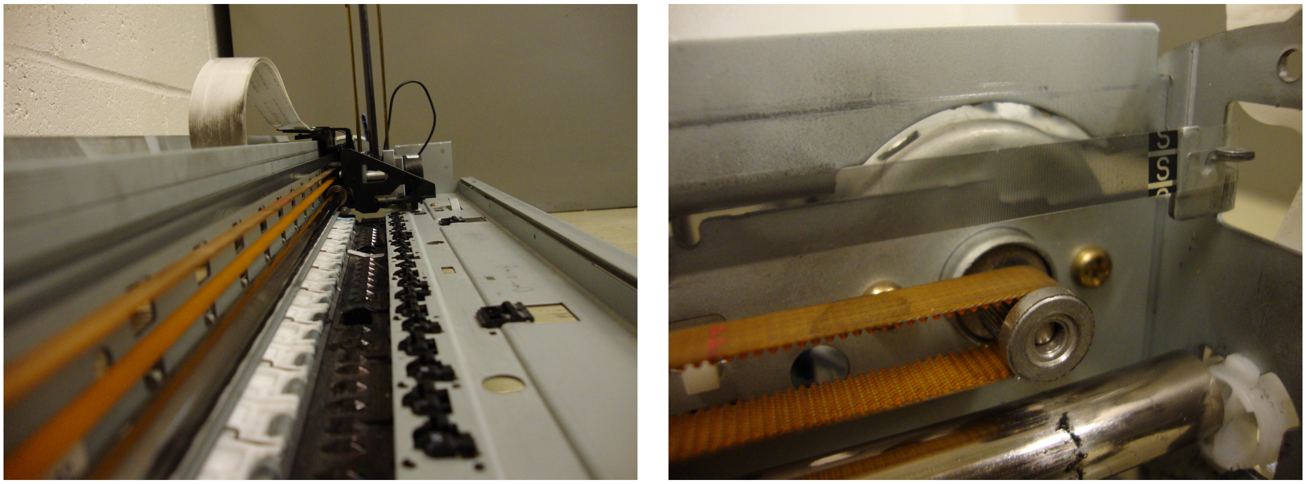
An ink-jet printer head control mechanism was chosen for the implementation of the cart since its motion is both fast and precise. The cartridge position can be controlled with a high positional accuracy and with negligible backlash or play.

An old ink-jet printer was used for this purpose. The Epson PM-3300C shown in Fig. 4.15(a) features a wide linear guide (13") that was used for printing large photos, a fast printing performance, and a 1440dpi resolution.

Figure 4.15(b) shows the printer dismantled and converted into an inverted pendulum. The ink cartridge holder was used as the cart and the cartridges were replaced by a mechanical joint coupled to a carbon fibre rod.

The cartridge holder follows a rail that consists of a metal rod that spans the whole width of motion, and a slider near its top that rubs against a sheet metal flange. This slider prevents the cartridge from spinning around the rod (Fig. 4.16(a)). To move the cart is attached to a belt which is driven by a DC motor (Fig. 4.16(b)). The cart is also equipped with a linear encoder.

Figure 4.17(a) shows a detail view of the cart-rod joint assembly. The carbon fibre rod is fixed to a bearing mounted joint equipped with an optical encoder. The encoder serves as a angular feedback signal for the control loop. Mounted on the joint assembly are also mechanical limits preventing the rod from falling too far away from



(a)

(b)

Figure 4.16: (a): Side view of the cart and its linear guide rail. (b): View of the drive pinion attached to a belt. The cart is driven by the belt. The transparent strip above the belt is a position encoder.

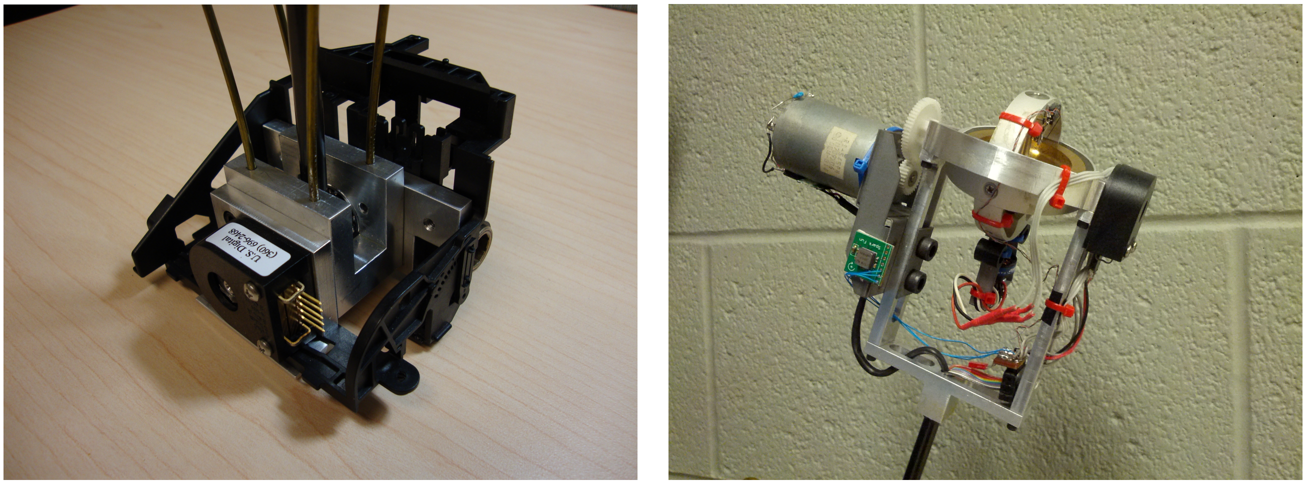
its equilibrium position.

Please refer to Appendix I for assembly and part drawings for the mechanical design. The digital files are also available for download here [26].

4.2.1.2 Gyroscopic assembly design

The top of the pendulum is equipped with a gyroscope/gimbal assembly shown in Fig. 4.17(b). Its layout and orientation matches the theoretical model shown in Fig. 4.3. This assembly consists of a gyroscopic disc driven by a brushless motor, which are mounted onto a gimbal. This gimbal is able to rotate relative to the bracket. A motor applies rotation torques to the gimbal, while on the opposite side an angular encoder reads the gimbal angle for use by the control loop.

The gimbal motor shown in Fig. 4.18(a) is a 42V motor with mild magnetic poles. Many motors with lower operating voltages have magnets too strong that result in cogging as the rotor coils pass to the next magnet. This low friction motor allows the precession torque of the gyroscope to drive the motor in reverse and the high motor torque can generate a precession force. An optical encoder shown in Fig. 4.18(b) is used to measure the gimbal rotation.



(a) (b)

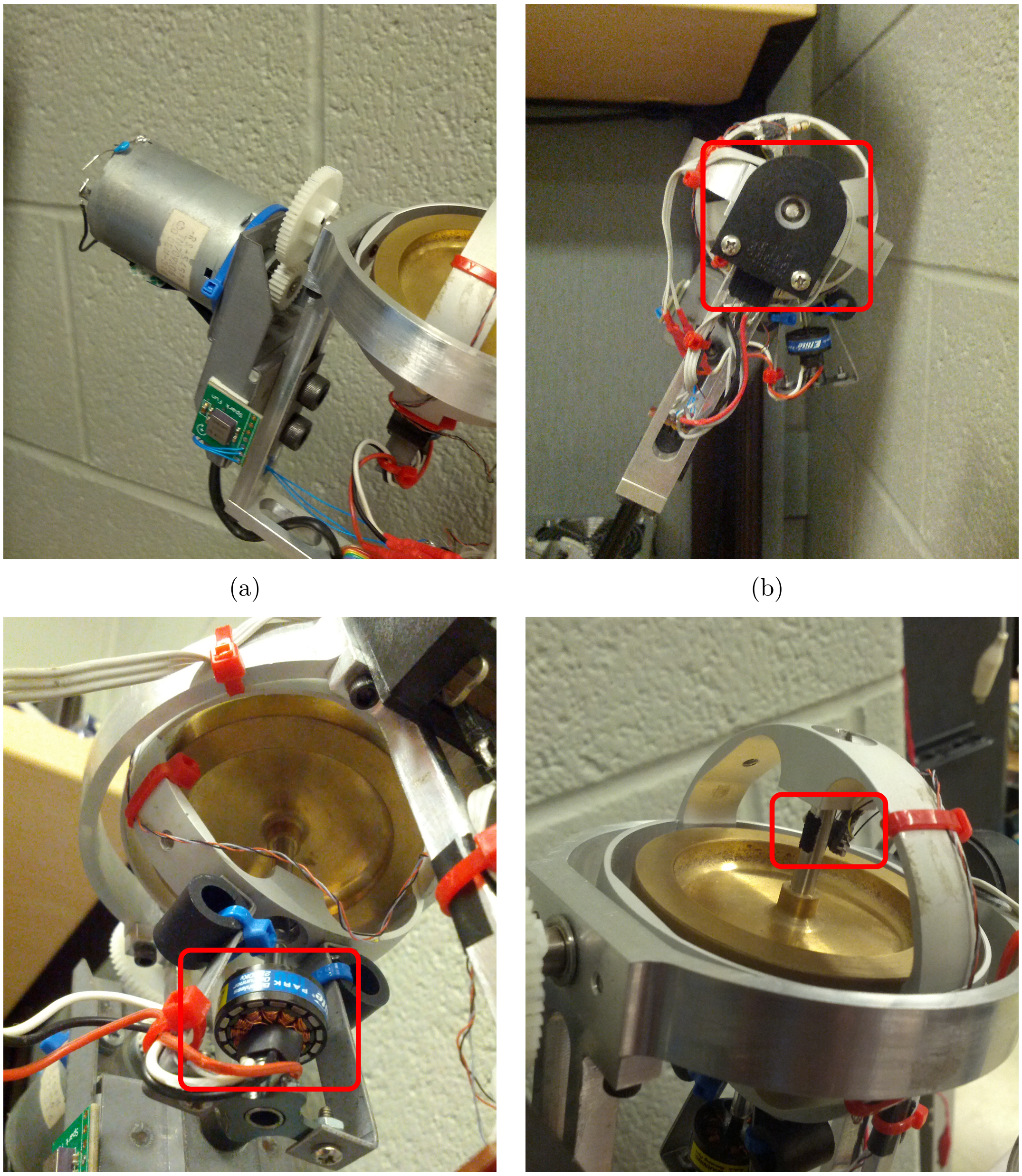
Figure 4.17: (a): View of the cart-rod joint assembly. (b): View of the complete gimbal/gyroscope assembly.

Attached to the gyroscopic disk is a brushless motor (Fig. 4.18(c)). On the other side of the disk is an optical sensor shown in Fig. 4.18(d). The optical sensor is custom made to measure the disc shaft rotation speed. Half of the shaft is painted black and an infrared reflective sensor detects what half of the shaft it is pointing to. This motor, coupled with the rotation sensor allows the control electronics to regulate the speed of the gyroscope to a constant speed. Please refer to Appendix I for assembly and part drawings for the mechanical design.

4.2.1.3 Control system

The sensors and motors are interfaced with the computer using custom circuitry specifically developed to work with the prototype. The main board is on the left of Fig. 4.19 and is connected to the computer using a USB connection. The board has a USB connector on the lower left, a ribbon connector that connects to the cart's components, three microcontrollers (black chips), six optocouplers (white chips), two fuses, status lights, and various electronics.

Two of the microcontrollers are dedicated encoder interfaces. They have custom firmware to keep track of the three encoder sensors on the apparatus and relate the information to the main controller chip. The main chip is the centre of the circuit and



(a) (b) (c) (d)

Figure 4.18: (a): The gimbal motor and the rod gyroscopic sensor (little circuit board at bottom). (b): Gimbal angular encoder sensor. (c): View of the brushless motor powering the disc mounted on the gimbal. (d): View of the disk rotation speed sensor.

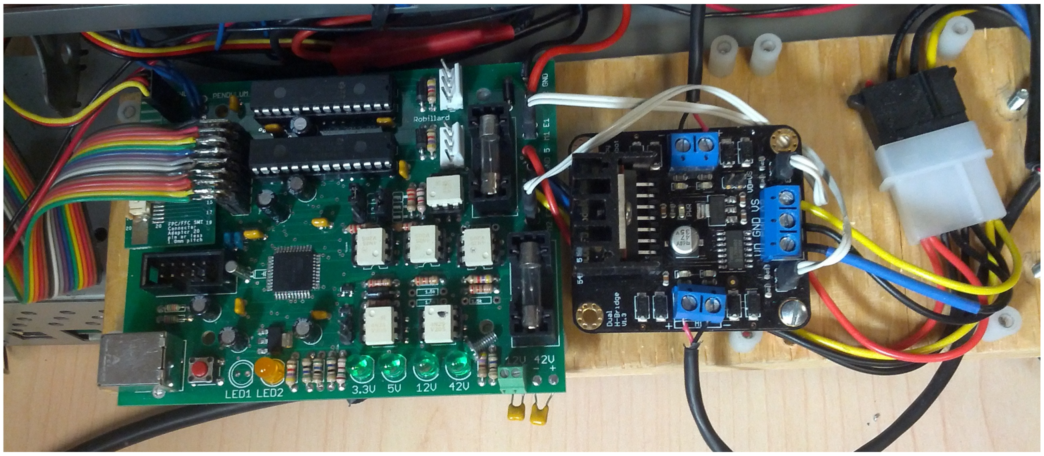
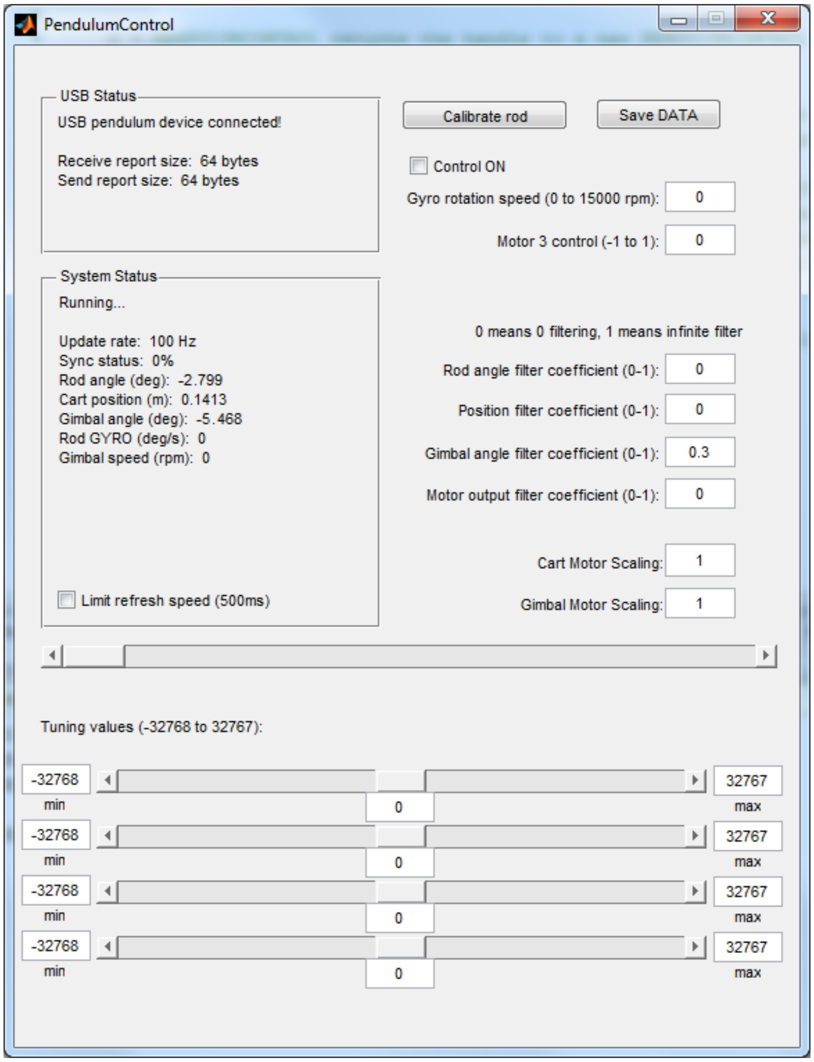


Figure 4.19: View of the control circuit board.



USB Status

USB pendulum device connected!

Receive report size: 64 bytes
Send report size: 64 bytes

System Status

Running...

Update rate: 100 Hz
Sync status: 0%
Rod angle (deg): -2.799
Cart position (m): 0.1413
Gimbal angle (deg): -5.468
Rod GYRO (deg/s): 0
Gimbal speed (rpm): 0

☐ Limit refresh speed (500ms)

☐ Control ON

Gyro rotation speed (0 to 15000 rpm): 0

Motor 3 control (-1 to 1): 0

0 means 0 filtering, 1 means infinite filter

Rod angle filter coefficient (0-1): 0

Position filter coefficient (0-1): 0

Gimbal angle filter coefficient (0-1): 0.3

Motor output filter coefficient (0-1): 0

Cart Motor Scaling: 1

Gimbal Motor Scaling: 1

Tuning values (-32768 to 32767):

min	value	max
-32768	0	32767
-32768	0	32767
-32768	0	32767
-32768	0	32767

Figure 4.20: MATLAB user interface controlling the apparatus.

connects to every other components directly. The optocouplers are used to protect the microcontroller from high voltage spikes originating from the motors and 42V power supply.

The circuit on the right hand side is a commercially available high voltage motor controller that connects to the cart and gimbal motors.

Inside the main microcontroller runs a PID control loop that regulates the disc rotation speed. The PID coefficients were manually tuned until the gyroscopic disk held a constant rpm without oscillations.

The system is controlled trough a MATLAB user interface shown in Fig. 4.20. This interface allows setting the gyroscope speed and displays a number of parameters in real-time. It also has the ability to record experimental data and tuning some parameters in real-time. The tuning of parameters is helpful during the development process. This interface reads all sensor readings, runs the control loop algorithm, and responds with motor commands at a rate of 100Hz. Refer to Appendix J for the complete circuit schematics.

4.2.2 Results of the apparatus

This section presents the results obtained with the hybrid stabilization system prototype. A number of tests were done to show the balancing characteristics under different conditions.

Modifying the gyroscope speed changes the amount of momentum H, which is a parameter in the state space model shown in Eq. 4.36. For each experiment, the LQR coefficients were kept constant, but the feedback gain $\boldsymbol{K}$ was recalculated if the gyroscope speed changed. Note that when this speed is zero, the gimbal has no effect other then being a mass.

4.2.2.1 Balancing performance

In this section we will show the balancing performance where the gyroscope is spinning at a constant speed of 10000rpm. We show the performance under steady hybrid balancing, response to step inputs, and balancing without the cart.

The steady state hybrid balancing is where both the gimbal motor and the cart motor work in tandem to keep the system stabilized. Figure 4.21(a) shows the steady state hybrid balancing with the gyroscope spinning at 10000rpm. The system is stable and the cart stays within ±1cm of its zero position. To keep the rod balanced, both the cart and the gimbal work together. The integral term ensures that on average the cart position is around zero, but since there is no integral term for the rod and gimbal angles, they are slightly offset from zero in order to compensate for sensor calibrations and centre of gravity offsets. The rod angle deflects about ±0.015 rad during steady state balancing while the gimbal deflects about ±0.15 rad.

Figure 4.21(b) shows the step response of the hybrid balancing at 10000rpm. Step inputs of 10cm every 10 seconds were applied to the reference input. The system correctly follows the reference input with some delay and an overshoot of 10%. A settling time of about one second is needed to be within the steady ±1cm error range.

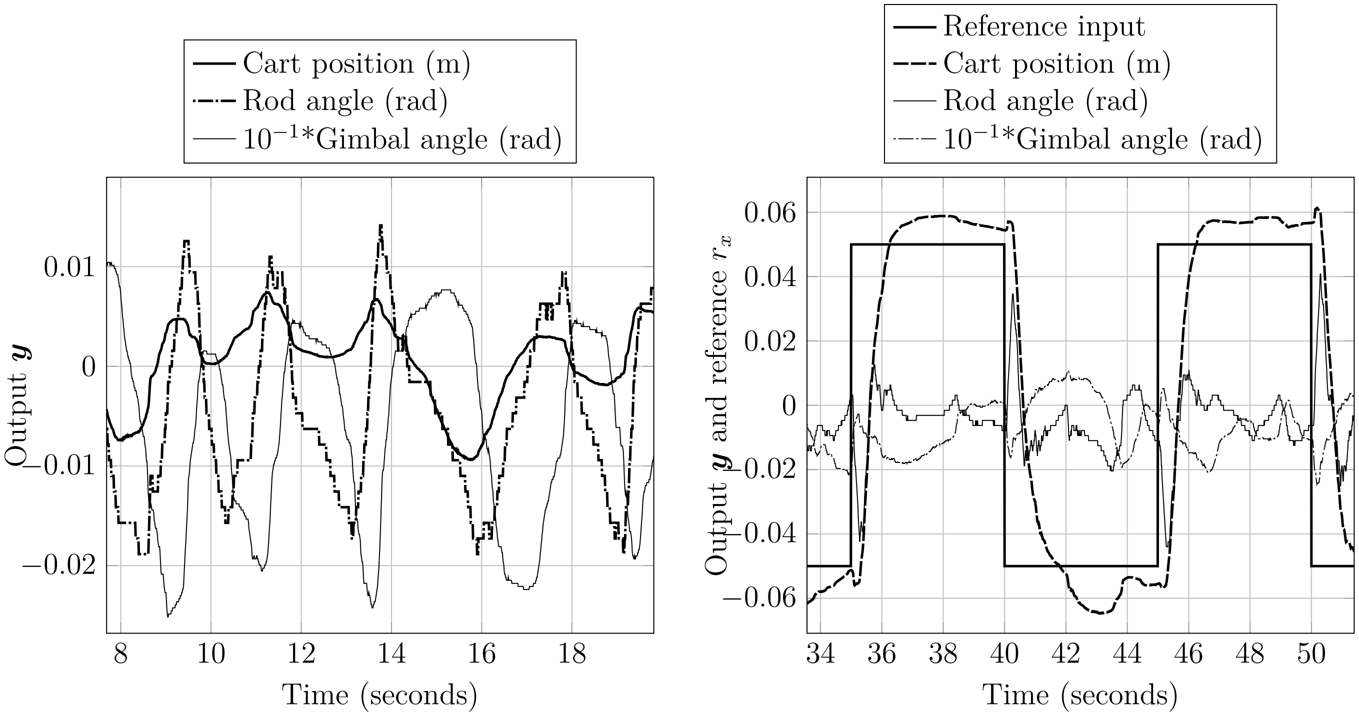
Fig. 4.21(c) shows the balancing of the hybrid control law with the drive motor disconnected. This simulates the conditions were the wheels of a self-balancing robot equipped with a hybrid balancing system would slip. The pendulum is stable and keeps its balance using just the gimbal motor as a control output. The average oscillations of the rod are the same as with the steady state balancing of Fig. 4.21(a), but since there is no balancing effort done by the cart motor, the gimbal moves further to keep the rod stable. The gimbal oscillates to about ±0.2 rad compared to the ±0.15 rad with the cart motor enabled.

This shows that a balancing robot equipped with a gyroscopic gimbal would be capable to stabilize on slippery grounds.

4.2.2.2 Effect of gyroscope rotational speed

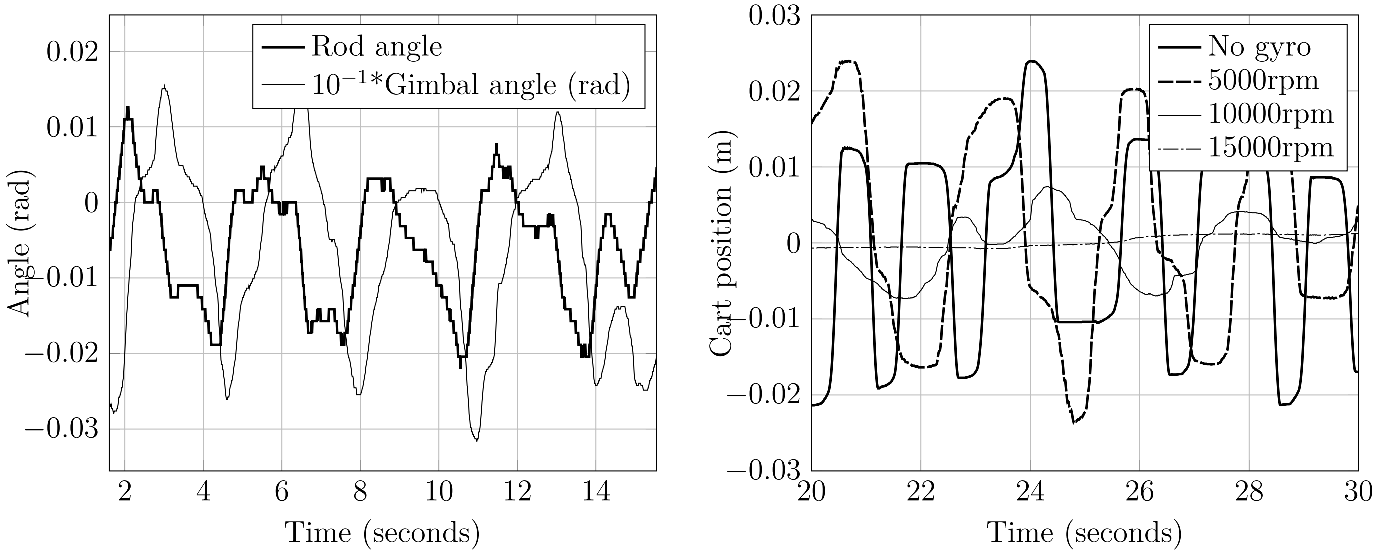
In this section, we varied the gyroscope rotation speed from 0 to 15000rpm to show its effect on stability.

The first test compares the effects of the gyroscope speed on steady balancing. Figures. 4.21(d) to 4.22(b) show the cart's position, the rod's angle, and the gimbal's



(a) Steady hybrid balancing at 10000rpm.

(b) Hybrid balance step input at 10000rpm.



(c) Balancing with cart disabled at 10000rpm.

(d) Balancing at various gyroscope speeds (cart position).

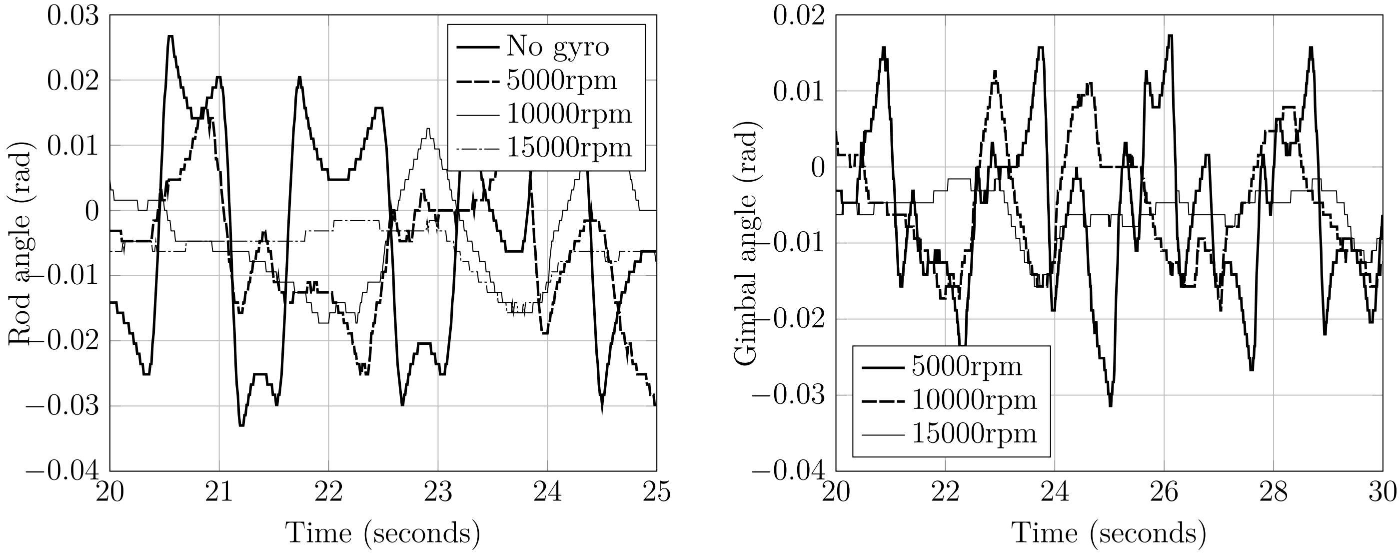
Figure 4.21: Apparatus results part 1/2.

angle respectively. As a general observation, increasing the gyroscope rotation speed correlates to an increase in the overall stability of the system: the amplitude of the steady oscillations decreases.

With no gyroscope, the system works like a classic inverted pendulum and the control system is able to keep the rod balanced while keeping the cart's position within about ±2cm and the rod's angle within ±0.025 rad. With the presence of a slow gyroscope (5000rpm), the oscillations tend to have roughly the same amplitude for the cart's motion, but their frequency reduced by a factor of 1.5 to 2, giving the system a smoother balance. The real advantage of the hybrid balancing scheme starts to show with the gyroscope speed of 10000rpm. The cart's position is now kept within the ±1cm seen previously in Fig. 4.21(a) and the rod angle oscillation amplitude is decreased.

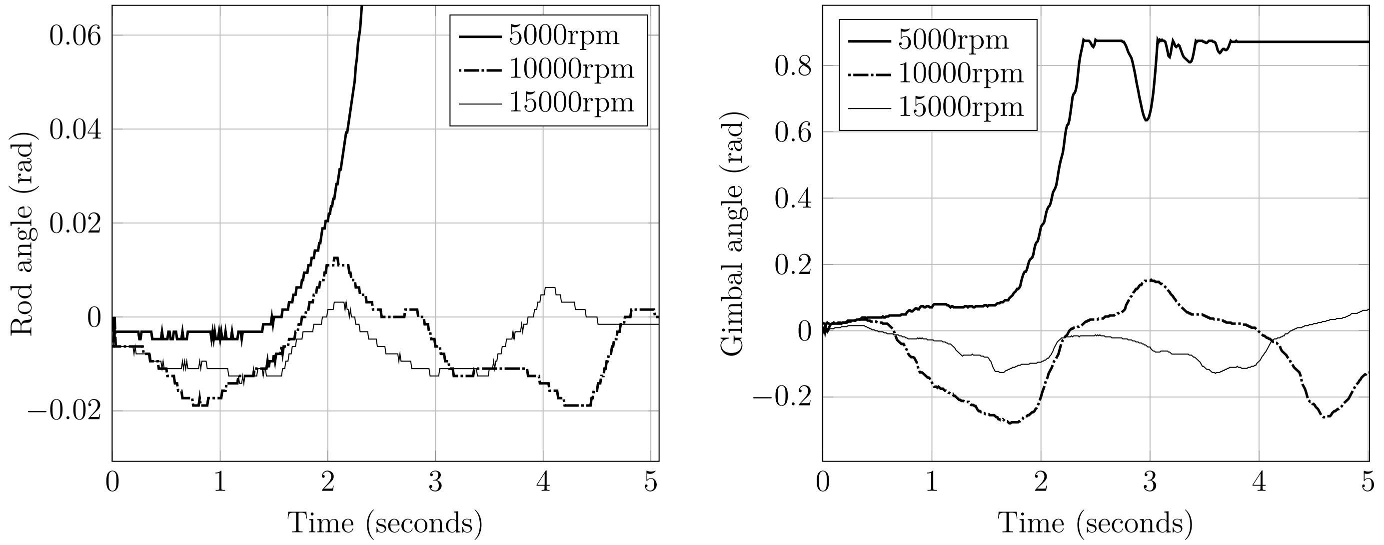
We showed the stability improves with the presence of the hybrid control scheme, but the decrease in oscillation amplitude is not proportional to the speed of the gyroscope. At 15000rpm, the cart's position error is kept within ±1mm, an order of magnitude less then at 10000rpm. The frequency of the cart's oscillations also decreased by a factor of five. In fact, at 15000rpm, the cart does little effort to keep the rod balanced. Also, because of the increase in momentum of the gyroscopic wheel, the gimbal requires a smaller deflection to keep the rod balanced, improving the robustness of the system to external disturbances.

Figures. 4.22(c) and 4.22(d) show the results for balancing without the cart's motor, using only the gimbal as an actuator. We previously observed in Fig. 4.21(c) that the gimbal oscillates to about ±0.2 rad at 10000rpm. At 5000rpm, the system is unstable and falls down. There is not enough momentum stored in the gyroscope to create the necessary precession torques to balance (without help from the cart). We see the gimbal bounces at a plateau of 0.9rad, which is the angle of the mechanical stop. At 15000rpm the gimbal oscillations are reduced to less than ±0.1 rad amplitude. Considering the gimbal has amplitude limits, a faster gyroscope speed corresponds to a stronger disturbance rejection, because less gimbal motion is needed.



(a) Balancing at various gyroscope speeds (rod angle).

(b) Balancing at various gyroscope speeds (gimbal angle).



(c) Cart disabled at various gyroscope speeds (rod angle).

(d) Cart disabled at various gyroscope speeds (gimbal angle).

Figure 4.22: Apparatus results part 2/2.

4.3 Discussion

4.3.1 Implementation issues with the Experimental device

A number of issues were encountered during the prototyping process. Only once all these issues were resolved could the system be successfully stabilized. Many of these problems occurred simultaneously causing difficulties in the troubleshooting process. In fact a tremendous amount of time was spent on solving these issues and contributed to the extensive time spent on the development of this thesis work. The next few paragraphs outline the main issues that needed to be solved.

Modelling issues:

The nonlinear model that was developed for the simulation does not include all parameters such as detailed friction models and flexibility in some mechanical parts that affect stability. Additionally, some parameters such as damping coefficients were difficult to obtain and had to be estimated since they would require specialized equipment to measure.

Control system complexity:

Initially the goal was to use an observer to estimate the full six states using encoder readings. The first tried solution instead of using the observer was approximating the missing states by taking the derivatives of the encoder readings. This method had issues: for instance the limited encoder resolution resulted in spikes in the derivatives. A solution was to add a low pass filter to smooth the encoder readings. This gave satisfactory results for the cart and gimbal encoders, but not for the rod because of lag associated with low pass filters. As a final solution, adding the gyroscopic sensor seen in Fig. 4.18(a) solved the issue.

Control code for three microcontrollers and MATLAB:

As explained before the system runs a control loop developed in MATLAB that communicates to the control circuits trough USB. The circuit interfaces with the various sensors and motors using a combination of three microcontrollers, all communicating with each other. This results in four different programs that must work together with precise timing.

Control loop delays:

At first the control law was computed by one of the circuit microcontrollers However it was determined that with the initial implementation with the observer, the control loop could not be run at a sufficient rate because of the limited processing power.

The next logical step was to control the pendulum from MATLAB directly, but this creates timing and communication delays. The simplest way to communicate with a computer is trough a serial port (using a USB to serial port converter). This method however is limited in communication rate and was not sufficient to successfully increase the control frequency. Then a true USB communication was developed. This uses a special hardware within the microcontroller to communicate at full USB speeds. The USB protocol is complex and custom software had to be written for this to work both with MATLAB and the control circuit.

At first when the control law was performed by the microcontrollers, the user interface in MATLAB had a real-time plot of the sensor readings. When the control was done with MATLAB, these plots had to be removed because their rendering slowed down the control loop and created issues.

Custom control circuitry:

The electronics interfacing MATLAB, the sensors, and the motors are custom designed. Therefore, with most custom circuits, troubleshooting has to be done in order to obtain a fully operational system. Some parts had to be redone as explained below:

Sensor noise:

The first version of the gimbal angle sensor was a magnetic encoder. It worked by attaching a magnet to the rotating shaft, and a fixed hall effect sensor (measuring magnetic field strength) measured the angle. This sensor had noise issues and could not be sufficiently filtered without adding too much lag. In the end, this sensor was replaced by an optical encoder.

Switching noise:

An issue that was very difficult to solve was associated with the power transistors controlling the motors. In order to vary the power to the motors, transistors switch ON and OFF at high frequency to approximate a variable voltage. The issue is that because of the high voltage and the inductive nature of electrical motors, high levels of electromagnetic radiation were emitted. This reset the microcontroller when the motor started moving.

Solution attempted include adding large capacitors at key locations in the circuit, rework grounding cables, and shielding high voltage wires. Reducing the switching frequency of the motors to 10kHz solved the resetting problems.

Cart friction: Friction in the cart rails was a significant issue. The rail was not designed to support cantilevered loads such as the gimbal assembly. When the system was in motion, the rod would oscillate sideways and because of the lever effect, the slider on the rail would stop the cart's motion due to large friction forces. The solution was to replace the plastic slider in the original printer design by ball bearings.

Disc motor weight: The first revision of the prototype used a brushed DC motor to power the gyroscopic disk. Because the motor was heavy it was difficult for the gimbal motor to counteract the resulting moment. The solution was to replace this motor by a compact brushless DC motor.

Gimbal motor control issue: It was found that the gimbal motor controller had not enough resolution to be properly controlled. This was solved by modifying the firmware in order to allow higher resolution timing of the pulses modulating the motor power.

4.3.2 Future work and improvements

Continuing development on this hybrid stabilization system would allow improving stability and robustness. In order to integrate this system as part of a real self-balancing robot, proper dimensioning of the device is required. To achieve this, a parameter sensitivity analysis should be performed. For the experimental device, a number of improvements could be applied to obtain better results and reduce the need for external sensors.

Parameter sensitivity

The following notes are mainly observations based on studying the structure of the equations of motion in the theoretical model (Section 4.1.2.2). It provides a guideline for future work on the parameter sensitivity.

In control theory, there are notions about controllability, observability, regulability, and estimability. The latter two terms are rarely used or encountered in control systems and little documentation is published on the subject. The research by Baram [34] shows their definition and how to access them. Essentially, a system is regulable if all the states can be stabilized, but not necessarily controllable in a system. There also exist notions of reachability, detectability, and constructability. Future work on studying these on the hybrid stabilization may help understanding parameter sensitivity on this system.

The gyroscope rotation speed can be varied, and this affects the H parameter in the model. If the gyroscopic wheel stops rotating, the gyroscopic effect becomes null ($H = 0$) and the system becomes a classic inverted pendulum with a static mass on the top. However, if H is infinitely high, then any motion of the cart will not affect ϕ or θ. By definition, the system will not be controllable. It will however be regulable, assuming the system starts in the equilibrium position. In other words, if the system begins in its equilibrium state, it will remain in this state indefinitely because of the infinite momentum in the gyroscope.

At very high H, instead of having a single large 7x7 state-space matrix, the system may be decoupled into two separate control systems. One controlling the cart's position, just like a normal cart system, allowing very fast response to control inputs, while the gimbal works independently to keep the rod balanced.

Experimental device

A number of design changes could be applied to the experimental device to improve its performance.

Ideally, one should use a properly designed gimbal motor. These motors are specifically designed for a similar purpose (camera stabilization on moving platforms) and would have have torque, with low friction and inertia. The motor currently used works well, but is not sized properly for the system (large, heavy, too powerful). About 5% of the maximum motor torque is used to balance the system.

Friction in the cart proved to be a large problem. In [35] authors researched ways to establish a dynamic model of friction on linear carriages. They showed that adding a dynamic friction compensation model improves stability and they proved that a stick slip friction model eliminated small amplitude oscillations. This model could be applied to this test platform to improve results.

Another improvement that may require rebuilding the complete device would be to better size all components to work together. Increasing the gyroscope diameter and speed, while reducing the weight of the gimbal assembly would be beneficial.

The improvements outlined above would be necessary before considering the integration of the hybrid stabilization system into the body of a self-balancing robot.

Chapter 5

Conclusion

Balancing robots are interesting platforms because of their efficiency, high speed, slim form factor, and light weight. They feature a simple mechanical design and can change direction on the spot.

A simple and effective design that gives self-balancing robots the ability to stand-up and climb obstacles such as stairs was proposed, developed, and tested. An overview of the concept and design of the balancing control loop were covered, and a model for the analysis of stair-climbing torques in both the arm and wheels were developed. This new stair-climbing model was then validated using experimental data on a prototype robot. The development process of the robot was covered and we showed that the addition of two long arms in the centre of a four-wheel drive robot allows it to stand-up for balancing and navigate stairs. The single extra degree-of-freedom (for a total of three counting the drive train) increases the field capabilities of self-balancing robots. We showed the robot was also capable of overcoming large obstacles such a rock piles and wood crates.

Then a controller for an inverted pendulum that has a gyroscopic stabilization apparatus on top was developed. To do so, a mathematical model of the system was first obtained and linearized. A simulation for both the nonlinear and linearized models was performed and we showed that the system was unstable without control. We then designed a closed-loop optimal controller to successfully stabilize the model. A full state observer was designed to allow the estimation of missing states in the experimental platform.

We were able to demonstrate that the gyroscopic effect contributes to the overall stability of the rod. During hybrid balancing mode (both the cart and gimbal motors are activated) the rod and cart oscillations diminished drastically as the rotation speed reached 15krpm. We showed the active gyroscopic gimbal rotating at 10krpm allowed the experimental device to keep its balance with no cart motion.

Future work could see the integration of the hybrid stabilization system into the stair-climbing self-balancing robot for improving stability on slippery surfaces. This would require a proper redesign to scale the gyroscope of the hybrid stabilization system to have a strong impact in the overall robot's balance. Then, automating the stair-climbing sequence so that the user does not have to worry about making the robot fall down. Once these steps are complete, the robot would be considered well rounded and ready for some real world mobile robot tasks that cannot be handled by self-balancing platforms. These may include: rough terrain exploration, soccer playing robots, high speed travel, buildings and stairs navigation. Achieving these tasks would make this type of robot well suited for surveillance, reconnaissance, bomb disposal, remote object manipulation, entertainment, robotics research, and other applications. It could also improve the safety of self-balancing vehicles designed to carry people by preventing slips on ice or gravel.

References

[1] Thomas Braunl. *Embedded Robotics: Mobile Robot Design and Applications with Embedded Systems*. Springer, 1 edition, August 2003.

[2] Larry Matthies, Yalin Xiong, R Hogg, David Zhu, A Rankin, Brett Kennedy, Martial Hebert, R Maclachlan, Chi Won, Tom Frost, et al. A portable, autonomous, urban reconnaissance robot. *Robotics and Autonomous Systems*, 40(2):163–172, 2002.

[3] Segway - the leader in personal, green transportation. `http://www.segway.com/`, November 2012.

[4] Cieslak Patryk and Buratowski Tomasz. The mono-wheel robot with dynamic stabilisation. *Robotics and Autonomous Systems*, 59:611–619, 2011.

[5] Feng M. Q. & Kawamura T. Yamafuji, K. Robots driven by parallel bicycles. *International Journal of Mechanics and Control*, 9(2):3–11, 2008.

[6] Seonghee Jeong and Takayuki Takahashi. Wheeled inverted pendulum type assistant robot: design concept and mobile control. *Intelligent Service Robotics*, 1(4):313–320, October 2008.

[7] Y. Takahashi, H. Nonoshita, T. Nakamura, and Y. Maeda. Behavioral development of ball kicking motion of a two-wheeled inverted pendulum mobile robot. In *2010 IEEE International Conference on Fuzzy Systems (FUZZ)*, pages 1 –6, July 2010.

[8] M. Kumagai and T. Ochiai. Development of a robot balancing on a ball. In *International Conference on Control, Automation and Systems, 2008. ICCAS 2008*, pages 433 –438, October 2008.

[9] T.B. Lauwers, G.A. Kantor, and R.L. Hollis. A dynamically stable single-wheeled mobile robot with inverse mouse-ball drive. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 2884 –2889, May 2006.

[10] R. Tang and R. Green. Obstacle avoidance on a mobile inverted pendulum robot. In *Image and Vision Computing New Zealand, 2009. IVCNZ '09. 24th International Conference*, pages 254 –259, November 2009.

[11] David P. Anderson. nBot, a two wheel balancing robot. `http://www.geology.smu.edu/dpa-www/robo/nbot/`, November 2012.

[12] Tao Feng, Tao Liu, Xu Wang, Zhao Xu, Meng Zhang, and Sheng-chao Han. Modeling and implementation of two-wheel self-balancing robot equipped with supporting arms. In *2011 6th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, pages 713 –718, June 2011.

[13] T. Takei, R. Imamura, and S. Yuta. Baggage transportation and navigation by a wheeled inverted pendulum mobile robot. *IEEE Transactions on Industrial Electronics*, 56(10):3985 –3994, October 2009.

[14] Pom Yuan Lam. Gyroscopic stabilization of a kid-size bicycle. In *2011 IEEE 5th International Conference on Cybernetics and Intelligent Systems (CIS)*, pages 247–252, September 2011.

[15] Chen-Yuan Chen, Bih-Yaw Shih, Chia-Hung Shih, and Li-Hui Wang. Enhancing robust and stability control of a humanoid biped robot: system identification approach. *Journal of Vibration and Control*, April 2012.

[16] Shouhong Miao and Qixin Cao. Modeling of self-tilt-up motion for a two-wheeled inverted pendulum. *Industrial Robot: An International Journal*, 38(1):76–85, January 2011.

[17] "Gyro monorail - Wikipedia, the free encyclopedia." [Online]. Available: `http://en.wikipedia.org/wiki/Gyro\_monorail`. [Accessed: 06-Apr-2012].

[18] Seakeeper inc. `http://www.seakeeper.com`.

[19] Sung Kyu Ha, Dong-Jin Kim, and Tae-Hyun Sung. Optimum design of multi-ring composite flywheel rotor using a modified generalized plane strain assumption. *International Journal of Mechanical Sciences*, 43(4):993–1007, April 2001.

[20] Hiroshi Yabuno, Hirokazu Takano, and Hirokazu Okamoto. Stabilization control of hunting motion of railway vehicle wheelset using gyroscopic damper. *Journal of Vibration and Control*, 14(1-2):209–230, January 2008.

[21] Quy-Hung Vu, Byeong-Sang Kim, and Jae-Bok Song. Autonomous stair climbing algorithm for a small four-tracked robot. In *International Conference on Control, Automation and Systems, 2008. ICCAS 2008*, pages 2356–2360, October 2008.

[22] P. Ben-Tzvi, Shingo Ito, and A.A. Goldenberg. Autonomous stair climbing with reconfigurable tracked mobile robot. In *International Workshop on Robotic and Sensors Environments, 2007. ROSE 2007*, pages 1–6, October 2007.

[23] Brian K. Taylor, Stephen Balakirsky, Elena Messina, and Roger D. Quinn. Modeling, validation and analysis of a whegs robot in the USARSim environment. In *Proceedings of SPIE 6962, Unmanned Systems Technology X, 69621B*, pages 69621B–69621B, April 2008.

[24] stairBOT - stair climbing robot. `http://www.stairbot.de/en\_beschreib.htm`.

[25] "Physics plugin for Google Sketchup." [Online]. Available: `https://code.google.com/p/sketchyphysics/`. [Accessed: 06-Apr-2012].

[26] Digital files associated with this thesis available for download. The link points to a .zip file and is password protected. The password is "g48sn29". `http://goo.gl/QlVC3M`.

[27] Raymond T. Stefani, Bahram Shahian, Clement J. Savant, and Gene H. Hostetter. *Design of Feedback Control Systems.* Oxford University Press, USA, 4 edition, August 2001.

[28] Arthur E. Bryson Jr and Yu-Chi Ho. *Applied Optimal Control: Optimization, Estimation, and Control.* Hemisphere Publishing Company, 1975.

[29] Precision mechanical components timing belts pulleys spur gears couplings bearings sprockets retaining rings brakes helical gears clutches universal joints. `http://www.sdp-si.com/`.

[30] RobotShop | robot store | robots | robot parts | robot kits | robot toys. `http://www.robotshop.com/ca/home.html`.

[31] Katsuhiko Ogata. *Modern Control Engineering.* Prentice Hall, 4 edition, November 2001.

[32] Ferdinand P. Beer. *Mécanique de l'ingénieur : dynamique : Volume 2.* Chenelière, January 2005.

[33] Kevin Goheen. *Lecture Notes MECH 5504: Guidance, Navigation, and Control.*

[34] Y. Baram and T. Kailath. Estimability and regulability of linear systems. *IEEE Transactions on Automatic Control*, 33(12):1116–1121, December 1988.

[35] Sue Ann Campbell, Stephanie Crawford, and Kirsten Morris. Friction and the inverted pendulum stabilization problem. *Journal of Dynamic Systems, Measurement, and Control*, 130(5):054502–054502, August 2008.

[36] Canada Mortgage and Housing Corporation. About your house: Preventing falls on stairs, 2010.

[37] Stairway Manufacturer's Association. SMA - home. `http://www.stairways.org/`.

Appendix A

Stair dimensions and standards

To successfully build stair-climbing robots that can be used outside of laboratory conditions, it is important for the robot to adapt for a wide range of stair dimensions. Most of the stairs in Canada are built following the National Building Code of Canada. In order to understand the code, a few terms need to be defined. These are illustrated in Fig. A.1.

- Nosing: The front or leading edge of a stair tread. In most home stairs, it projects over the tread below.
- Rise: The vertical height of a step.
- Riser: The vertical component of a step. There are two types of risers: closed risers (where the back vertical portion of the step is solid) and open risers (where the back vertical portion of the step is open).
- Run: The horizontal distance measured from riser to riser.
- Tread: The horizontal part of a step.
- Landing: flat section at the top and bottom of a staircase.

The National Building Code of Canada permits a maximum (200mm [7-3/4in.]) and a minimum (125mm [5in.]) rise; a maximum (355mm [14in.]) and a minimum (210mm [8-1/4in.]) run; and a maximum (355mm [14in.]) and a minimum (235mm

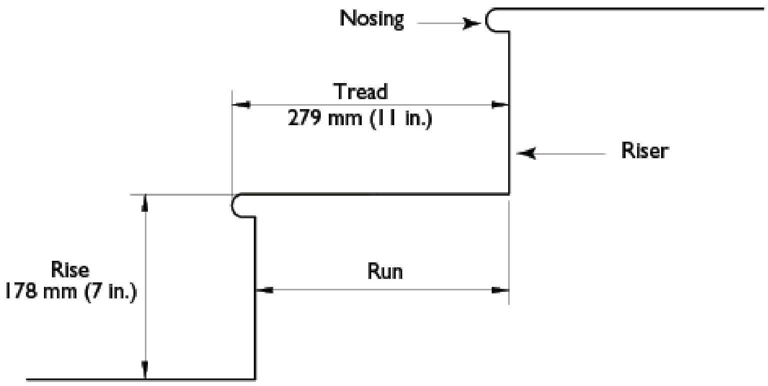


Figure A.1: Nomenclature for the elements of a step [36].

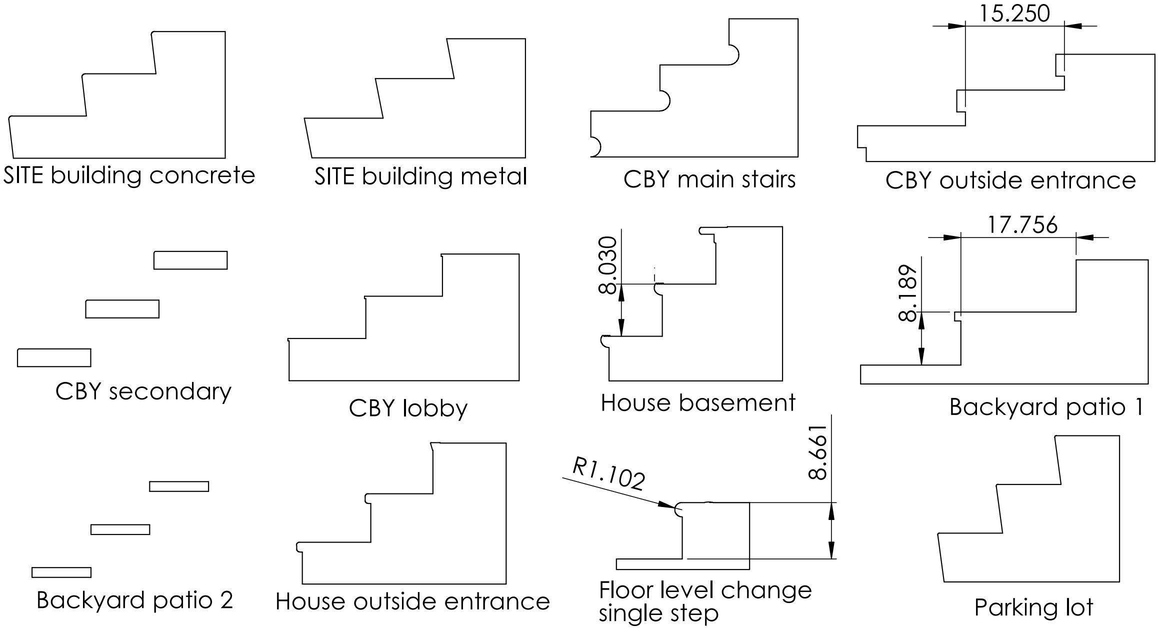


Figure A.2: Layouts of stairs found at various locations on the University of Ottawa campus and at residential places.

[9-1/4in.]) tread depth [36]. Also the landing should be at least 860mm (larger for public stairs).

Although information is hard to obtain without purchasing material from the building code, there seems to be general rules to follow in the construction that are internationally accepted. According to the stairway manufacturer's association [37]:

- There is a 1/2in. maximum chamfer, or 9/16in. maximum radius of curvature for the nosing.
- There is a maximum slope of 2% from horizontal on the thread

The above standards are a good guide on how to build compliant stairs, but as we found they are not always the norm for existing stairs. Figure A.2 shows the layout of stairs found around the campus and other locations. The places that have dimensions are the ones found to be outside the specifications of the code. Ideally, the robot developed will be able to handle all these stair scenarios.

Appendix B

Author's previous work on self-balancing robots

B.1 Overview

In 2008, D.Robillard developed a self-balancing robot using a low-cost gear motor and a breadboard was used to build the circuitry. This robot showed that it was possible to build a very low cost inverted pendulum.

Once the prototype worked well, a professionally looking and more reliable version was developed with a few design tweaks. Figure B.1 shows the assembled robot. All the electronics were encased in a space efficient package.

The first objective of this robot was reducing the cost while keeping the look of a professionally finished product. Obtaining a high performance robot (high stability, strong disturbance rejection, and fast travel) would have cost many times more. In fact, the cost breakdown of this robot is less than \$200 (Table B.1).

The second objective, somewhat in relation with the first was to balance the robot without the use of an accelerometer. Most self-balancing robots currently combine the readings of a dual axis accelerometer with the readings of a gyroscope to obtain an estimate of the body's tilt angle. This version uses only a gyroscope to keep its balance, a feature that none of the robots found in the literature seem to do. This helped save cost and processing power.

B.2 Mode of operation

All self-balancing robots have a similar principle of operation. They keep their balance by rotating their wheels based on the tilting motion of their body. Analogous to balancing a broom in the palm of one's hand, if the robot falls to the front, it will catch itself by accelerating forward.

For this to work in practice the robot requires a few key components:

Figure B.1: Assembly of the self-balancing robot

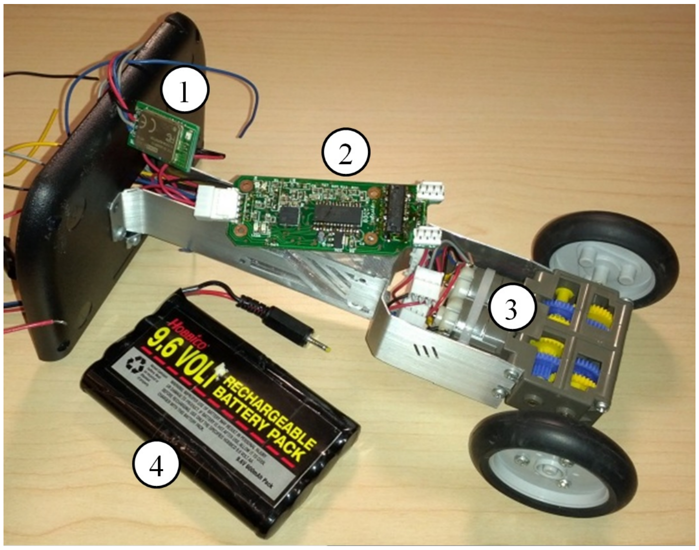


Figure B.2: Robot's internal components. 1: wireless control module. 2: main circuit board. 3: gearbox and motors. 4: battery.

Table B.1: Symbol definitions and system parameters used in the simulation.

Description	**Store**	**Model**	**Price**
Dual Motor Gearbox	RobotShop.ca	RB-Tam-01	\$9.79
Wheels	RobotShop.ca	RB-Tam-26	\$7.35
Head Enclosure	Polycase.com	AG-54MBR	\$7.16
Battery	hobbiesr.com	HCAP6005	\$25.99
Gyroscopic Sensor	Digikey.ca	497-11079-1-ND	\$4.94
Microcontroller	Digikey.ca	DSPIC33FJ128MC802	\$7.13
Motor Driver	Digikey.ca	296-9911-5-ND	\$2.48
Voltage Regulator	Digikey.ca	LM1117MP-3.3	\$1.17
Passive Components	Digikey.ca	varied	\$20.00
Bluetooth Module	semiconductorstore.com	WT12	\$26.66
PCB Circuit	batchPCB.com	N/A	\$22.50
Encoder Sensors	Digikey.ca	OR637CT-ND	\$2.88
Fasteners	fastenal.com	varied	\$10.00
Laser Cut Panels	CBR Laser	N/A	\$22.00
Various Expenses			\$20.00
		TOTAL:	**\$190.05**

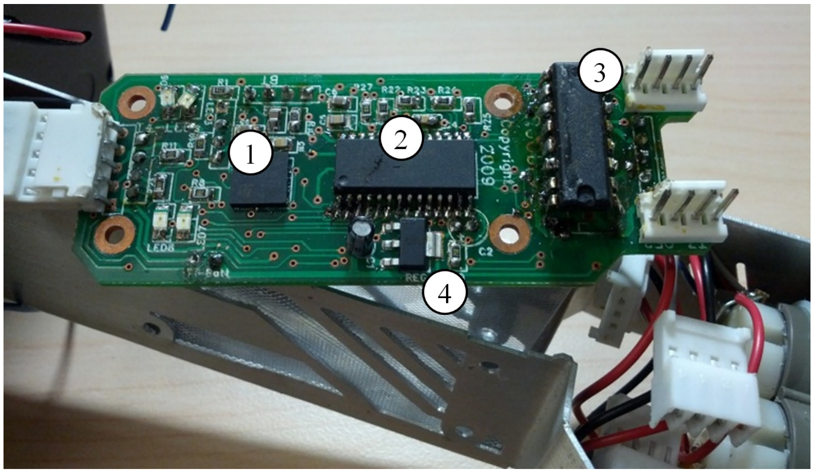


Figure B.3: Detail view of the circuit. 1: gyroscopic sensor. 2: microcontroller. 3: motor controller (power transistors arranged in an H-Bridge configuration). 4: 3.3V voltage regulator.

- A gyroscopic sensor to estimate the angle of tilt
- Wheel encoders to measure the robot's motion
- A microcontroller to run the control loop
- Motors controlled by the microcontroller
- A power source

The following explains the actual implementation of a self-balancing robot. As an example, the author's robot is used, but most self-balancing robots operate using similar hardware.

Figure B.2 shows the robot's main structure. It consists essentially of a frame on which two wheels are mounted. Each wheel is independently controlled by its own motor and gearbox. A circuit gives commands to the motors in order to keep the robot's balance. On top of the robot is a battery to provide the necessary power for the motors and circuitry.

Figure B.3 shows a close-up of the circuit. The circuit is custom made to include the key components of the self-balancing robot which are the microcontroller, the gyroscopic sensor and the motor driver.

Typically DC motors have a low torque but spin at high speed. This is why for most applications a gearbox is needed. Figure B.4 shows the Tamiya dual motor gearbox that was used in the robot. Each wheel is controlled independently to allow the robot to steer on the spot while balancing.

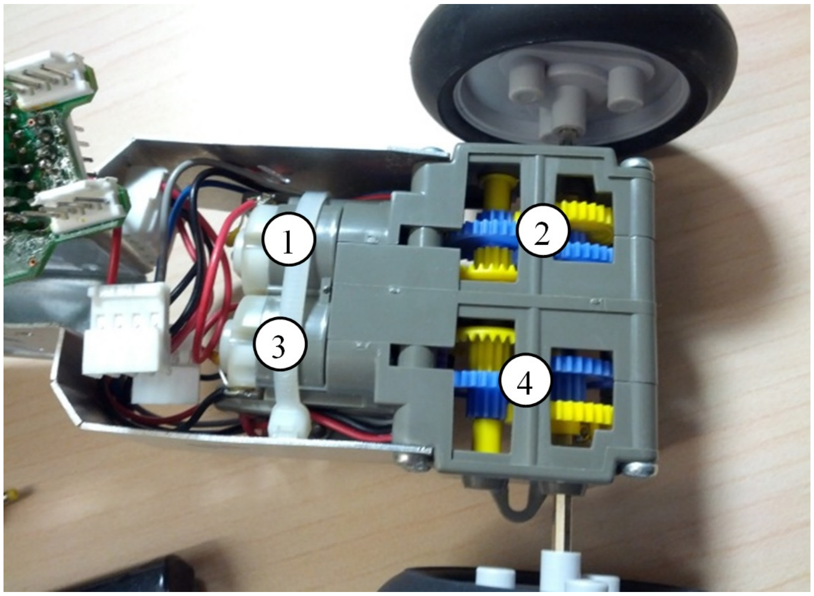


Figure B.4: Close-up of the two motors and gearbox mounted with wheels. 1: right motor. 2: right gearbox. 3: left motor. 4: left gearbox.

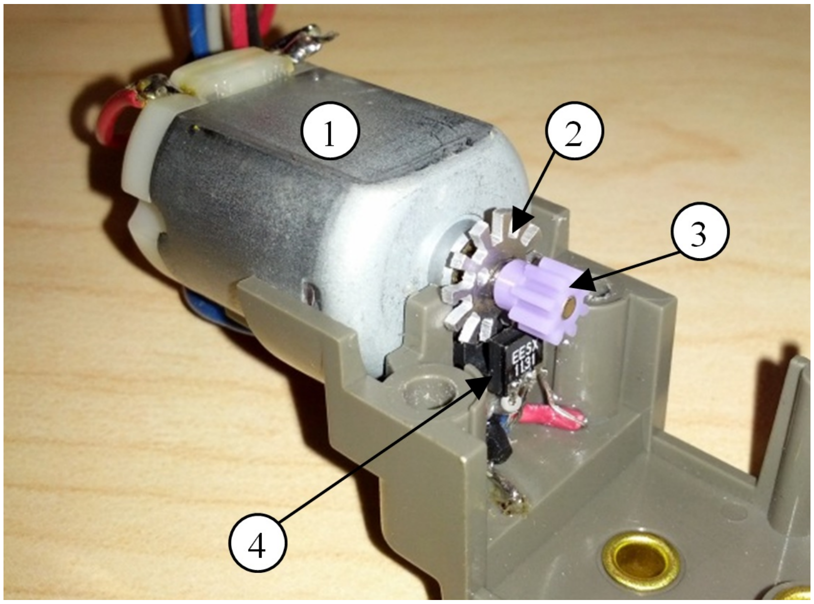


Figure B.5: View of the custom encoder added to the Tamiya motor/gearbox. 1: motor. 2: laser-cut encoder disk. 3: motor pinion. 4: optical sensor.

To effectively control the robot's balancing, the controller needs positional feedback from the motors. Unless a perfect model of the motor is available, and backlash compensation techniques are applied in the software, this is a necessity.

The motor/gearbox combo that was selected does not have any feedback so custom encoders were built for this robot. A disk with grooves (see Fig. B.5) is attached to the pinion of the motor, and an optical sensor senses the grooves of the disk. As the motor rotates, the microcontroller can count the pulses generated and keep track of the motor's position and speed.

To keep its balance, a microcontroller continuously monitors the inputs (gyroscope measurements, encoder readings) and processes the data in a control loop to calculate the necessary response of the motors. The control loop can be implemented different ways. This particular robot uses two Proportional Integral Derivative (PID) loops (one for the gyroscope and one for the encoders) and their results are added. Other possible methods that were seen in the literature would be to use state space control, fuzzy logic, or even neural networks to balance the robot.

Appendix C

Conceptual design

A summary of the conceptual design shown in this Appendix is shown in Table C.1.

C.1 Selection criteria

To perform a decision analysis for rating the various concepts, selection criteria are established.

Cost/Complexity a) Mechanical complexity: how many motors are required and how many additional components are needed to transfer the power to the right actuators? This is directly related to the cost.
b) The climbing sequence is complex if there are many steps that require timing or controlled motion to achieve a successful climb. A complex sequence means more programming.
c) The balance control system is complex if it needs to adapt to the changing configuration of the robot where its centre of mass location changes dynamically or if additional control systems are needed on top of the inverted pendulum balance.
d) The electrical systems are complex if they need additional components such as rotating connectors or slip rings.

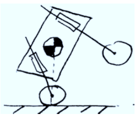
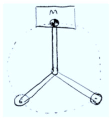
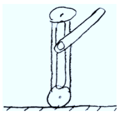
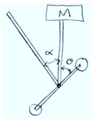
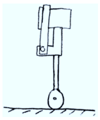
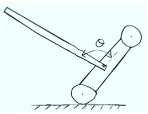
1: Lowest score 10: Highest score		**Concept 1** **4 prismatic legs**		**Concept 2** **Star configuration**		**Concept 3** **Arm on track**		**Concept 4** **Rotating base**		**Concept 5** **Piston jumper**		**Concept 6** **Long Arm**	
													
Criterion	**Weight**	Score	Weighted Score	Score	Weighted Score	Score	Weighted Score	Score	Weighted Score	Score	Weighted Score	Score	Weighted Score
Cost/Complexity	**40**	4	160	7	280	5	200	7	280	6	240	9	360
Size/Shape	**15**	8	120	8	120	4	60	5	75	9	135	8	120
Efficiency/Effectiveness	**15**	4	60	5	75	7	105	5	75	8	120	8	120
Reliability/Adaptability	**30**	7	210	7	210	4	120	9	270	8	240	8	240
TOTAL	**100**		**550**		**685**		**485**		**700**		**735**		**840**

Table C.1: Summary of the criteria weighting for the six concepts shown in Appendix C.

Size/Shape a) Storage size: can the robot fit inside a briefcase for transportation? If not could it be achieved by arms that can be easily folded or removed?
b) Ideally the robot is able to stand tall (to reach door knobs and see far) but have a slim profile.
c) Can the design easily accommodate a cargo space?

Efficiency/Effectiveness a) Sensitivity to backlash in joints while balancing. This means higher cost joints and jittery balancing.
b) How fast can be the climbing sequence?
c) Is the stair-climbing efficient? It is if most of motion contributes to increasing potential energy.

Reliability/Adaptability a) How much does it rely on friction? Would the stair-climbing work on icy steps?
b) Can it protect from damaging falls if an obstacle blocks the wheels (arm in front)?
c) How well can it adapt to various steps heights and lengths?
d) How reliable is the climbing process? What are the chances that the robot misses a step or recovers from a missed step?
e) How suitable is this robot for field applications such as surveillance? If the body constantly rotates when climbing stairs it makes it harder to add sensors and a camera. Also, if the body rotates, a manipulator arm may be difficult to install as it may interfere with the climbing sequence.

C.2 Considered concepts

First concept: four prismatic legs

Configuration This robot consists of a main body with four legs. Each leg is able to rotate relative to the main body and each leg is able to change its extension. Figure C.1 shows the possible motions of the robot.

For stair-climbing, it is sufficient that the left and right side of the robot are linked. Thus at minimum four motors are needed to control the four degrees of freedom. This means the left and right sides will always move in synchronization. Also the front

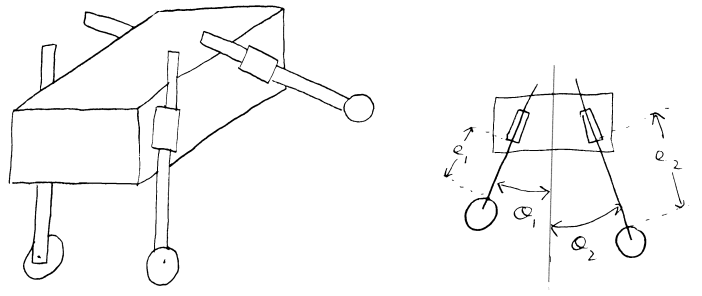


Figure C.1: Configuration of the first concept. (left): 3D view of the layout. (right): side view with available degrees of freedom.

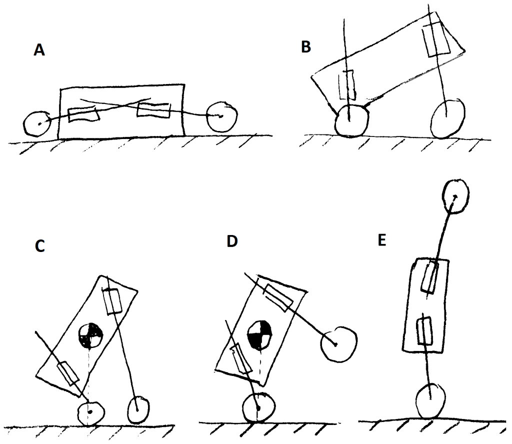


Figure C.2: Example sequence of standing up for the first concept.

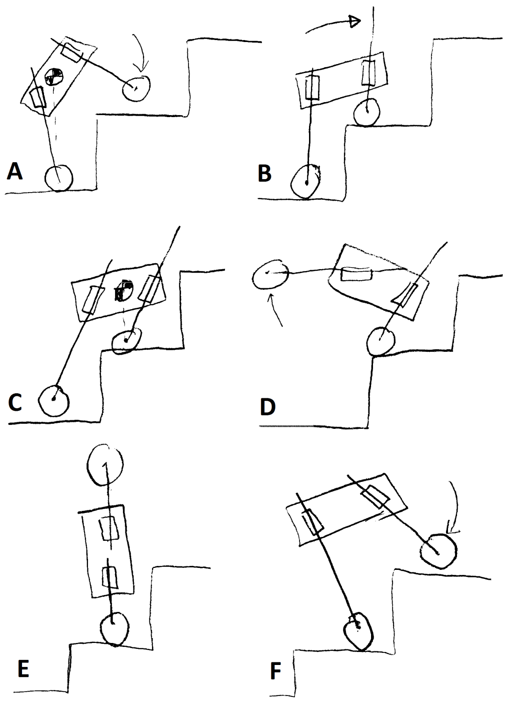


Figure C.3: Stair-climbing sequence for the first concept.

and back wheels are linked together by some sort of internal pulleys, where the left and right sides operate separately to enable the robot to steer. Alternatively each wheel could also have its own motor.

Standing up from rest To stand up from a resting position (or after the robot falls if it hits an obstacle) the arms move in order to bring the centre of mass of the robot in alignment with the contact point of one of the wheels. Figure C.2 shows how this robot would stand up.

A: Starting position.
B: Both legs rotate to put the wheels underneath the body.
C: The arms move in a way to bring the centre of gravity of the robot just above one of the wheels. The robot is now in equilibrium and can start balancing.
D: As soon as the balancing starts the other wheel raises to avoid interfering with the balance.
E: The robot can now navigate while balancing. The unused arm can be put in any

position as long as it does not touch the ground.

Climbing two steps Figure C.3 shows the typical sequence the robot would do to climb stairs. For all the concepts presented here, two steps will be shown as some designs would only work on one step because of the limited horizontal span of a step.

A: While balancing, the upper wheel approaches the top of the first step and the robot can now stop balancing by falling on the first step.
B: The robot moves its arms in order to transfer most of its weight to the other wheel.
C: The robot balances on the top wheel.
D: While steadily balancing, the lower arm rotates around the body.
E: The arm continues to rotate.
F: The arm stops just before touching the second step and the robot can restart the sequence from A.

Characteristics This design is effective at climbing stairs. It is able to adapt to a range of stair dimensions as long as the arms are able to extend more than the height of a step. However, because the driven wheels are at the end of the arms, they either all need their own drive motor, or a complex set of power transmission components within the arm. In any case, the mechanism would make the robot bulky and heavy. Because of the two degrees of freedom for each arm, any sensor or motor wires would be complicated to route.

The fact that the robot relies on a steady balance during the stair-climbing process makes the control system more complex and increases the chances that the robot tumbles down the stairs.

A nice feature of this robot is that when it is balancing, it has a free arm that can move around. This arm can be kept low in front of the robot in order to catch itself if the wheels hit an obstacle. The arm can also be raised high and could be fitted with a manipulator to open door knobs or carry a camera to see further away.

Possible variations When the robot is balancing while moving forward, it has to limit its curvature radius in order to prevent tipping sideways. A solution to this would be to allow the arm lengths to be adjustable independently on the left and right sides. This would require one additional motor to allow different lengths only for the bottom arms (see Fig. C.2-E) while balancing, enabling the robot to tilt sideways and counteract the centrifugal force.

Second concept: star configuration

Configuration Figure C.4 shows the star configuration where a central joint actuates two sets of arms and the bulk of the robot's mass at the end of another rod. Each arm is equipped with its own wheel, and like the previous concept, all the left wheels are rotated from the same motor and all the right wheels from another. Therefore the robot needs only four motors.

Standing up from rest Figure C.5 shows the sequence the robot performs to stand up. Variations from this sequence are possible.

A: Start position.
B: The two angles increase at the same rate.
C: The angles keep increasing.
D: Until about this position.
E: The mass tilts to one side (by decreasing one angle and increasing the other). The

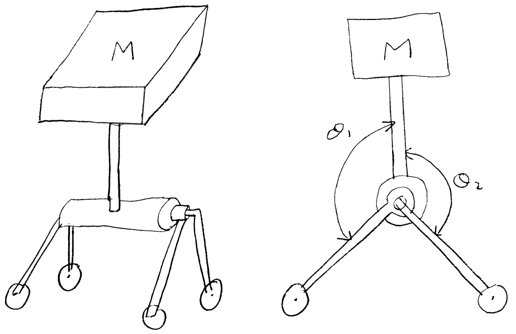


Figure C.4: (left): 3D view of the second concept. (right): side view.

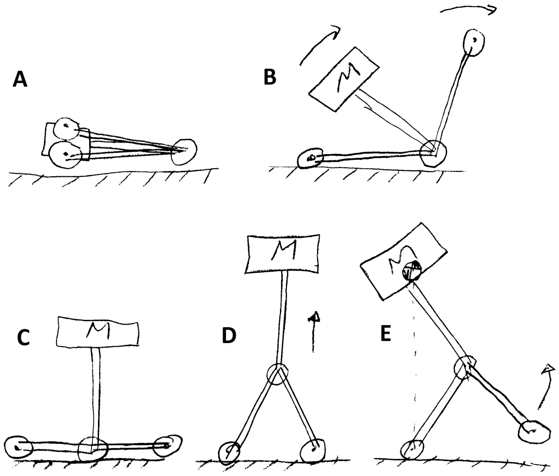


Figure C.5: The star concept standing up sequence.

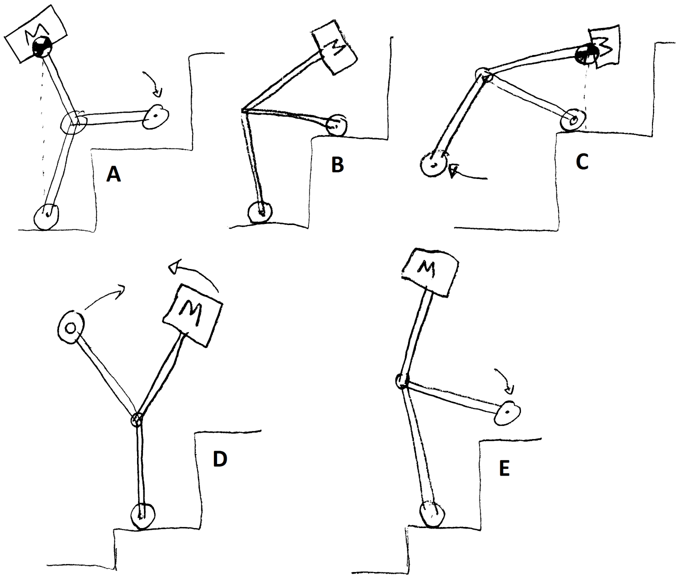


Figure C.6: Sequence of movement for the star configuration to climb steps.

robot can start balancing when the mass is above one of the wheels.

Climbing two steps Figure C.6 shows the movements required to climb stairs.

A: While balancing the robot lowers one wheel on the first step.

B: The robot moves its weight to above the wheel on the first step.

C: The robot takes balance on the first step and rotates the lower arm.

D: Past the mass.

E: Until it's about to contact the second step. At this point the sequence can be continued from A.

Characteristics Because of the absence of prismatic joints, it is relatively simple to enclose all the motors in the robot's body, where all the mass needs to be concentrated. To drive the wheels and the two joints, a combination of pulleys or chains can be used. So this design is simpler than the first concept. But thoughts must be put in the design to keep the arm's weight to a minimum.

The distance from the central pivot to the centre of gravity should be about the same as the distance to the centre of the wheel. If the mass is too far, it becomes impossible for the robot to stand up. If the mass is too close, the robot won't be able to climb a step. For this reason the robot will not work if the centre of gravity is off. This concept works on a smaller range of stair dimensions then the first one.

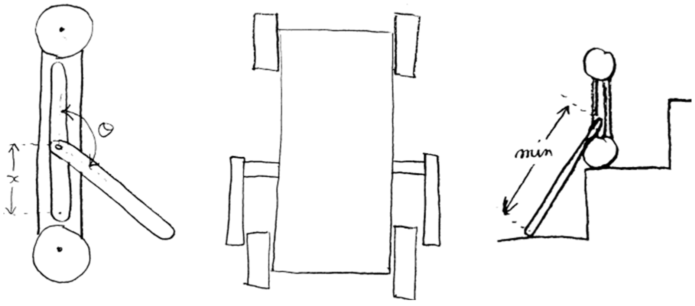


Figure C.7: (Left): side view, (Centre): front view, (Right): arm length.

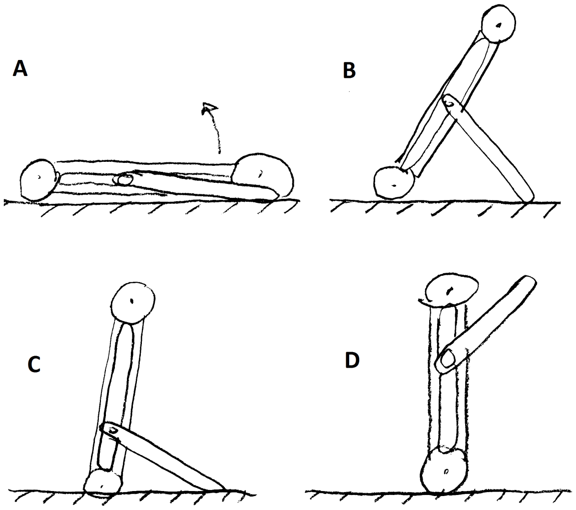


Figure C.8: Stand up sequence of the third concept.

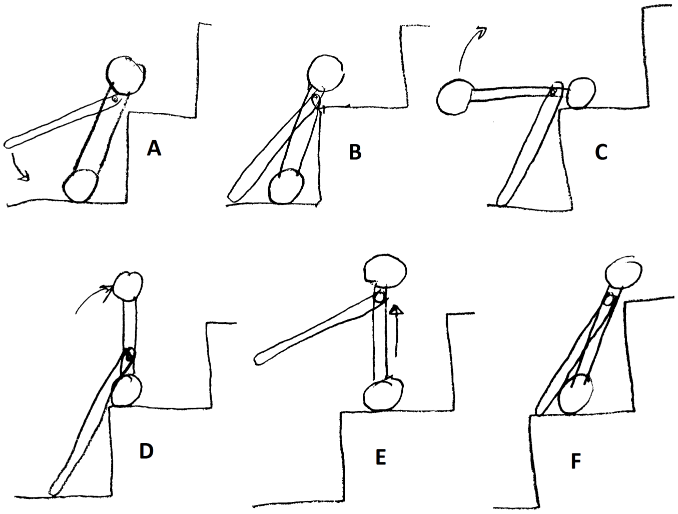


Figure C.9: The step climbing process of the third concept.

Third concept: revolute arm on a track

Configuration This concept encases the wheels into the body. A rotating arm is attached to the body, but its rotation joint is also able to move along a guide rail. Figure C.7 illustrates the concept.

As all the concepts, the left and right side wheels are controlled by two different motors to allow the robot to steer. But both the top and bottom wheels are linked to the same motor on each side.

Standing up from rest Figure C.8 shows the stand up sequence of the robot.

A: Starting position. The arm joint is about halfway between the two wheels.
B: The arm rotates to lift the robot.
C: The rotation joint may lower if needed.
D: When vertical, the robot can balance.

Climbing two steps Figure C.9 describes the sequence to climb steps.

A: The robot leans against the first step and positions the arm joint at the top.
B: The arm is rotated downwards until it touches the floor.
C: The arm keeps rotating in the same direction. Because the arm cannot sink into the floor, a reaction force will raise the robot's body.

Characteristics The length of the arm must be a minimum in order to work correctly (Fig. C.7 Right) but this minimum length can conflict with configuration in Fig. C.9-F, where an arm too long will not fit in the space available. A workaround is to increase the overall body length of the robot to allow the arm's joint to rise higher. But this can create issues at steps C and D in Fig. C.9. One possible issue is that the centre of gravity at step C is further on the left and is beyond the contact point of the arm, which would make the robot tip back. To help alleviate this last problem, most of the heavy components could move with the joint inside the robot, moving the centre of gravity forward. These specific issues make the robot function properly only on some specific stair proportions and limits its usability.

Another negative aspect of this robot is that because of the rail, most of the space within the casing is empty, but cannot be used because it has to allow the motion of the joint. This makes the robot look bulky and not space efficient. There isn't much space left for cargo too.

On the positive side, this robot can be made very robust. The arm is not very long, and most breakable components are hidden inside the body. It is also able to protect itself against falls while travelling at high speed by putting its arm in front close to the floor.

Since the drive wheels are located on the same rigid part as the body, there is no rotational joint between the drive force and the robot's principal mass. Because of this, the robot's balancing ability is not affected by backlash in joints and could be made with cheaper components.

This concept is a better performing balancing robot at the expense of a reduced stair-climbing ability.

Fourth concept: rotating base

Configuration This robot is made of a rotating base that has two wheels on each end. At the middle of the base is a joint that attaches the main mass of the robot.

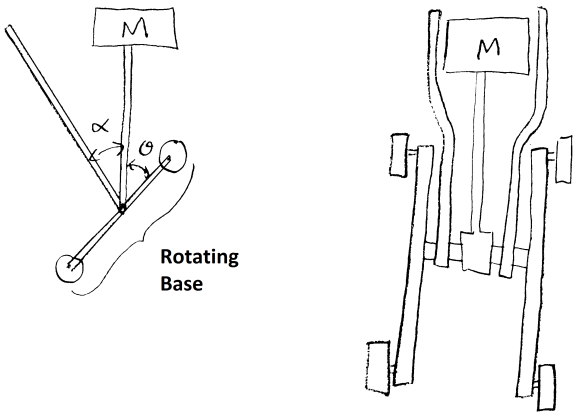


Figure C.10: Side and front views of the rotating base concept.

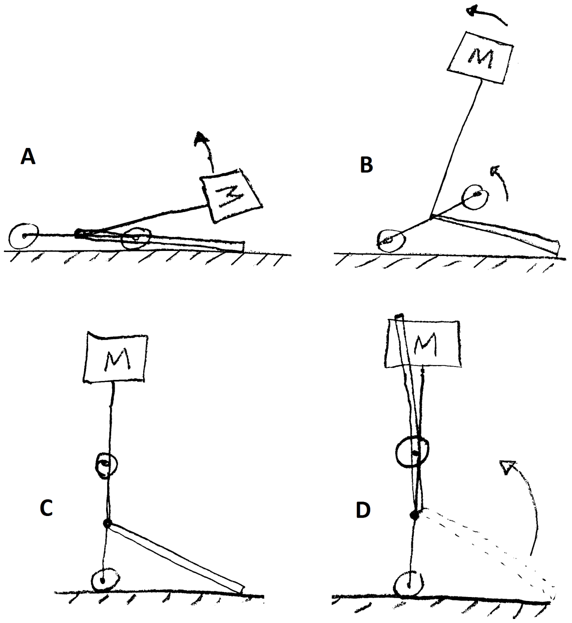


Figure C.11: Stand up sequence of the fourth concept.

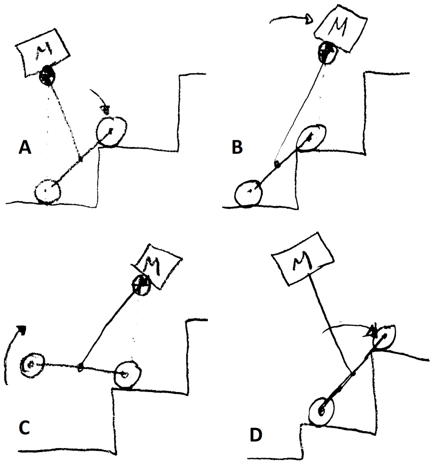


Figure C.12: Stair-climbing for the rotating base concept. The stand-up arm is not shown for simplicity.

An additional arm is needed to allow the robot to stand up, but is not used in the stair-climbing process. Figure C.10 illustrates the concept.

Standing up from rest Figure C.11 shows how the stand up arm is used.

A: The main mass moves up using the stand-up arm as a reaction force.
B: At a certain point the rotating base can also turn relative to the stand-up arm.
C: When almost vertical, the robot can start balancing.
D: The stand-up arm can be lifted away

Climbing two steps Figure C.12 shows the stair-climbing process for this robot.

A: While balancing, the robot puts one wheel on the first step.
B: The centre of gravity is shifted above the wheel on the first step.
C: While balancing the bottom wheel lifts.
D: When the wheel touches the second step, the sequence can start again from A.

Characteristics This design is interesting because the stand up and climbing sequences are simple to implement, although the robot still relies on its balancing ability to climb steps. Also the robustness of the balancing depends on the amount of backlash in the joint. With a well implemented control system, this robot has the potential to quickly climb stairs.

Because the wheels are at the ends of the arms and that most of the mass is in the top portion, there is a certain mechanical complexity. If one motor is used on each side (left/right) of the robot, there needs to be a combination of belts and pulleys to carry the motion of the motors to the wheels.

Fifth concept: piston jumper

Configuration This robot uses a piston joint to extend its legs. The piston is quick enough to actually make the robot jump into the air higher than a step. It also

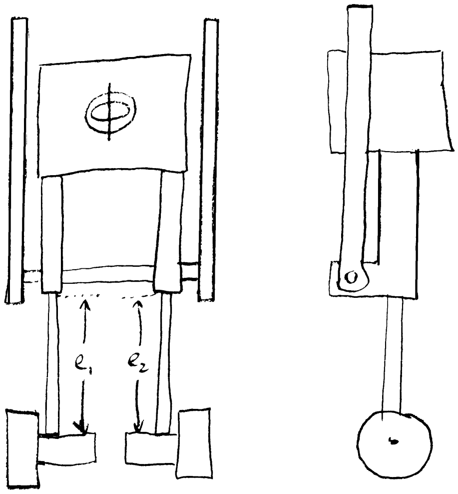


Figure C.13: Layout of the stair jumping robot.

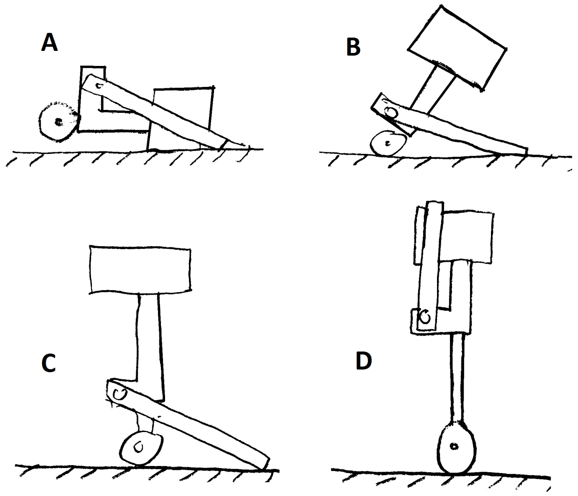


Figure C.14: Stand-up sequence of the jumping robot.

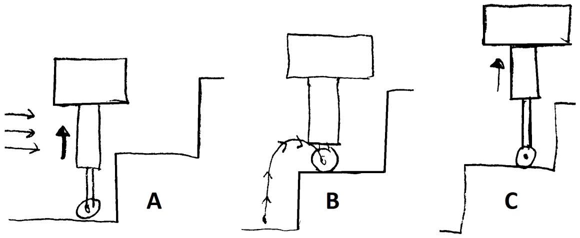


Figure C.15: Climbing sequence of the jumping robot.

has an arm to stand up and protect itself from falls. Figure C.13 shows a possible configuration.

Some sort of orientation stabilization mechanism is required while the robot is in the air. A rotating mass inside the body of the robot acts like a gyroscope and will dampen any of the robot's rotation while in the air.

Standing up from rest The stand-up sequence is the same as the other robots that use an arm to stand up. Figure C.14 shows the steps.

A-B: The stand-up arm rotates.
C: When vertical, the robot can start balancing.
D: The piston can extend to raise the robot higher.

Climbing two steps Figure C.15 shows how this robot would climb stairs.

A: While travelling towards the step, the robot activates its jumping piston just at the right moment. While in the air, the orientation stabilization mechanism is active.
B: The robot settles on the second floor and while moving forward prepares for the next jump.
C: At the right moment, the robot jumps again. This procedure can be repeated for more steps.

Characteristics With the right timing and powerful actuators this robot has the potential to be very effective at climbing stairs quickly. As long as the sensors trigger the jump at the right time the robot does not even need to stop its forward motion between steps and could go as fast as 2-3 steps per second.

This robot is also a good platform for the balance because there is no rotational joint between the main mass of the robot and the driving wheels.

One major drawback with this robot is its mechanical complexity. A powerful and fast linear piston that can take shocks and move quickly enough to jump a step requires careful engineering. It could be made with pneumatic actuators, but they would need a source of pressurized air (compressor or reservoir) and pneumatic valves. Oil pistons would be even worse as they require maintenance; leaks are messy and would not work as well as air. It could also be a spring that releases but motors or solenoids would be needed to recharge and release the spring.

There is also a difficulty associated with the control system of this robot. The balancing must be robust and sensors must be used to detect the edge of a step. The presence of the gyroscope is a source of power consumption and also affects the balancing control loop, adding complexity to it.

Possible variations Instead of having a mechanical gyroscope to stabilize the orientation while in the air, the rotational inertia of the wheels could be used. If the robot's orientation needs to be corrected in flight, the wheels could start spinning in the required direction. The mass moment of inertia of the wheels will generate a torque from the angular acceleration of the wheels and in turn will create a correcting moment on the body of the robot. Larger wheels would be more effective in this case.

Sixth concept: long rotating arm

Configuration This concept involves a single rotating arm fitted in the middle of the robot's body. This arm can indefinitely rotate around its joint and is about twice as long as the body. The body carries all the heavy components and has four wheels. Like the other concepts, the left and right sides of the robot each have their own drive

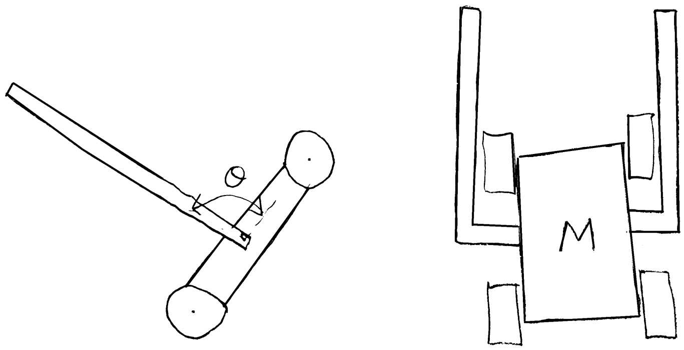


Figure C.16: Side and front views of the long rotating arm concept.

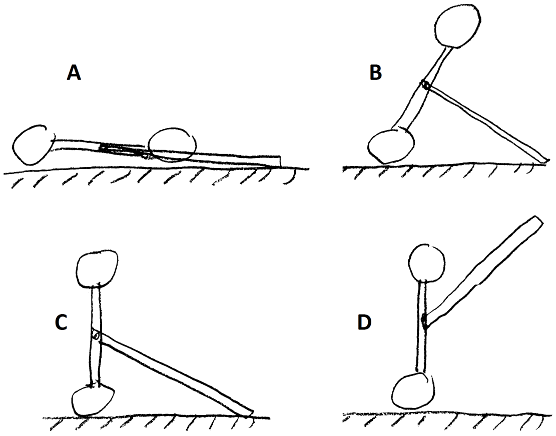


Figure C.17: Stand up sequence of the sixth concept.

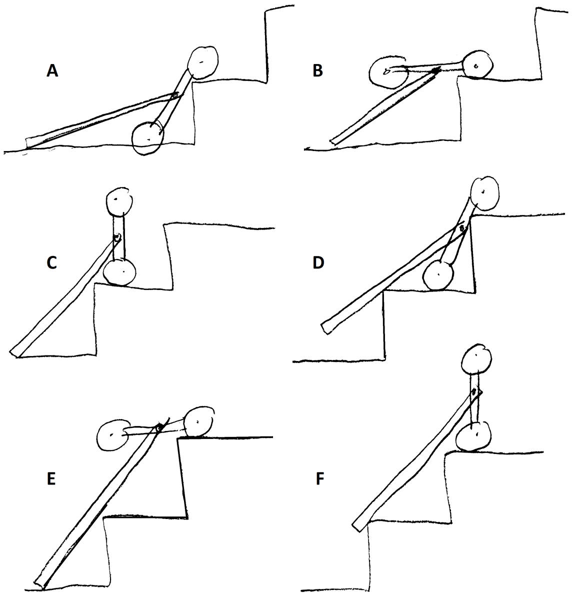


Figure C.18: Climbing sequence of the long arm concept.

motors. Figure C.16 illustrates the concept.

Standing up from rest The standing sequence is the same as with other concepts that use a stand-up arm. Figure C.17 shows how it is done.

A-C: The arm rotates and a reaction force on the floor forces the body to lift. Once vertical the robot can start balancing.
D: The arm can lift and rotate to any desired position.

Climbing two steps Figure C.18 shows how this robot can climb steps. Note that the climbing sequence is very simple to perform: just rotate the arm. The multiple images are just to show how it happens.

A: The robot lays on the first step.
B-F: The arm rotates and the reaction force will force the body to turn up.

Characteristics The whole body of the robot continuously rotates. This makes it more difficult to use sensors and cameras while the robot is climbing stairs.

For this principle to work, the arm needs to be a minimum length, which will need to be found analytically.

There is a little reliance on the friction between the wheels and the stairs. It needs to be high enough to compensate the friction force generated by the arm as it slides along the edge of the stair. This should not be a problem if the arm is long the friction force is reduced due to the leverage and the wheels are made of a high friction material like rubber. They are helped by the fact that most of the mass is above the wheels. The arm could be made of a low friction polymer and built as light as possible.

Because of the long arm towards the bottom of the stairs and because the centre of gravity is kept low, the climbing sequence is reliable. If the robot misses a step, it is almost impossible for the robot to tumble all the way down. The other concepts have a much higher chance of this happening. The reliability of the climbing sequence is further improved by the fact that the robot does not rely on the balancing ability while climbing steps. The robot can pause indefinitely in any position.

The design is simple. This is the only concept that can balance, steer, stand up, and climb stairs with only three motors total and it does it without fancy mechanisms. Balancing and climbing are two separate processes which make the control system simple to implement. The climbing sequence is easy to program: just turn the arm until the top of the stairs is reached.

C.3 Criteria weighting

For this project, time and budget is limited and the simplest option maximizes the chances of obtaining a successful model at the end of the term. For these reasons the most important aspect is the cost/complexity. 40% of the points are from this criterion. The reliability/adaptability is the second most important aspect. First, the robot needs to work. A more reliable design minimizes the chances that things go wrong forcing a redesign. 30% The other two criteria will count as 15% as they are not as critical to a successful design within the given time constraints.

C.4 Decision analysis

Table C.1 shows the results of the decision analysis. The clear winner is the sixth concept (long rotating arm) with a score of 840. This concept is the best choice for this project because of its simplicity over the other concepts.

Close behind is the fifth concept (piston jumper) with a score of 735. In fact, if time and budget was not as strict, this would make a very interesting robot with good potential. A future project with concept 5 would be interesting.

Appendix D

Additional design analysis for the development of the self-balancing stair-climbing robot

D.1 Body grousers

The initial intent was to add rubber grousers on the complete robot's body, but because of time constraints it was decided to go with aluminium grousers integrated within the structure of the robot. This way they are added to the laser cut part with no additional fabrication work.

The spacing and number of grousers was determined such that a compromise is made between how much slip there is (distance between each grouser), and the ability to grab rounded edges (enough spacing to accommodate the curvature). This could be changed on a later revision.

D.2 Arm power transfer

A means to carry the arm gear motor's torque to the arm shaft is needed. Several options were considered and the final option is to use ladder chain and sprockets.

Using a belt was considered (choosing its width so that the belt has a low safety factor in tension) but unfortunately the available minimum length was still too long to fit the design.

Using gears was also considered but it was difficult to find strong gears with a 0.5in. bore diameter to match with the gearbox, yet that are still about 1.25in. pitch diameter. The costs of these gears finally lead to the choice of chain and sprocket.

After finding a suitable sprocket drive of 1.25in. pitch diameter the required strength of the chain is evaluated as follow:

$$F = \frac{\tau_a}{r_{sprocket}} = \frac{6.23}{1.25 \cdot 0.5 \cdot 0.0254} = 392.4\text{N} = 88.22\text{ lbf} \tag{D.1}$$

Material	Price/ft	Yield Point lb.	Safety Factor
Steel	$2.92	80	0.9
Stainless Steel	$9.04	80	0.9
High-Tensile Steel	$4.15	150	1.7

Table D.1: Available size 14 chain materials.

The selected sprocket is compatible with 0.353 (size 14) ladder chain links. From SDP-SI [29], three choices of chain materials shown in Table D.1 are available.

Both the steel and stainless steel chains would be too weak and would break under high load conditions. The high-tensile steel has a factor of safety of almost two. We would not want a higher safety factor because the chain is a cheap component and is simple to replace, thus it is desirable for the chain to break under excessive loading conditions (instead of causing damage to the gearbox or other components).

D.3 Arm material and construction

Referring to the mechanical drawings of the robot in Appendix E, we use a square aluminium channel for the first portion of the arm where we find the highest bending moment. Then the tube will be made of a low friction plastic material.

It is difficult to determine exactly what material would be suitable because most plastics do not have an elastic region like most materials do. Thus manufacturers tend to not publish elastic modulus for these plastics and approximations need to be made.

A quick CFD analysis (Fig. D.1) was performed using Solidworks Xpress. The results should be taken lightly as the materials available in the software did not match exactly the one available for purchase. Also, since it is the Xpress edition of the finite element software, there are many limitations.

A number of parameters were varied before obtaining a suitable design. Requirements were that the arm would be somewhat flexible and resist bending without failing. Rigid plastics that could shatter upon failure were discarded for safety. A

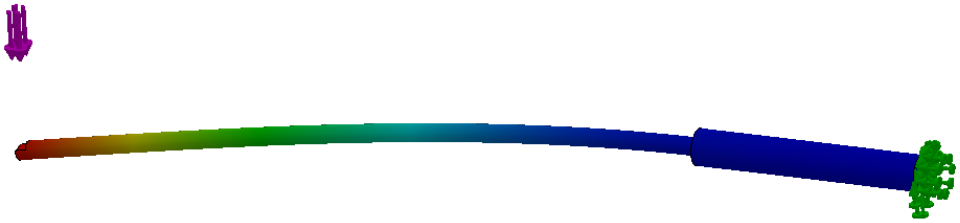


Figure D.1: View of the arm under load in simulation Xpress

low coefficient of friction is important to ensure a successful climbing operation, so PTFE or Teflon were best.

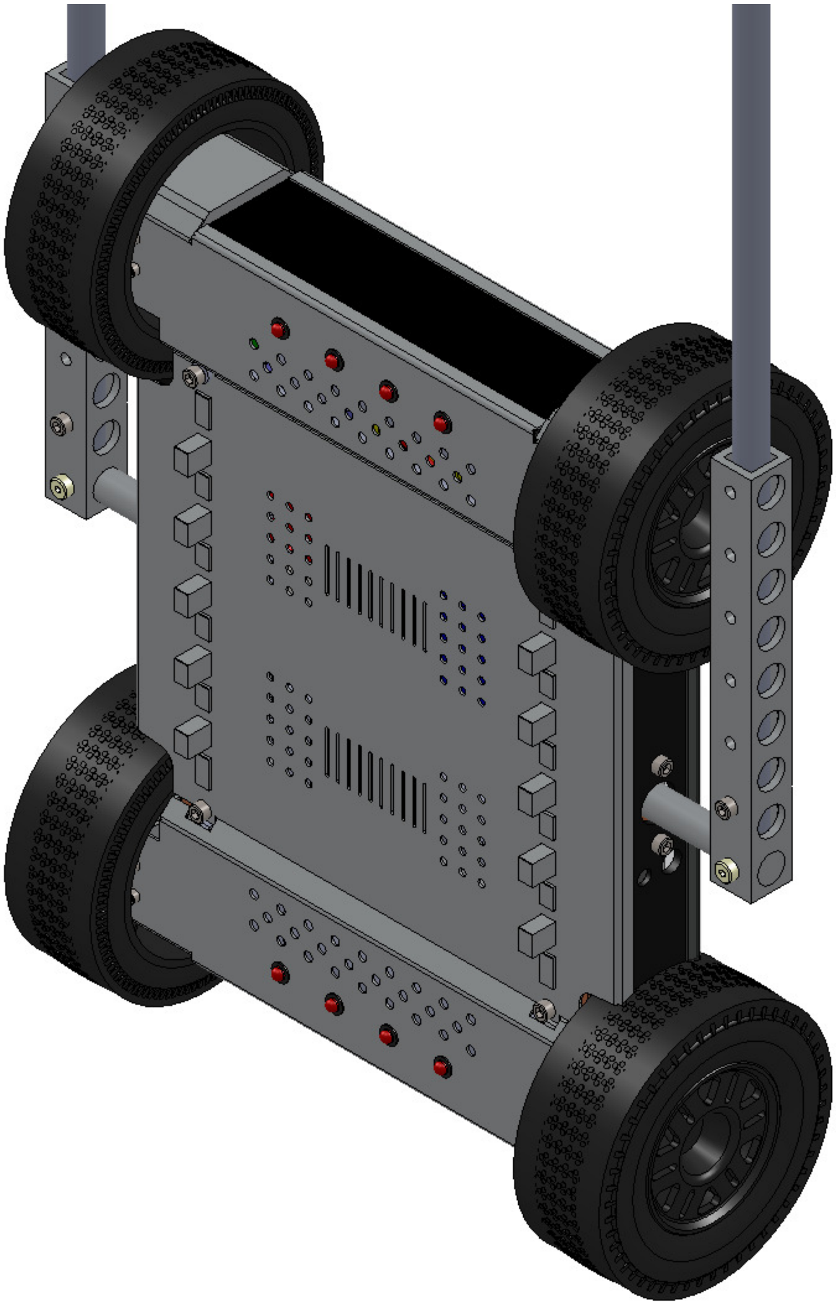
After varying the rod diameters and alternating between solid rods and tubes, it was determined that PTFE or Teflon were structurally not suitable. They would be suitable by adding internal reinforcement such as steel or carbon fibre inside the plastic tube but would have added to the complexity.

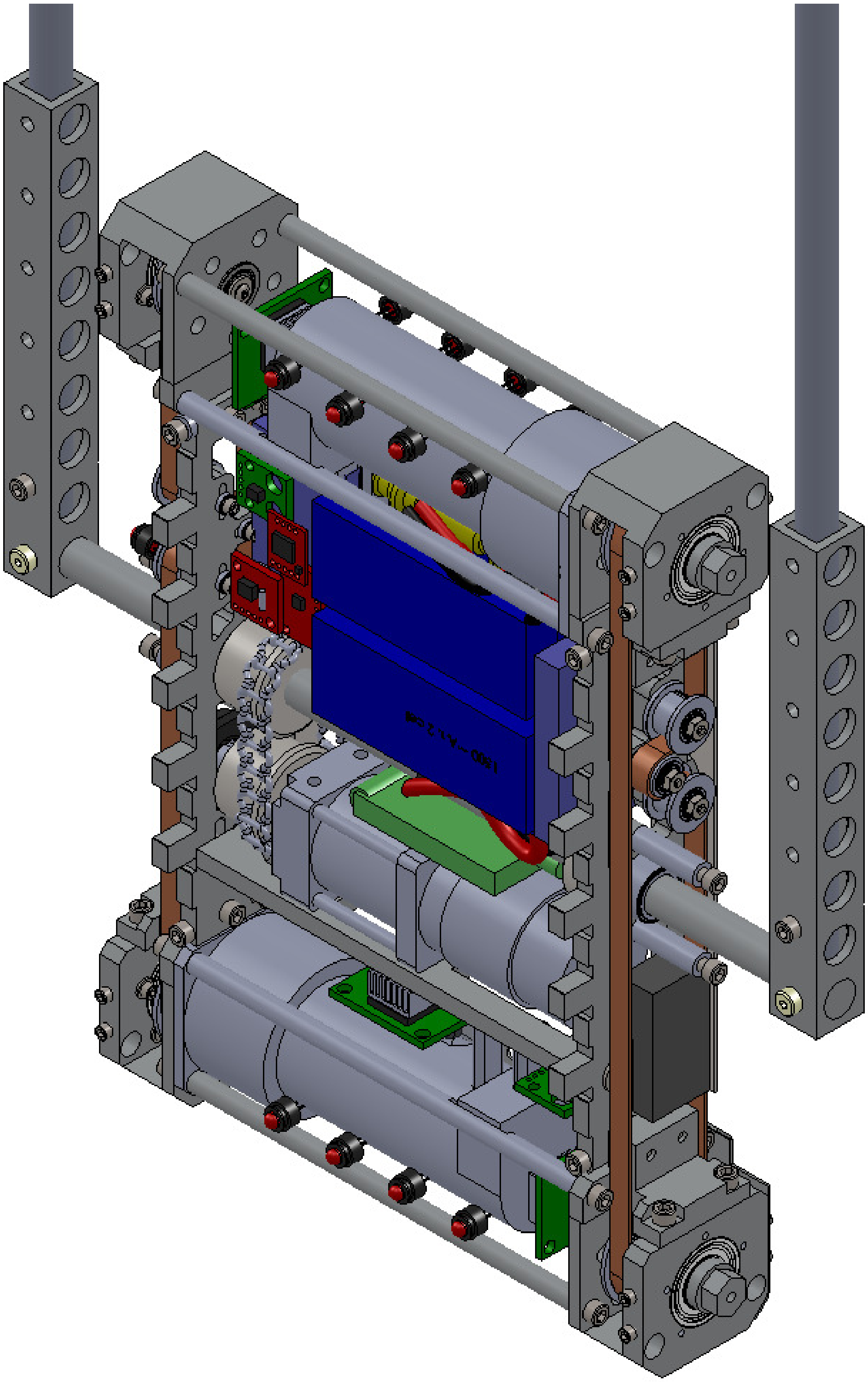
Switching to acetal provided the best results with a safety factor of about 10 against failure and a tip deflection of about 18mm.

Appendix E

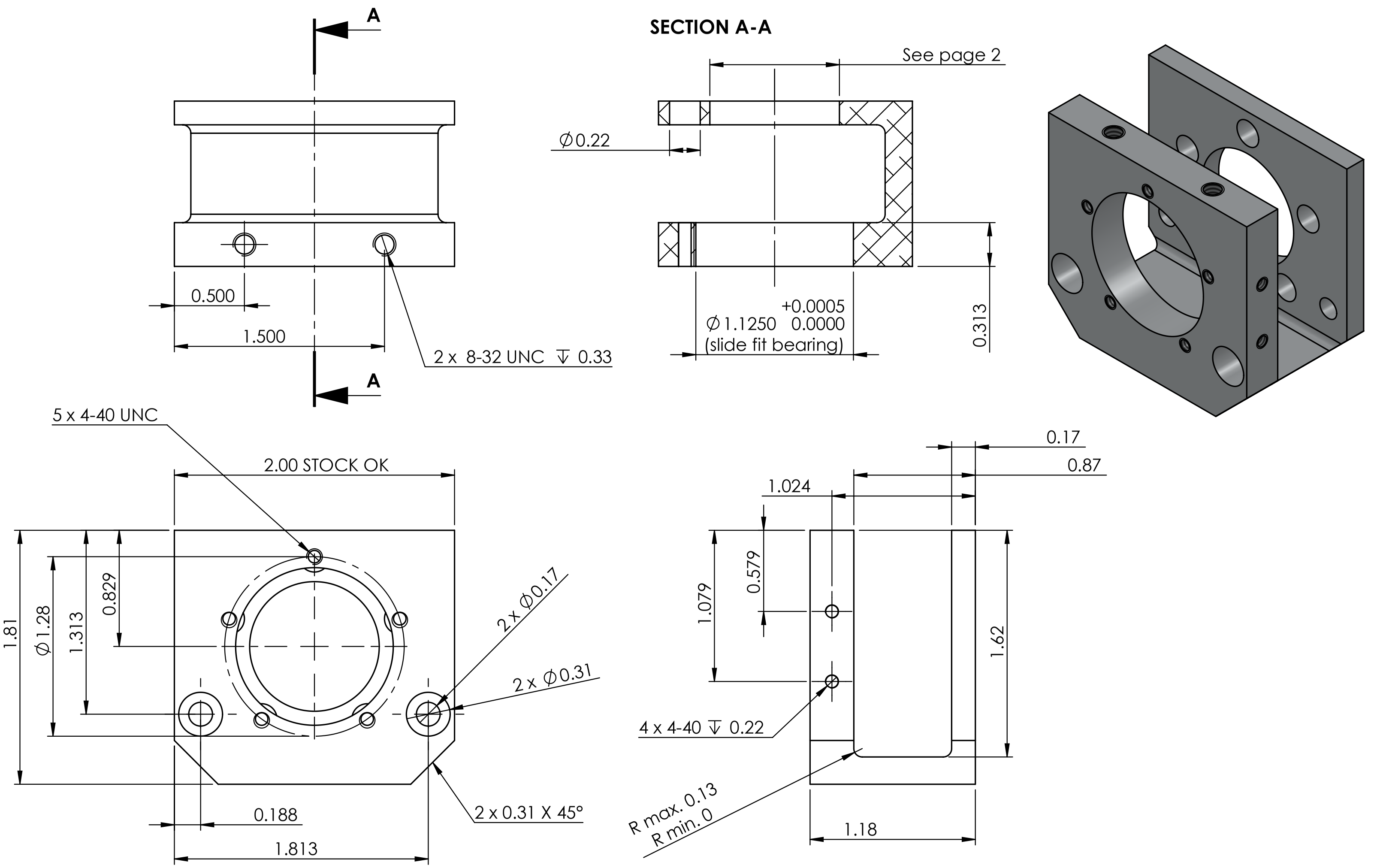
Mechanical design for the stair-climbing and self-balancing robot

The digital CAD files are also available for download here [26].





Need 8 copies. 4 of configuration 1, and 4 of configuration 2 (see page 2).



uOttawa

TOLERANCES
Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001
UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**
MATERIAL **6061 Alloy**
FINISH
PLOT SCALE: 1:1
THIRD ANGLE PROTECTION

DRAWN BY	DATE
Dominic Robillard	02/11/2012

COMMENTS:
Design of a stair climbing inverted pendulum robot. For a directed studies course in mechanical engineering.

Supervisor: Dr. Eric Lanteigne

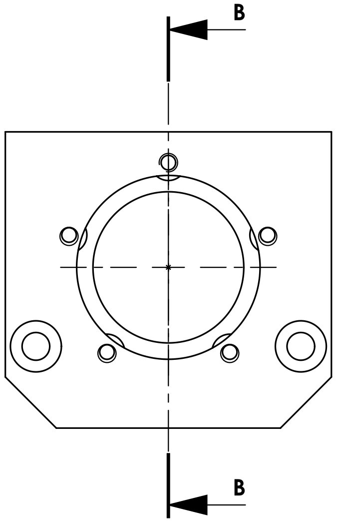
Filename:
UO-SR-001 Wheel Hub Bottom Bracket

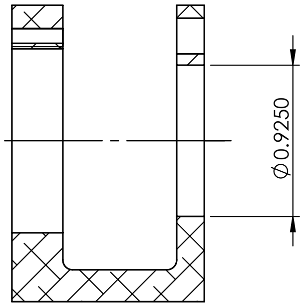
SIZE A
PROJECT: **Stair Robot**
DWG. NO. **UO-SR-001**
REV. **2**
SCALE: 1:1
WEIGHT: 60.79 g
SHEET 1 OF 3

See page 1 for rest of drawing.

Configuration 1:

Motor Side.
Need 4 copies.

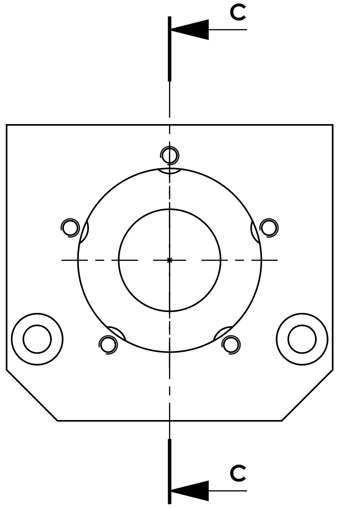


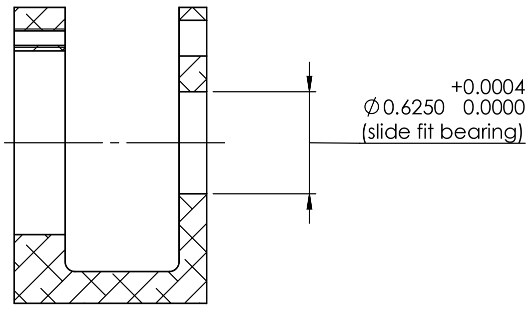


SECTION B-B

Configuration 2:

Pulley Side.
Need 4 copies.





SECTION C-C

uOttawa

TOLERANCES
Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001
UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**
MATERIAL **6061 Alloy**
FINISH
PLOT SCALE: 1:1
THIRD ANGLE PROTECTION

DRAWN BY	DATE
Dominic Robillard	02/11/2012

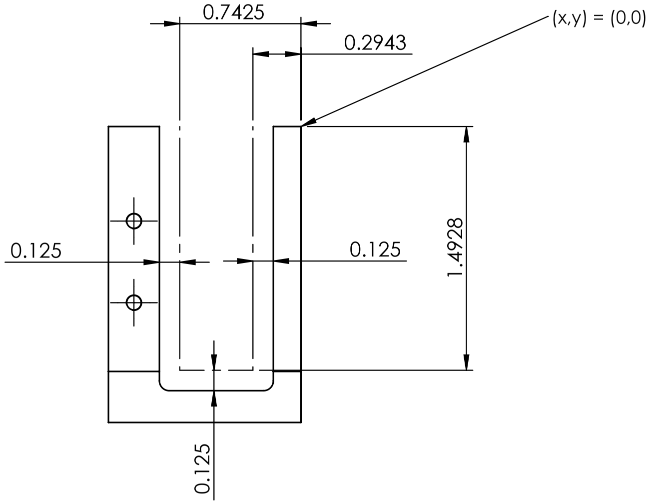
COMMENTS:
Design of a stair climbing
inverted pendulum robot.
For a directed studies course
in mechanical engineering.

Supervisor: Dr. Eric Lanteigne

Filename:
UO-SR-001 Wheel Hub Bottom Bracket

SIZE	PROJECT:	DWG. NO.	REV.
A	**Stair Robot**	**UO-SR-001**	**2**

SCALE: 1:1 WEIGHT: 60.79 g SHEET 2 OF 3



x	y
0.2943	0
0.2943	1.4928
0.7425	1.4928
0.7425	0

uOttawa

TOLERANCES

Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001

UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**
MATERIAL **6061 Alloy**
FINISH
PLOT SCALE: 1:1
THIRD ANGLE PROTECTION

DRAWN BY: Dominic Robillard
DATE: 02/11/2012

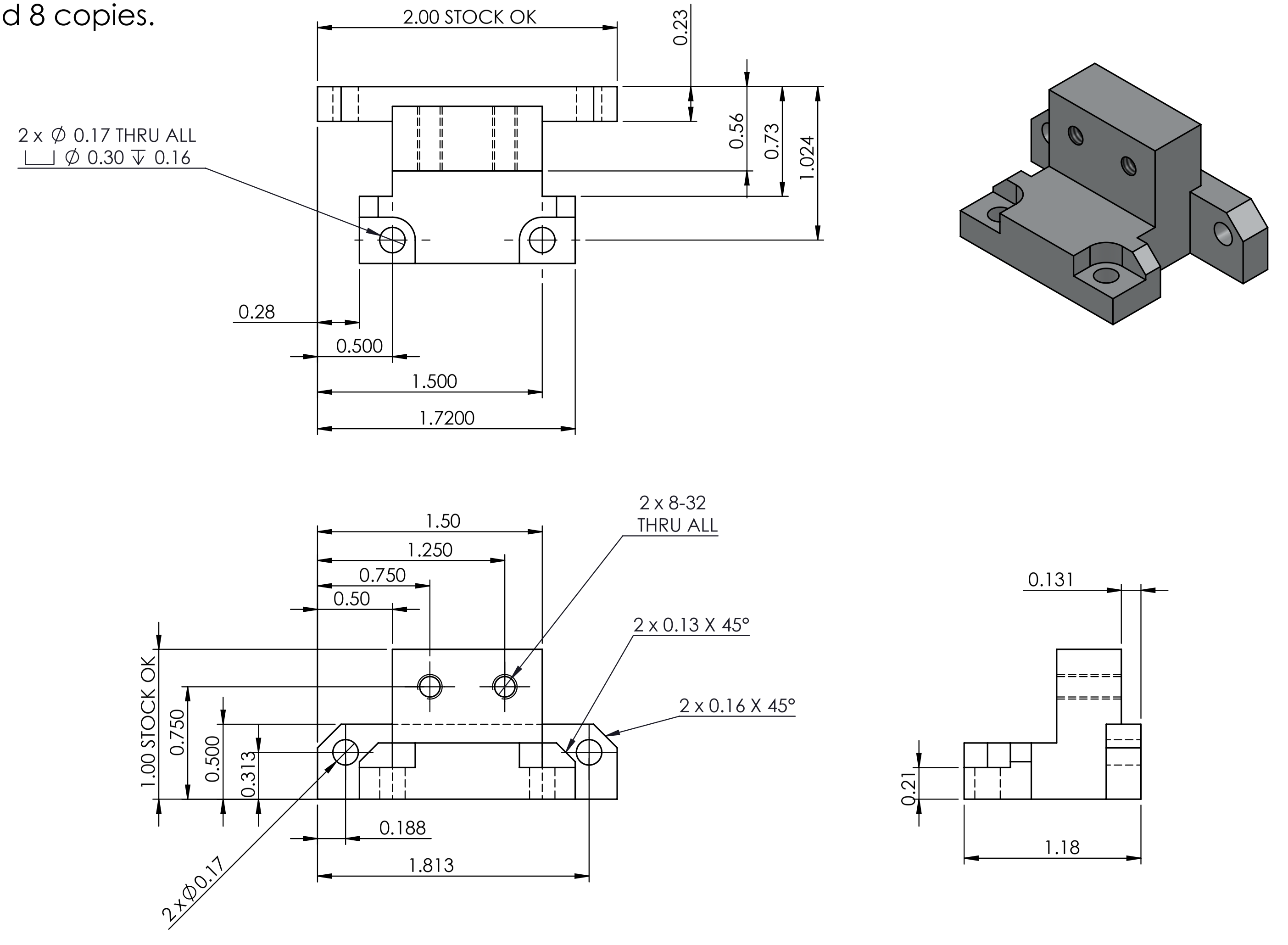
COMMENTS:
Design of a stair climbing inverted pendulum robot.
For a directed studies course in mechanical engineering.

Supervisor: Dr. Eric Lanteigne

Filename:
UO-SR-001 Wheel Hub Bottom Bracket

SIZE **A**
PROJECT: **Stair Robot**
DWG. NO. **UO-SR-001**
REV. **2**
SCALE: 1:1
WEIGHT: 60.79 g
SHEET 3 OF 3

Need 8 copies.



uOttawa

TOLERANCES

Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001
UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**
MATERIAL **6061 Alloy**
FINISH
PLOT SCALE: 1:1
THIRD ANGLE PROTECTION

DRAWN BY	DATE
Dominic Robillard	06/11/2012

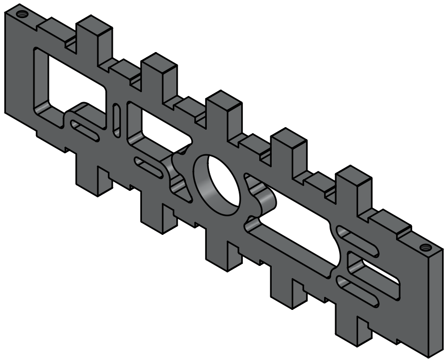
COMMENTS:
Design of a stair climbing inverted pendulum robot.
For a directed studies course in mechanical engineering.

Supervisor: Dr. Eric Lanteigne

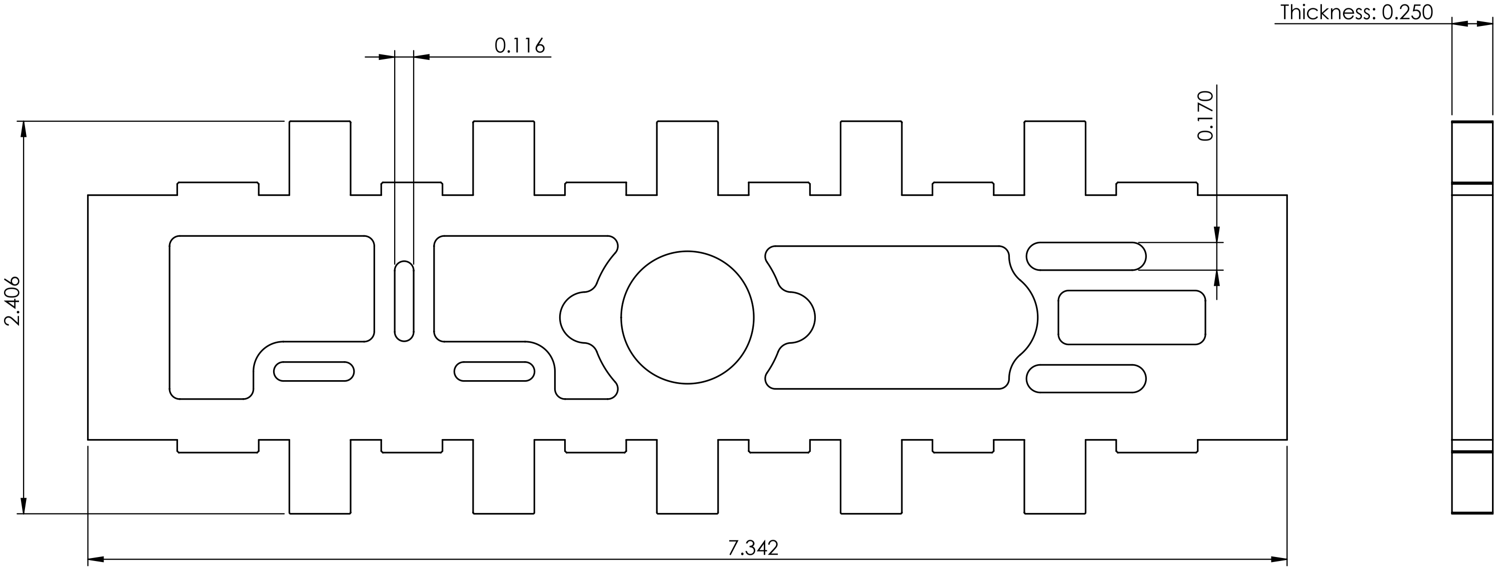
Filename: UO-SR-002 Wheel Hub Top Bracket

SIZE	PROJECT:	DWG. NO.	REV.
A	**Stair Robot**	**UO-SR-002**	**2**
SCALE: 1:1	WEIGHT: 37.25 g		SHEET 1 OF 2

NEED 4 COPIES



Rev. 1:
-Changed external dimensions
(before: 1.5 x 7.325in)
-Removed holes too small for laser
cutting



uOttawa

TOLERANCES

Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001

UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**
MATERIAL **6061 Alloy**
FINISH
PLOT SCALE: 1:1
THIRD ANGLE PROTECTION

DRAWN BY	DATE
Dominic Robillard	04/11/2012

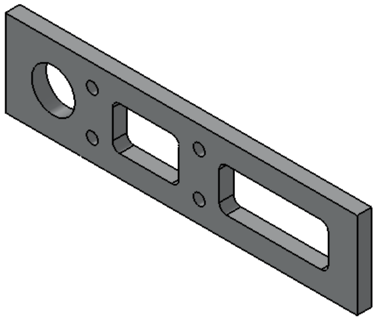
COMMENTS:
Design of a stair climbing
inverted pendulum robot.
For a directed studies course
in mechanical engineering.

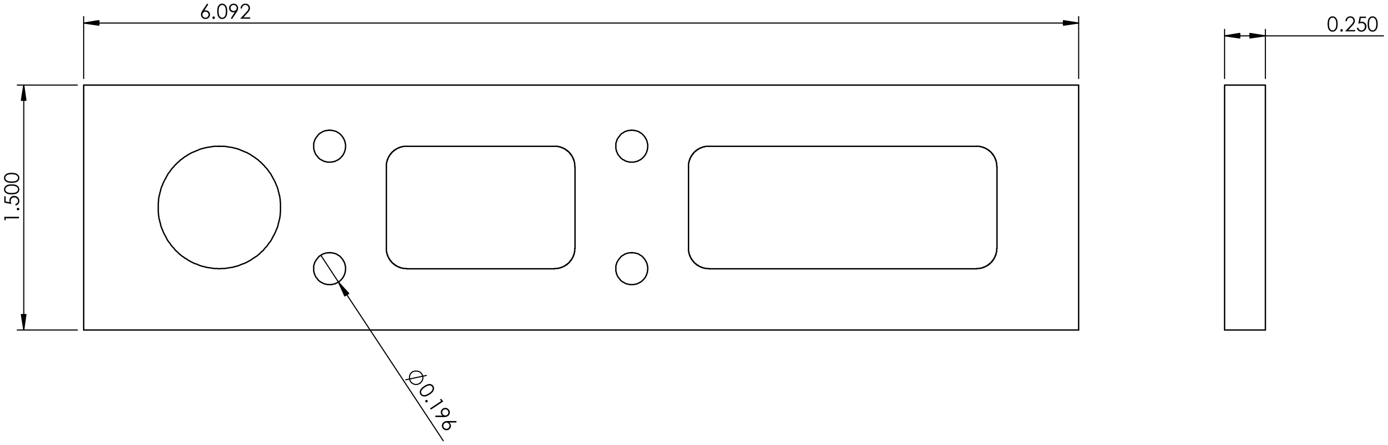
Supervisor: Dr. Eric Lanteigne

Filename: UO-SR_003 Vertical Beam Laser Cut

SIZE	PROJECT:	DWG. NO.	REV.
A	**Stair Robot**	**UO-SR-003**	**1**
SCALE: 1:1	WEIGHT: 96.05 g		SHEET 1 OF 1

NEED 2 COPIES





TOLERANCES

Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001

UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**
MATERIAL **6061 Alloy**
FINISH
PLOT SCALE: 1:1
THIRD ANGLE PROTECTION

DRAWN BY	DATE
Dominic Robillard	24/10/2012

COMMENTS:

Design of a stair climbing
inverted pendulum robot.
For a directed studies course
in mechanical engineering.

Supervisor: Dr. Eric Lanteigne

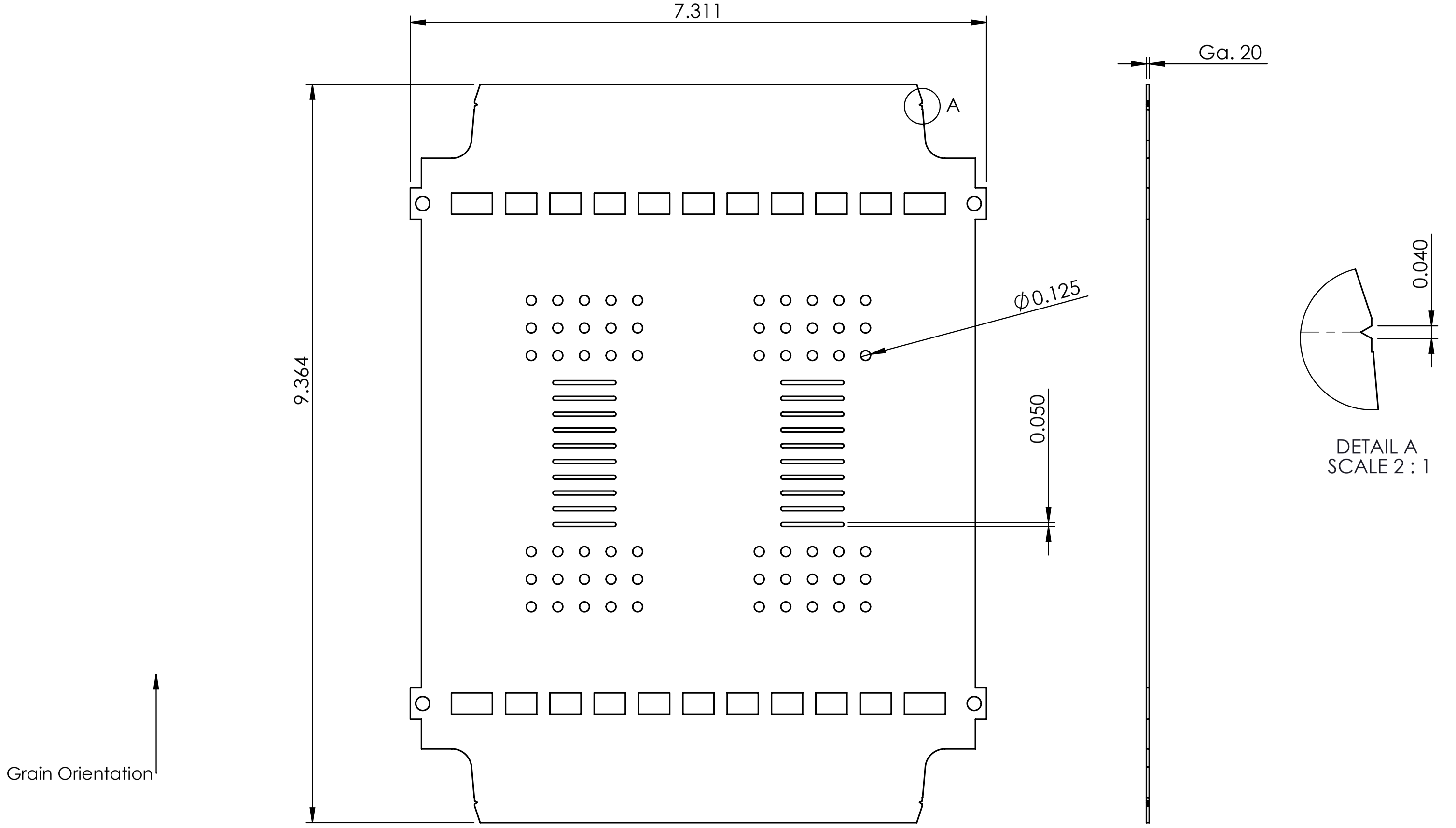
Filename:
UO-SR_004 Horizontal Beam Laser Cut - Copy

SIZE **A**
PROJECT: **Stair Robot**
DWG. NO. **UO-SR-004**
REV. **0**
SCALE: 1:1
WEIGHT: 68.12 g
SHEET 1 OF 1

Rev. 1: Added fold notches along edge,
Changed material to Alu 5052-H32

Rev. 2: Width slightly increased (from 7.294),
Changed panel holes design.

NEED 4 COPIES



uOttawa

TOLERANCES

Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001

UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**

MATERIAL **5052-H32**

FINISH

PLOT SCALE: 1:2
THIRD ANGLE PROTECTION

DRAWN BY	DATE
Dominic Robillard	04/11/2012

COMMENTS:

Design of a stair climbing inverted pendulum robot.
For a directed studies course in mechanical engineering.

Supervisor: Dr. Eric Lanteigne

Filename: UO-SR_005 Main Panel

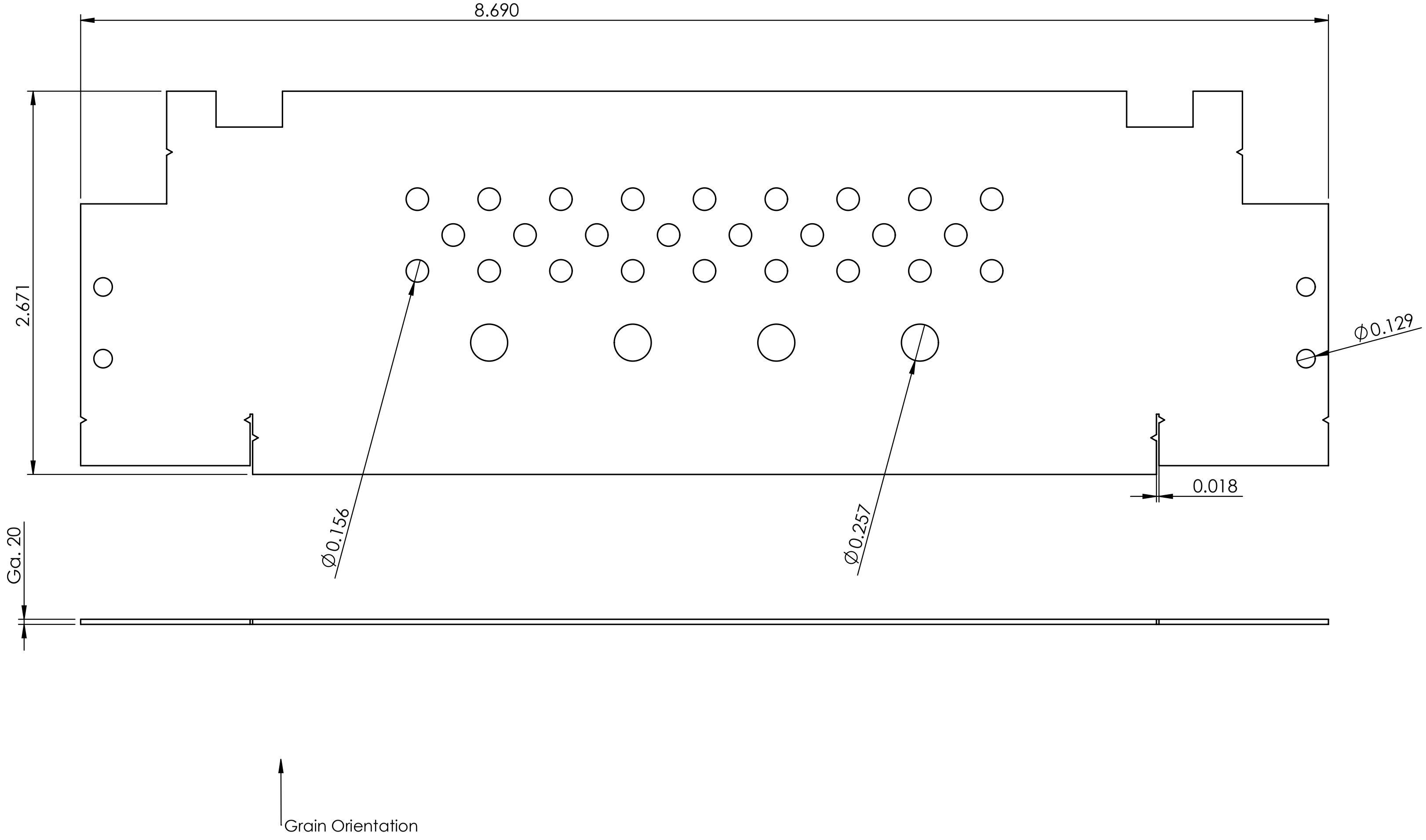
SIZE	PROJECT:	DWG. NO.	REV.
A	**Stair Robot**	**UO-SR-005**	**2**

SCALE: 1:2 WEIGHT: 92.60 g SHEET 1 OF 1

NEED 8 COPIES

Rev. 1: Added fold notches along edge,
Changed material to Alu 5052-H32

Rev. 2: Width slightly increased from 8.565in.



uOttawa

TOLERANCES

Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001

UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**
MATERIAL **5052-H32**
FINISH
PLOT SCALE: 1:1
THIRD ANGLE PROTECTION

DRAWN BY	DATE
Dominic Robillard	04/11/2012

COMMENTS:

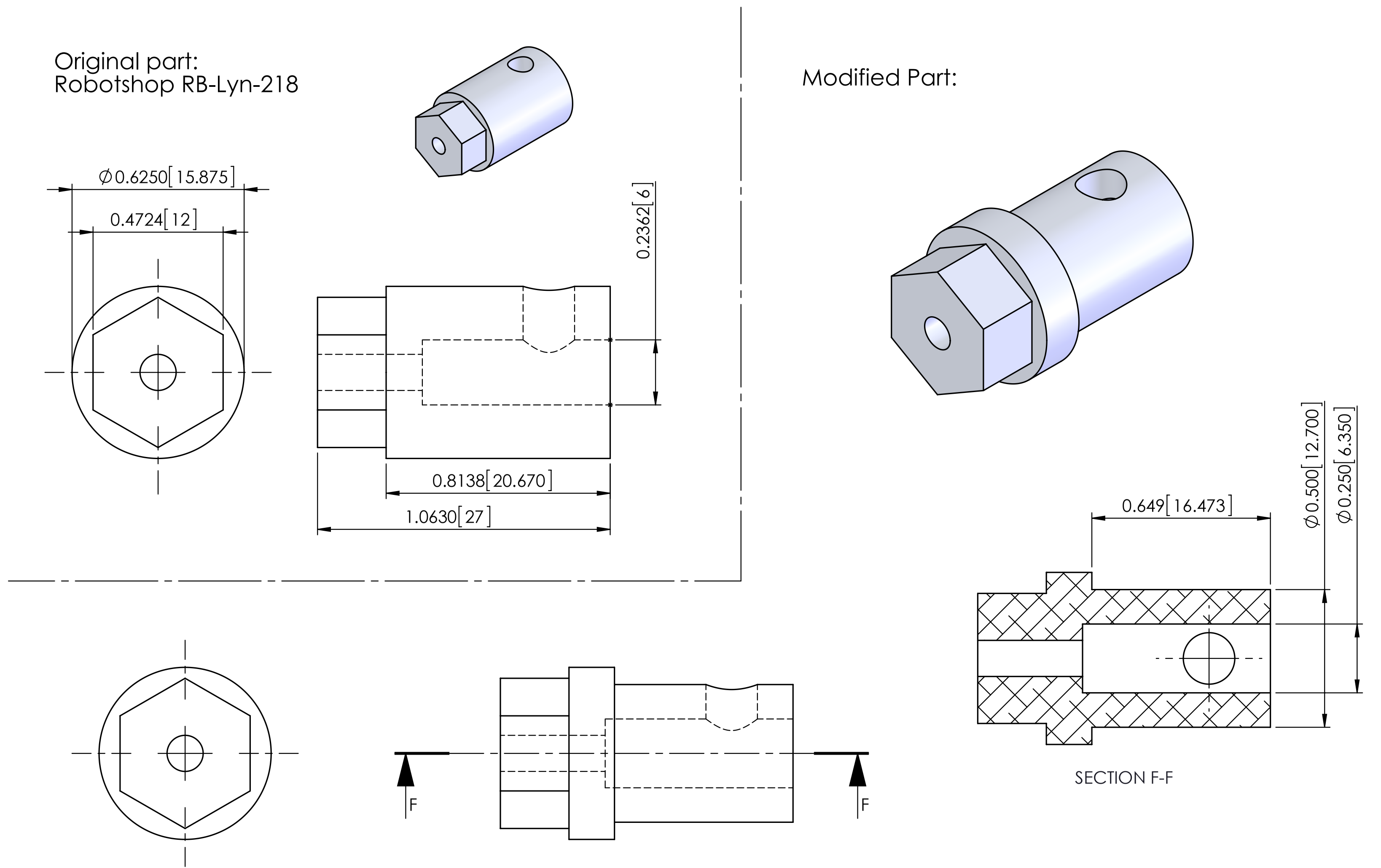
Design of a stair climbing
inverted pendulum robot.
For a directed studies course
in mechanical engineering.

Supervisor: Dr. Eric Lanteigne

Filename: UO-SR_006 Motor Panel

SIZE	PROJECT:	DWG. NO.	REV.
A	**Stair Robot**	**UO-SR-006**	2

SCALE: 1:1 WEIGHT: 32.86 g SHEET 1 OF 1



uOttawa

TOLERANCES (Inches)

Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001

UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**
MATERIAL
FINISH
THIRD ANGLE PROTECTION

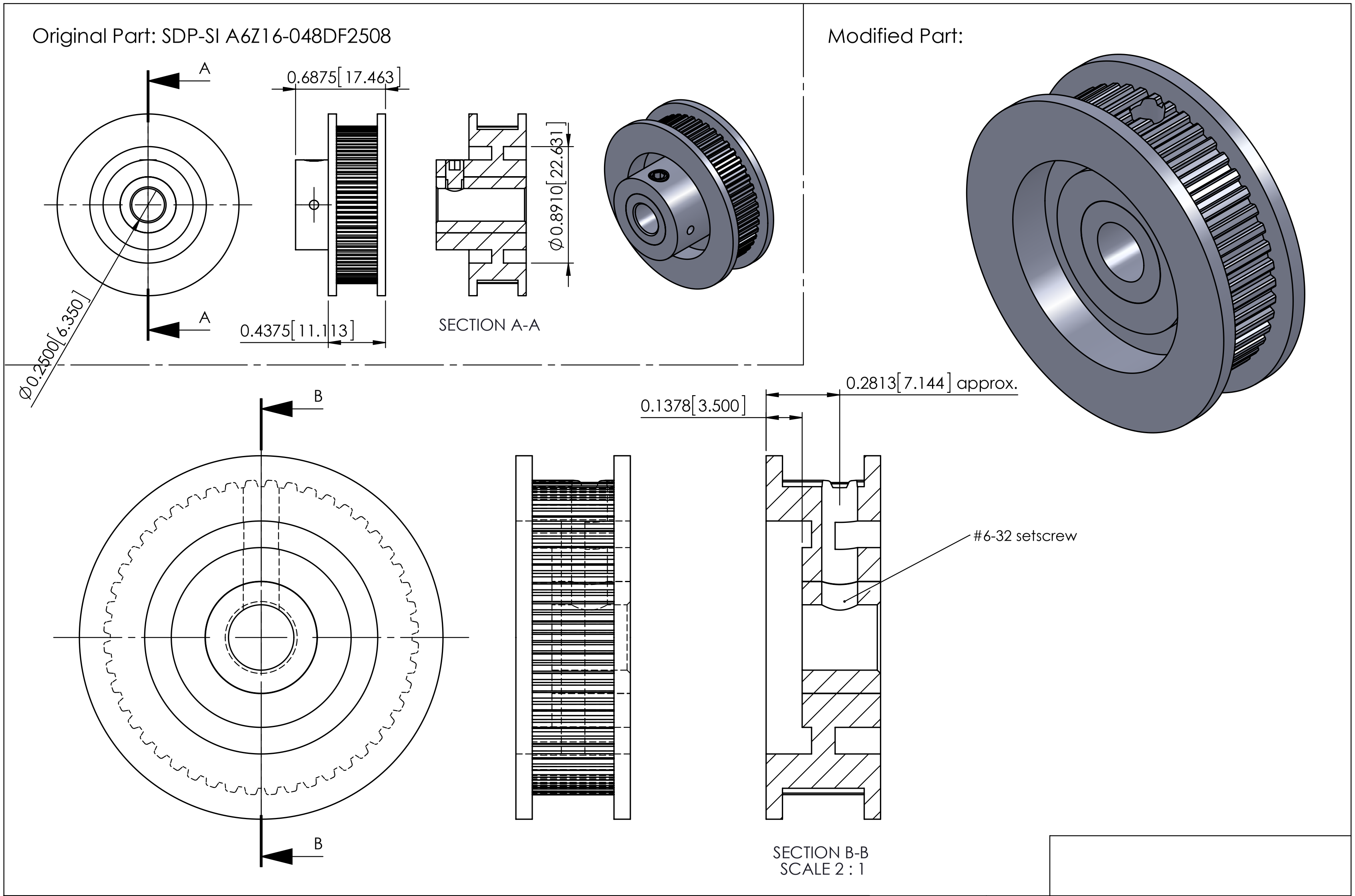
DRAWN BY Dominic Robillard
DATE 29/10/2012

COMMENTS:
Design of a stair climbing inverted pendulum robot. For a directed studies course in mechanical engineering.

Supervisor: Dr. Eric Lanteigne

Filename: UO-SR-007 12mm HEX adapter

SIZE **A**
PROJECT: **Stair Robot**
DWG. NO. **UO-SR-007**
REV. **0**
SCALE: 2:1
WEIGHT: g
SHEET 1 OF 1



uOttawa

TOLERANCES (Inches)

Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001
UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS
Inches [mm]
MATERIAL
FINISH
THIRD ANGLE PROTECTION

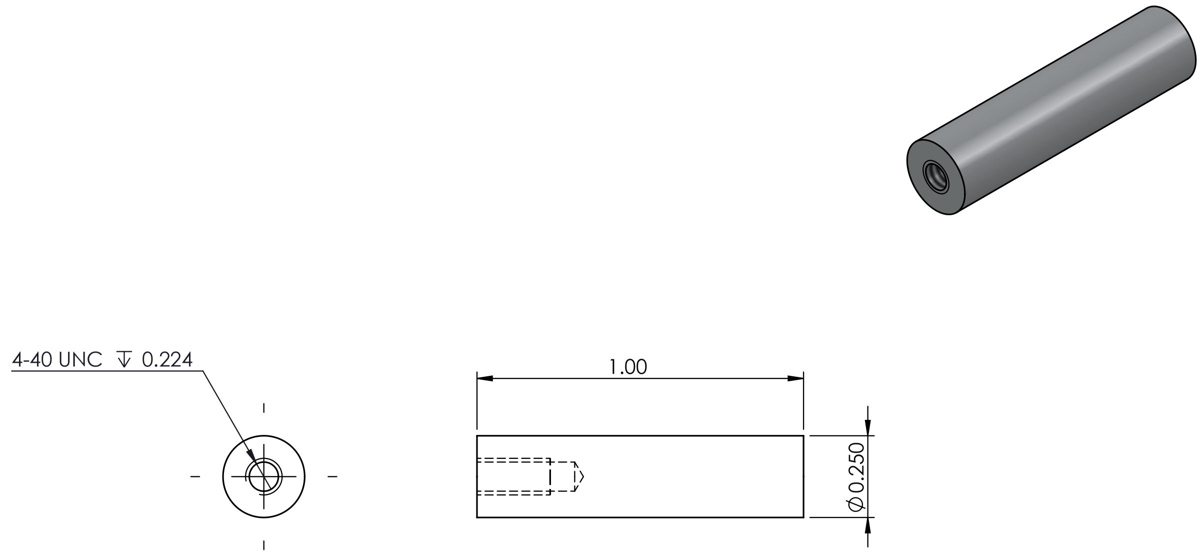
DRAWN BY	DATE
Dominic Robillard	29/10/2012

COMMENTS:
Design of a stair climbing inverted pendulum robot.
For a directed studies course in mechanical engineering.

Supervisor: Dr. Eric Lanteigne

Filename:
UO-SR-008 Modified drivetrain pulley

SIZE **A**
PROJECT: **Stair Robot**
DWG. NO. **UO-SR-008**
REV. **0**
SCALE: 1:1
WEIGHT: g
SHEET 1 OF 1



TOLERANCES (Inches)

Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001

UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**
MATERIAL **6061 Alloy**
FINISH
THIRD ANGLE PROTECTION

DRAWN BY	DATE
Dominic Robillard	29/10/2012

COMMENTS:

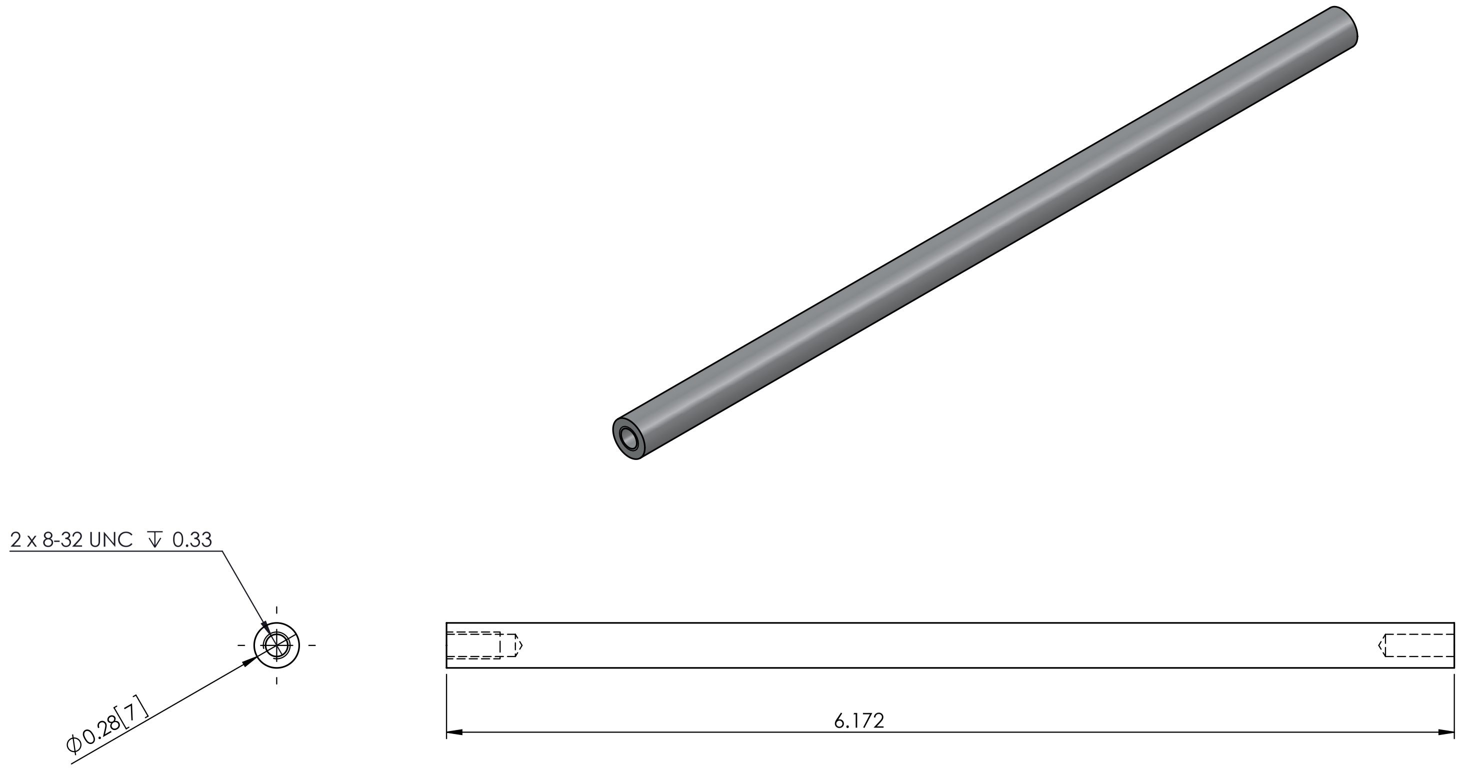
Design of a stair climbing
inverted pendulum robot.
For a directed studies course
in mechanical engineering.

Supervisor: Dr. Eric Lanteigne

Filename: UO-SR-009 Pulley Shaft

SIZE	PROJECT:	DWG. NO.	REV.
A	**Stair Robot**	**UO-SR-001**	**0**

SCALE: 2:1 | WEIGHT: 2.09 g | SHEET 1 OF 1



uOttawa

TOLERANCES (Inches)

Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001
UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**
MATERIAL **6061 Alloy**
FINISH
THIRD ANGLE PROTECTION

DRAWN BY	DATE
Dominic Robillard	29/10/2012

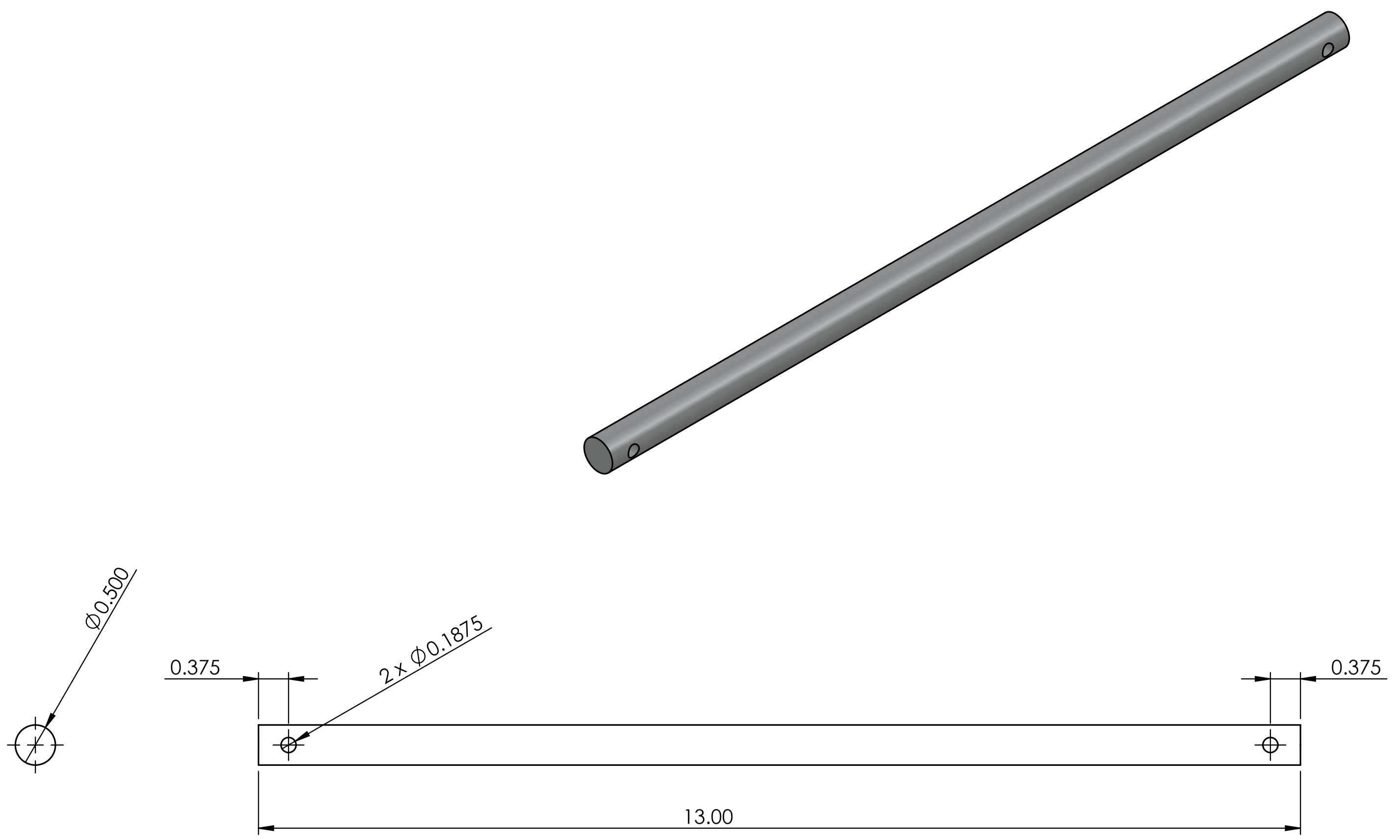
COMMENTS:
Design of a stair climbing inverted pendulum robot.
For a directed studies course in mechanical engineering.

Supervisor: Dr. Eric Lanteigne

Filename: UO-SR-010 Alignment Rod

SIZE	PROJECT:	DWG. NO.	REV.
A	**Stair Robot**	**UO-SR-010**	**0**

SCALE: 1:1 | WEIGHT: 15.73 g | SHEET 1 OF 1



uOttawa

TOLERANCES (Inches)

Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001

UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**
MATERIAL **6061 Alloy**
FINISH
THIRD ANGLE PROTECTION

DRAWN BY	DATE
Dominic Robillard	29/10/2012

COMMENTS:

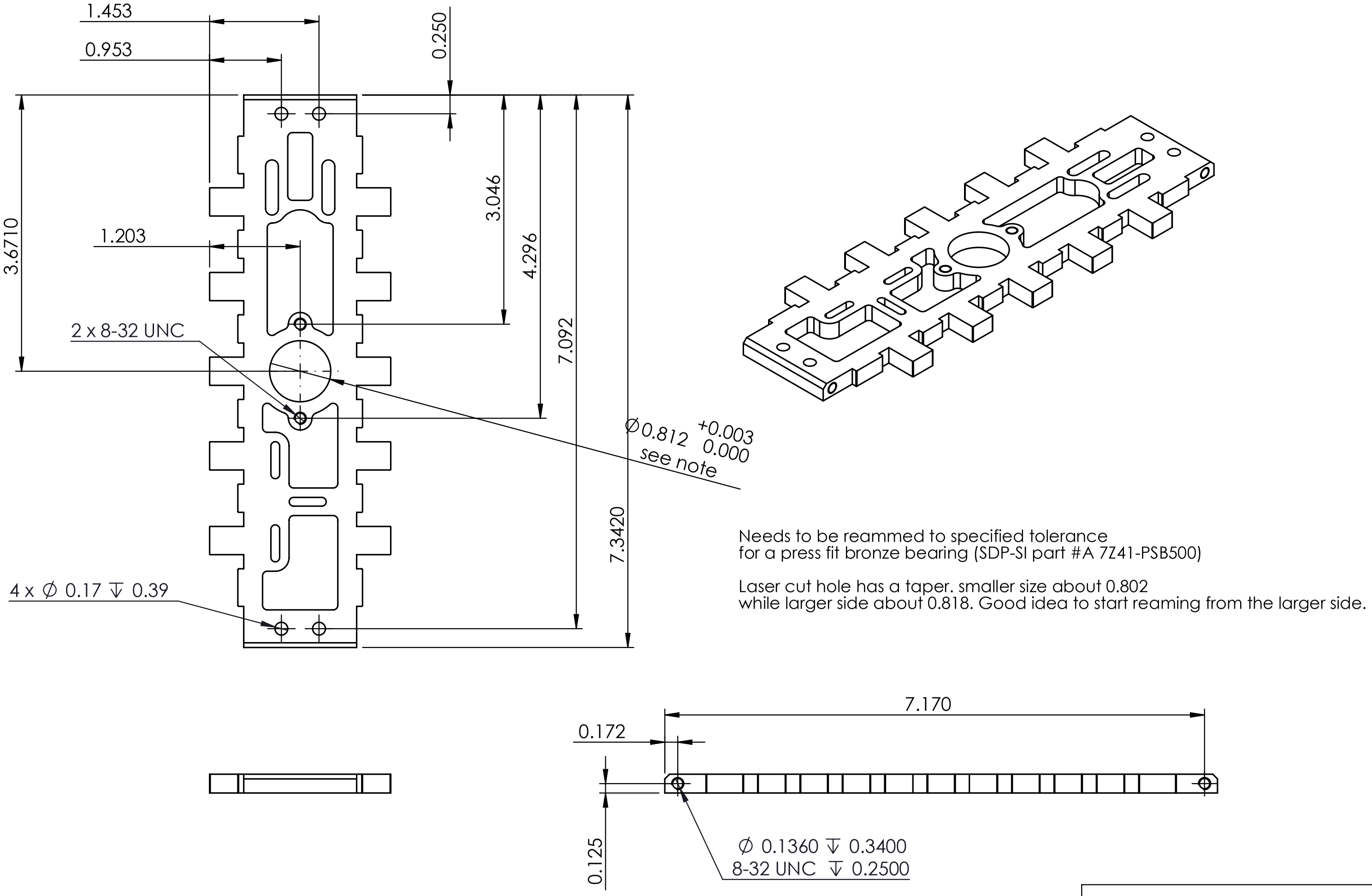
Design of a stair climbing inverted pendulum robot.
For a directed studies course in mechanical engineering.

Supervisor: Dr. Eric Lanteigne

Filename: UO-SR-011 Arm shaft

SIZE **A** PROJECT: **Stair Robot** DWG. NO. **UO-SR-011** REV. **0**
SCALE: 1:2 WEIGHT: 111.74 g SHEET 1 OF 1

Modified from UO-SR-003



Needs to be reammed to specified tolerance
for a press fit bronze bearing (SDP-SI part #A 7Z41-PSB500)

Laser cut hole has a taper. smaller size about 0.802
while larger side about 0.818. Good idea to start reaming from the larger side.

uOttawa

TOLERANCES (Inches)

Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001
UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

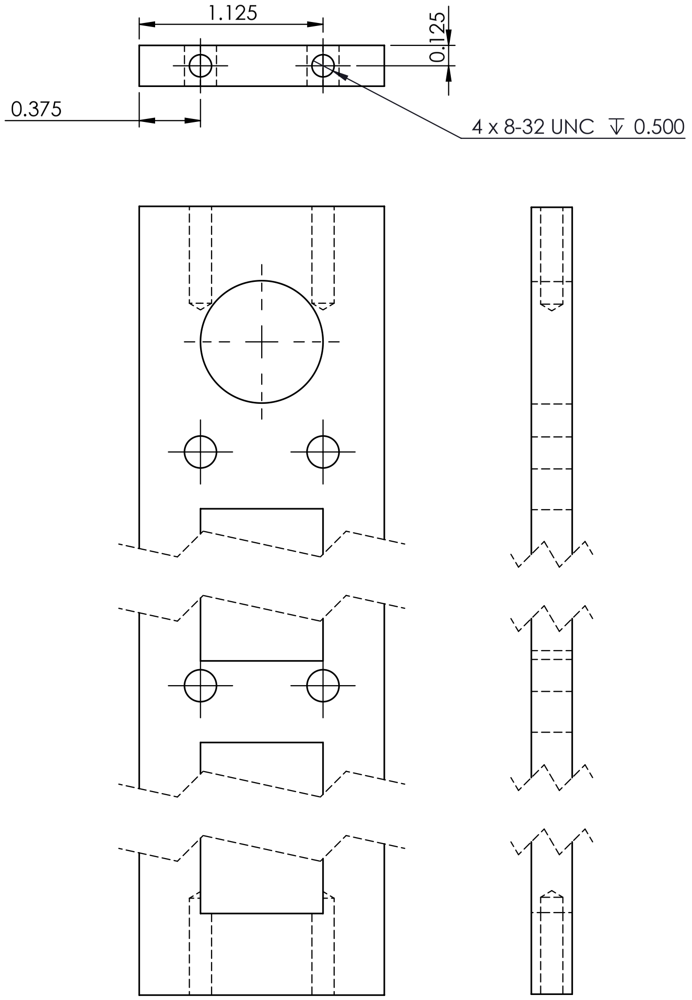
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS: **Inches [mm]**
MATERIAL: **6061 Alloy**
FINISH:
THIRD ANGLE PROTECTION

DRAWN BY: Dominic Robillard
DATE: 24/05/2014
COMMENTS:
Design of a stair climbing inverted pendulum robot.
For a directed studies course in mechanical engineering.
Supervisor: Dr. Eric Lanteigne

Filename:
UO-SR-012 Vertical Beam modification

SIZE: **A**
PROJECT: **Stair Robot**
DWG. NO. **UO-SR-012**
REV. **0**
SCALE: 1:2
WEIGHT: 96.05 g
SHEET 1 OF 1



Modified from UO-SR-004

uOttawa

TOLERANCES (Inches)

Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001
UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**
MATERIAL **6061 Alloy**
FINISH
THIRD ANGLE PROTECTION

DRAWN BY	DATE
Dominic Robillard	24/05/2014

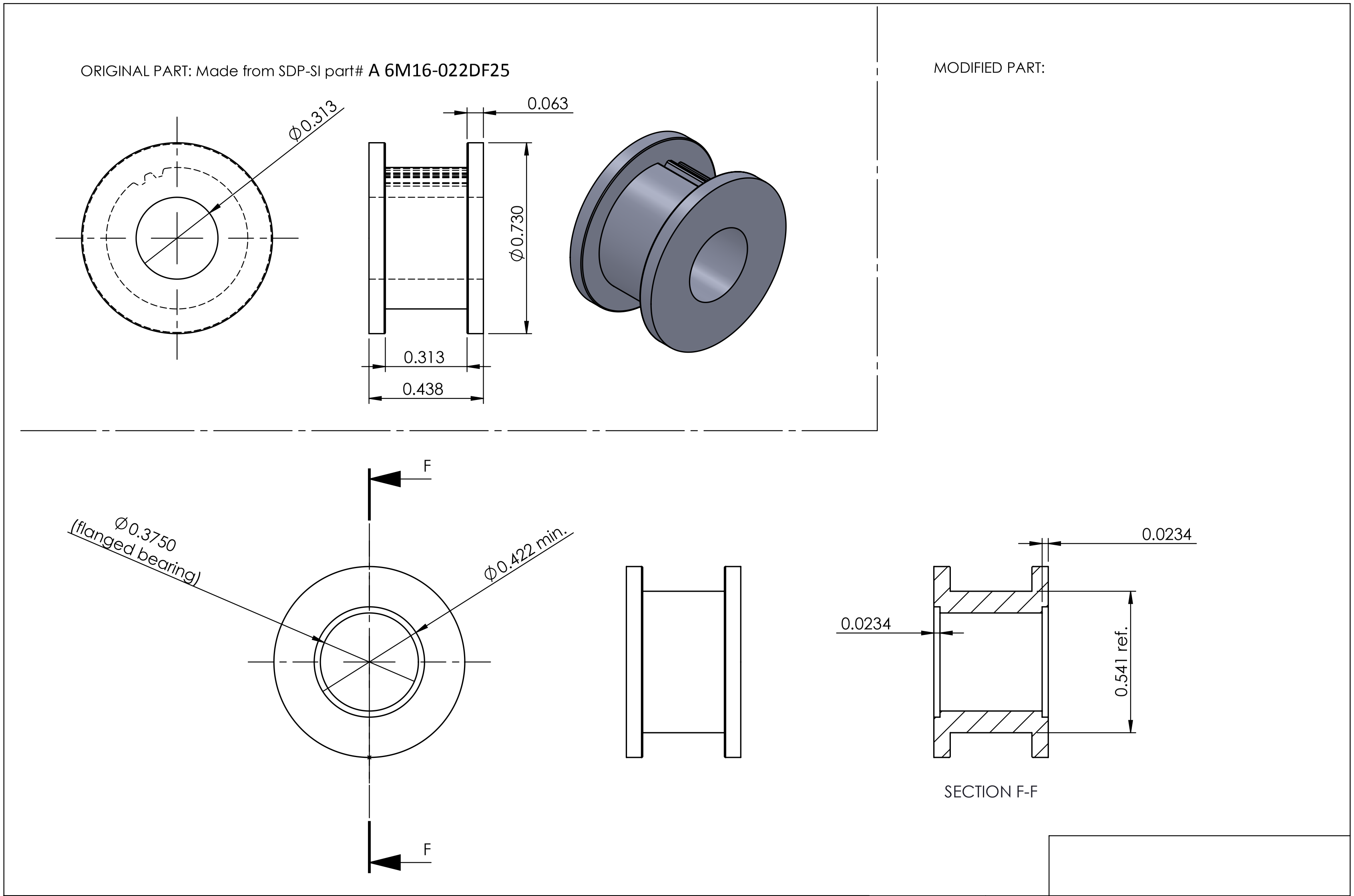
COMMENTS:

Design of a stair climbing inverted pendulum robot. For a directed studies course in mechanical engineering.

Supervisor: Dr. Eric Lanteigne

Filename: UO-SR-013 Arm motor mount modification

SIZE **A**
PROJECT: **Stair Robot**
DWG. NO. **UO-SR-013**
REV. **0**
SCALE: 1:1
WEIGHT: 66.49 g
SHEET 1 OF 1



uOttawa

TOLERANCES (Inches)

Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001
UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**
MATERIAL
FINISH
THIRD ANGLE PROTECTION

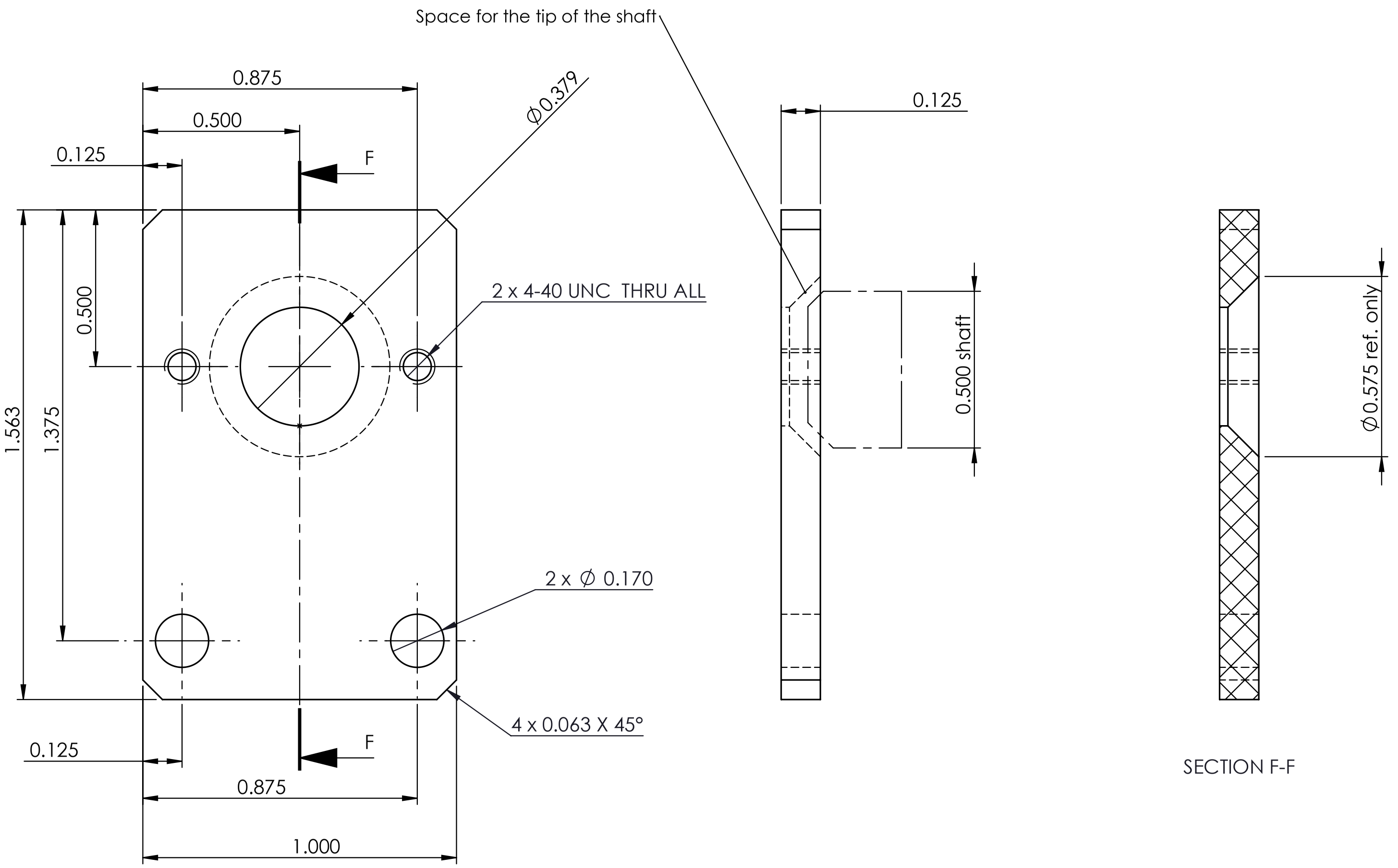
DRAWN BY	DATE
Dominic Robillard	24/05/2014

COMMENTS:
Design of a stair climbing inverted pendulum robot.
For a directed studies course in mechanical engineering.

Supervisor: Dr. Eric Lanteigne

Filename: UO-SR-014 Idler Toothed pulley modification

SIZE **A**
PROJECT: **Stair Robot**
DWG. NO. **UO-SR-014**
REV. **0**
SCALE: 2:1
WEIGHT: g
SHEET 1 OF 1



uOttawa

TOLERANCES (Inches)

Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001
UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**
MATERIAL **ABS**
FINISH
THIRD ANGLE PROTECTION

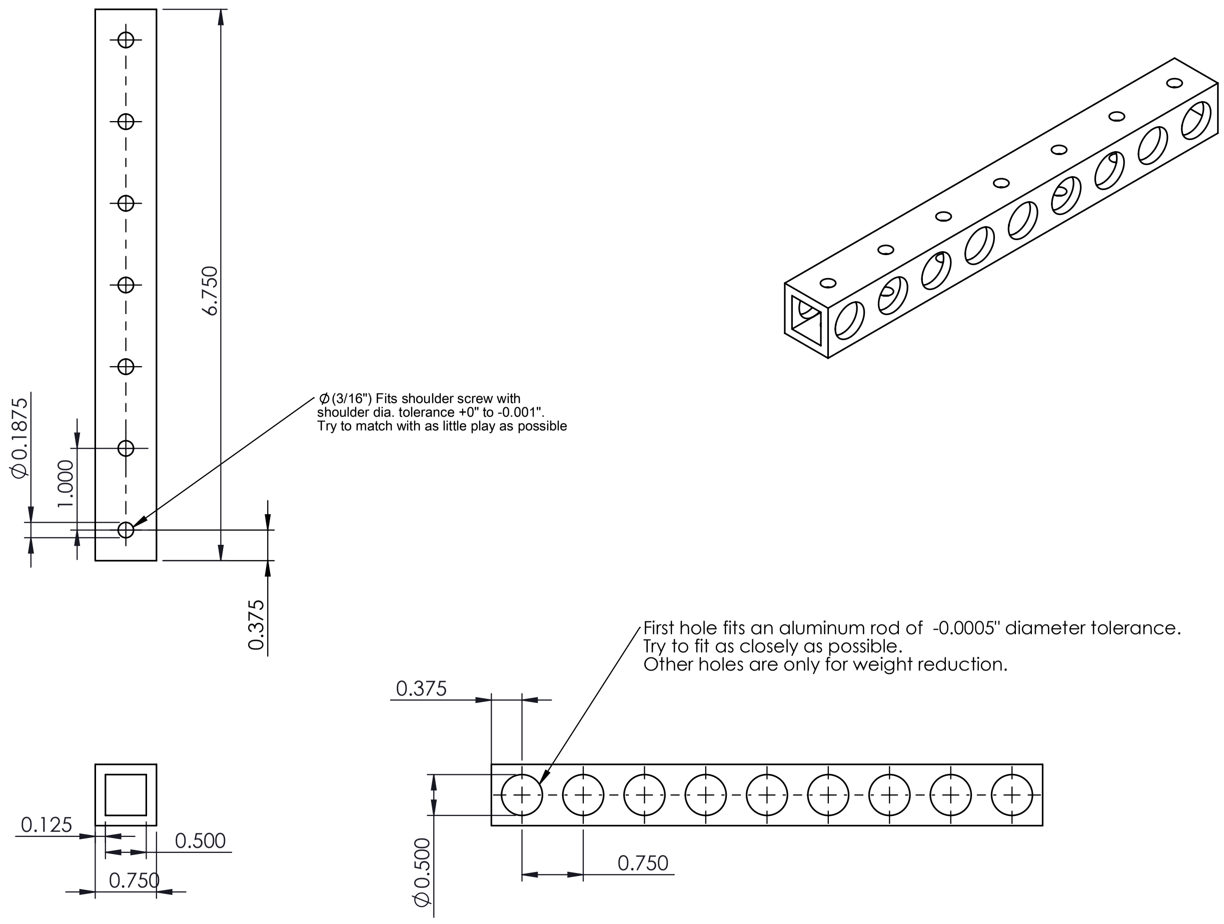
DRAWN BY	DATE
Dominic Robillard	24/05/2014

COMMENTS:
Design of a stair climbing
inverted pendulum robot.
For a directed studies course
in mechanical engineering.

Supervisor: Dr. Eric Lanteigne

Filename: UO-SR-015 Encoder Plate

SIZE **A**
PROJECT: **Stair Robot**
DWG. NO. **UO-SR-000**
REV. **0**
SCALE: 2:1
WEIGHT: 2.78 g
SHEET 1 OF 1



Ø(3/16") Fits shoulder screw with shoulder dia. tolerance +0" to -0.001". Try to match with as little play as possible

First hole fits an aluminum rod of -0.0005" diameter tolerance.
Try to fit as closely as possible.
Other holes are only for weight reduction.

uOttawa

TOLERANCES (Inches)

Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001
UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

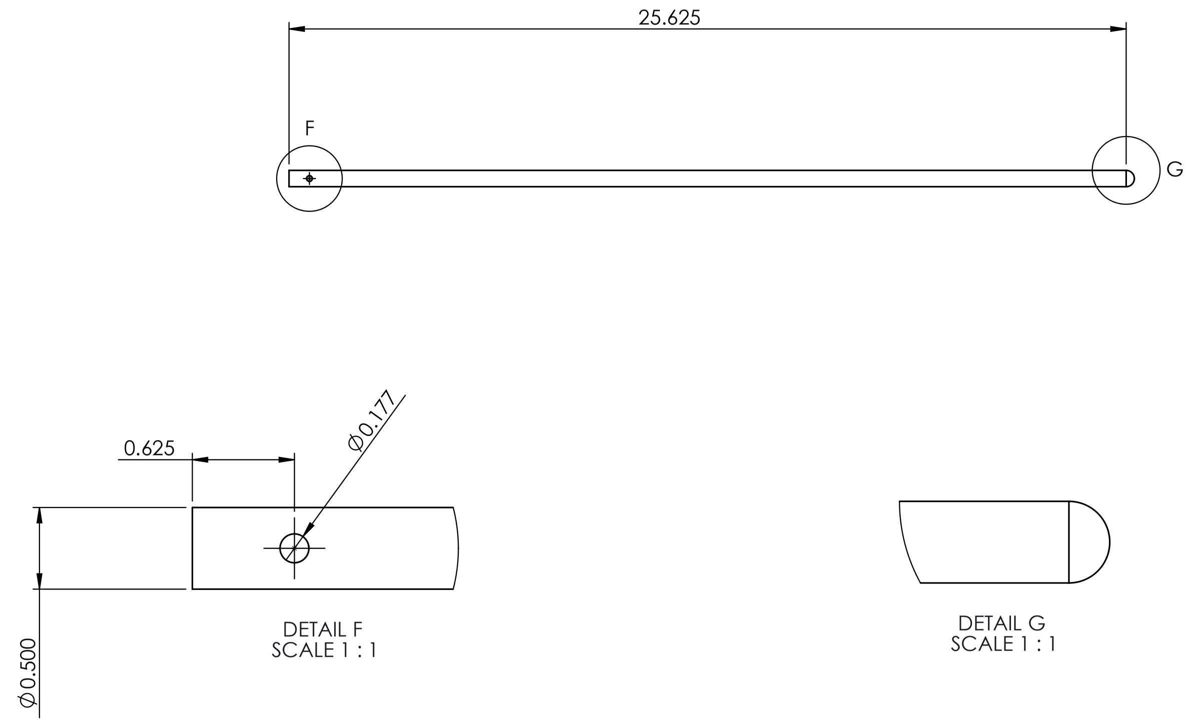
UNITS **Inches [mm]**
MATERIAL **6063-T5**
FINISH
THIRD ANGLE PROTECTION

DRAWN BY Dominic Robillard
DATE 24/05/2014
COMMENTS:
Design of a stair climbing inverted pendulum robot.
For a directed studies course in mechanical engineering.

Supervisor: Dr. Eric Lanteigne

Filename: UO-SR-016 Square Tube Pieces

SIZE **A**
PROJECT: **Stair Robot**
DWG. NO. **UO-SR-016**
REV. **0**
SCALE: 1:2
WEIGHT: 71.64 g
SHEET 1 OF 1



uOttawa

TOLERANCES (Inches)

Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001
UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**
MATERIAL
FINISH
THIRD ANGLE PROTECTION

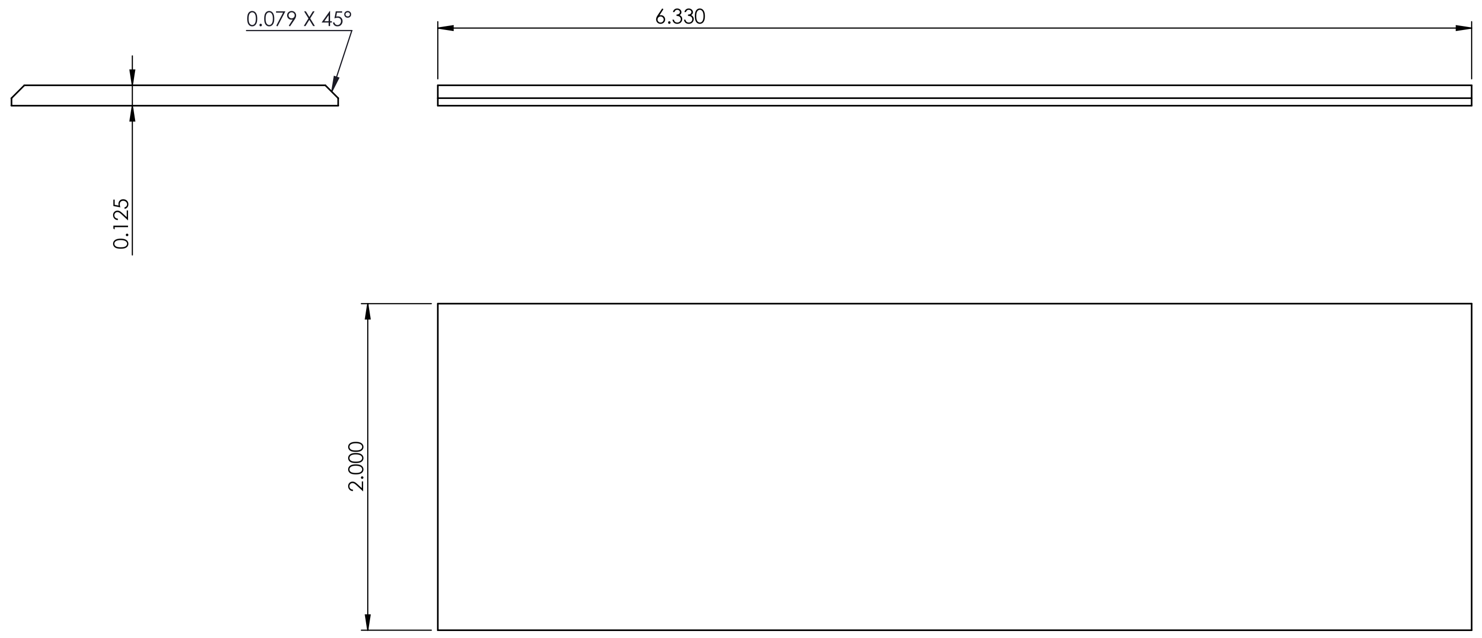
DRAWN BY	DATE
Dominic Robillard	24/05/2014

COMMENTS:
Design of a stair climbing inverted pendulum robot.
For a directed studies course in mechanical engineering.

Supervisor: Dr. Eric Lanteigne

Filename: UO-SR-017 Acetal Arm Rod

SIZE **A**
PROJECT: **Stair Robot**
DWG. NO. **UO-SR-017**
REV. **0**
SCALE: 1:5
WEIGHT: g
SHEET 1 OF 1



uOttawa

TOLERANCES (Inches)

Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001
UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**
MATERIAL **ABS**
FINISH
THIRD ANGLE PROTECTION

DRAWN BY	DATE
Dominic Robillard	24/05/2014

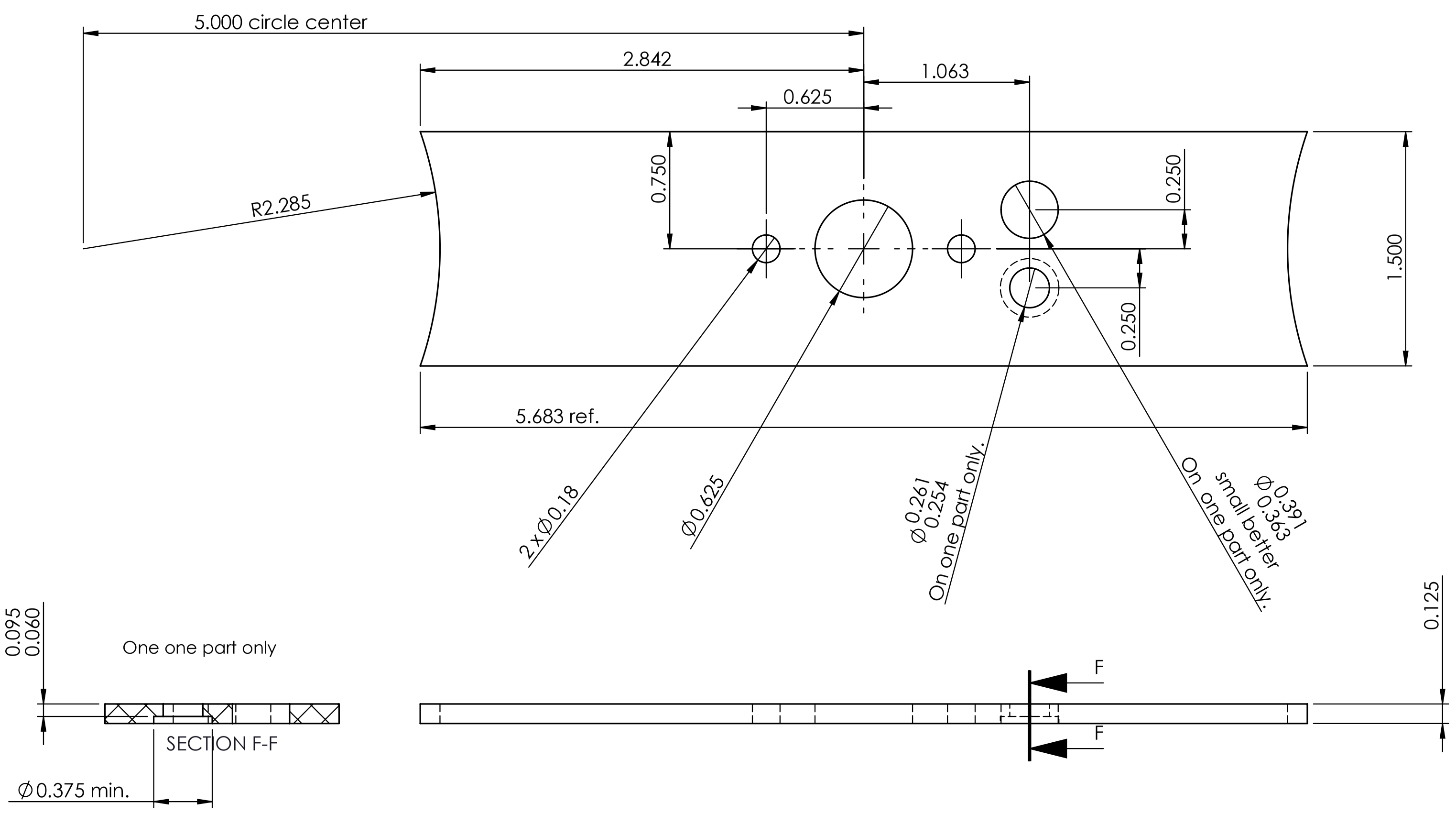
COMMENTS:
Design of a stair climbing inverted pendulum robot.
For a directed studies course in mechanical engineering.

Supervisor: Dr. Eric Lanteigne

Filename: UO-SR-018 Bottom ABS panel

SIZE **A**
PROJECT: **Stair Robot**
DWG. NO. **UO-SR-018**
REV. **0**
SCALE: 1:1
WEIGHT: 25.79 g
SHEET 1 OF 1

MAKE 2 COPIES. One with the side holes and one without



uOttawa

TOLERANCES (Inches)

Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001
UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**
MATERIAL **ABS**
FINISH
THIRD ANGLE PROTECTION

DRAWN BY: Dominic Robillard
DATE: 24/05/2014

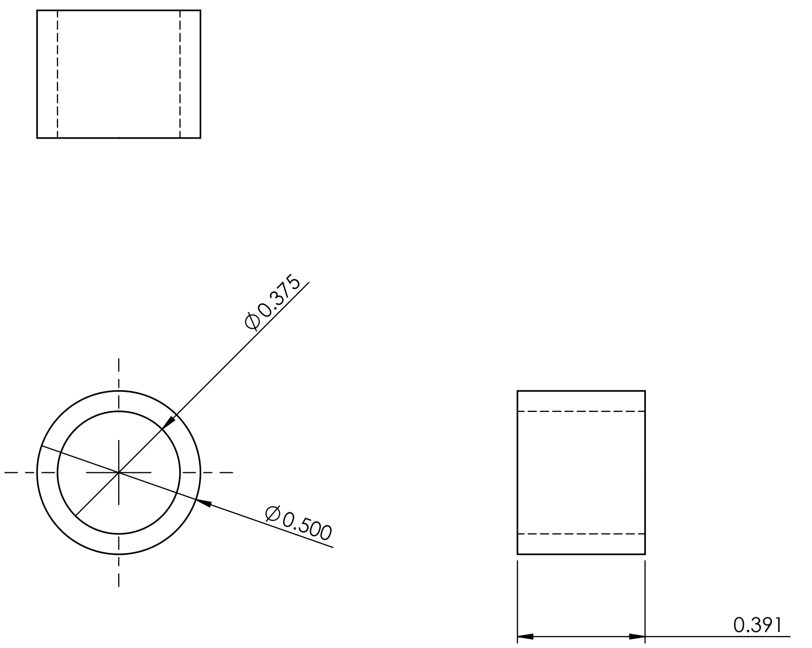
COMMENTS:
Design of a stair climbing inverted pendulum robot.
For a directed studies course in mechanical engineering.

Supervisor: Dr. Eric Lanteigne

Filename: UO-SR-019 Side ABS panel

SIZE **A**
PROJECT: **Stair Robot**
DWG. NO. **UO-SR-019**
REV. **0**
SCALE: 1:1
WEIGHT: 16.16 g
SHEET 1 OF 1

Made from McMaster Nylon tube 8628K28



uOttawa

TOLERANCES (Inches)

Fractional: ±1/64
Angular: ±1°
Two Place Decimal: ± 0.010
Three Place Decimal: ± 0.005
Four Place Decimal: ± 0.001
UNLESS OTHERWISE SPECIFIED

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**
MATERIAL
FINISH
THIRD ANGLE PROTECTION

DRAWN BY	DATE
Dominic Robillard	24/05/2014

COMMENTS:
Design of a stair climbing
inverted pendulum robot.
For a directed studies course
in mechanical engineering.

Supervisor: Dr. Eric Lanteigne

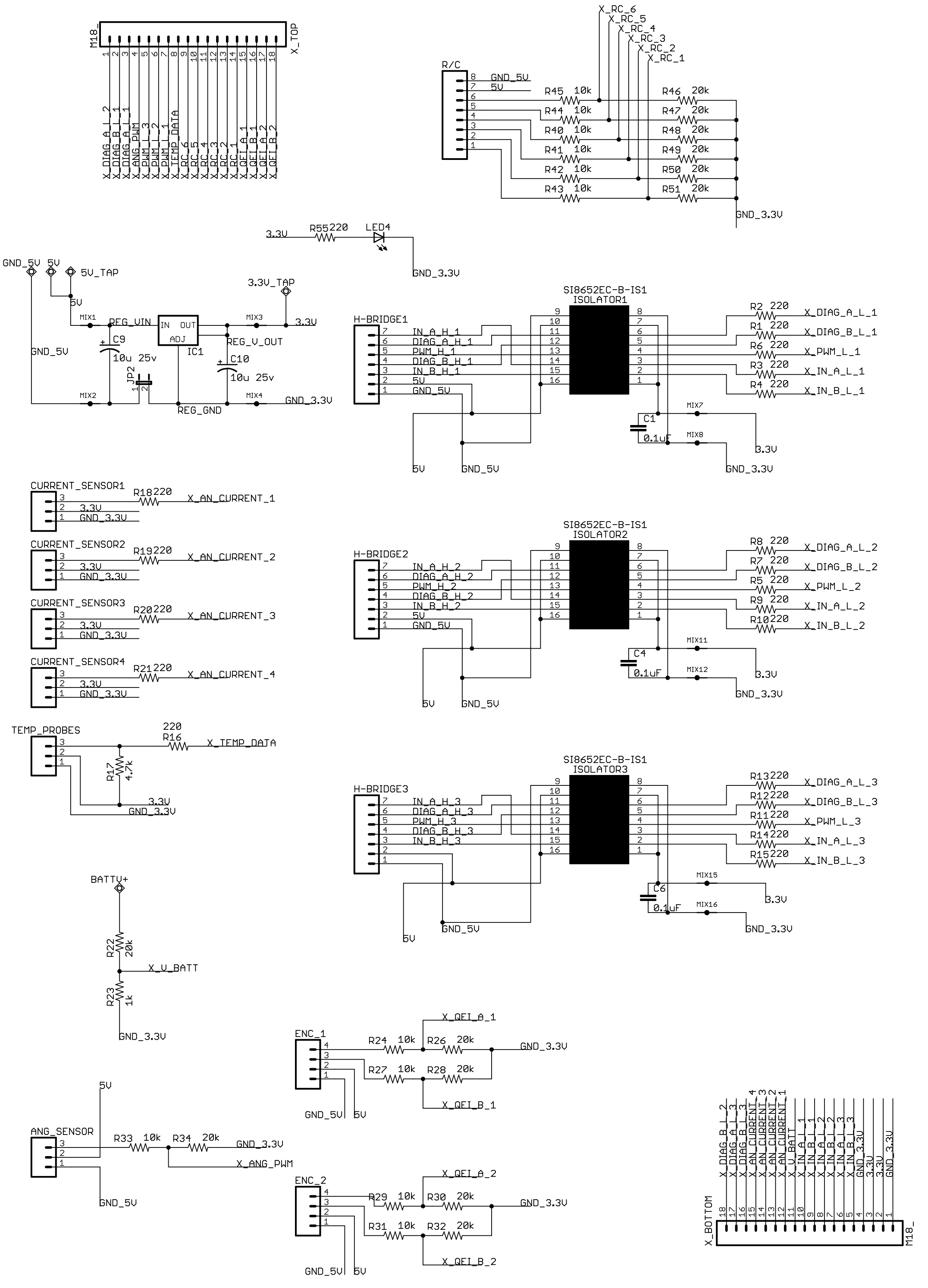
Filename: UO-SR-020 Idler Flat pulley

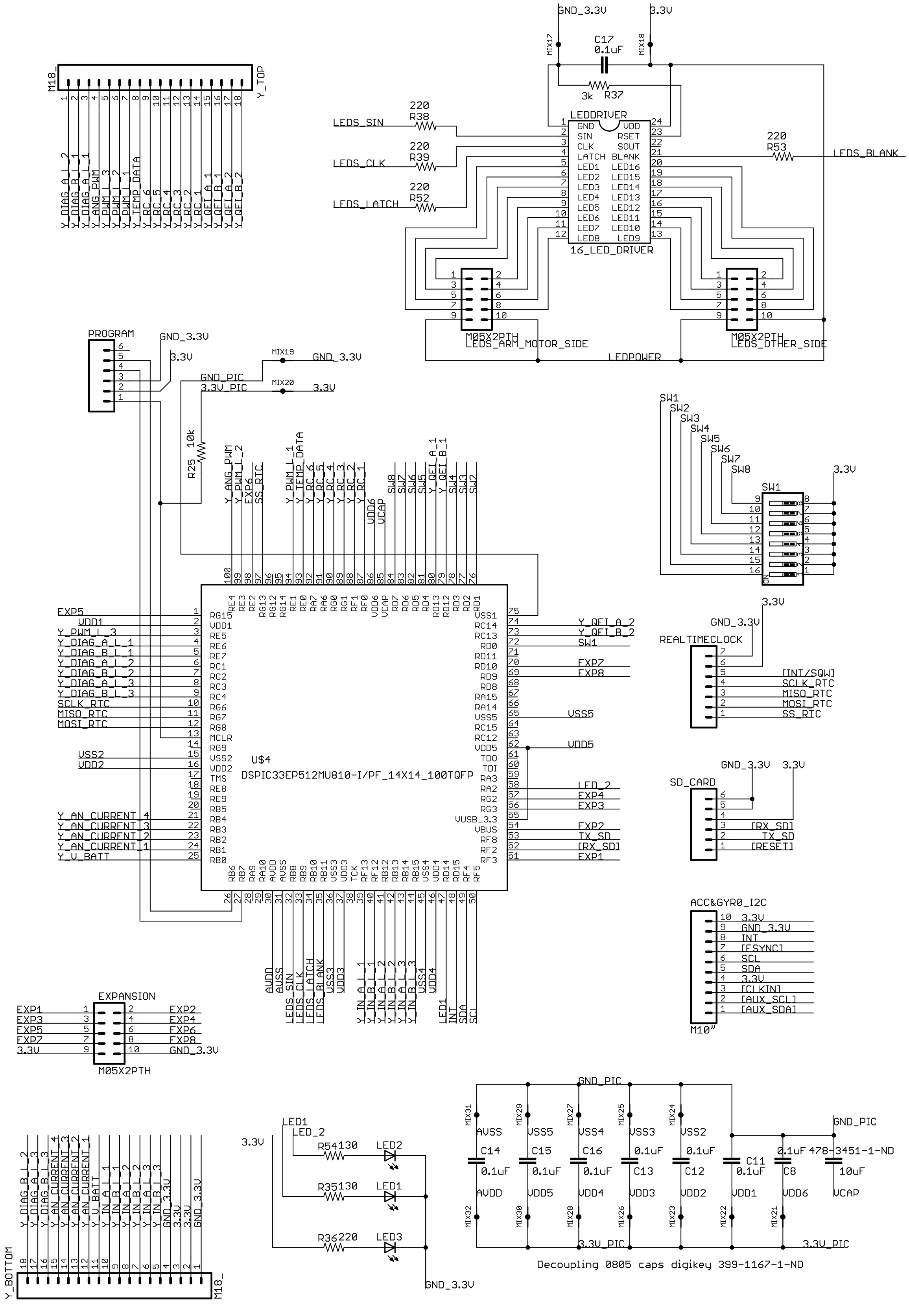
SIZE	PROJECT:	DWG. NO.	REV.
A	**Stair Robot**	**UO-SR-020**	**0**

SCALE: 2:1
WEIGHT: g
SHEET 1 OF 1

Appendix F

Circuit schematic of the stair-climbing self-balancing robot





Decoupling 0805 caps digikey 399-1167-1-ND

Appendix G

Publication on the stair-climbing self-balancing robot

DEVELOPMENT OF COMPACT SELF-BALANCING ROBOT WITH SELF-STANDING AND STAIR CLIMBING CAPABILITY

Dominic Robillard, Eric Lanteigne

ABSTRACT

Self-balancing robots can turn on the spot using differential steering with better efficiency than tracked or four-wheel drive robots, and they can be many times their width in height. However they have two major limitations: they cannot stand-up on their own and cannot climb stairs. In this paper, a novel design is proposed to address these issues. A single degree of freedom is added to the center of a four-wheel drive robot. This arm allows the robot to climb stairs and stand-up on its own. A model and simulation of the balancing and the stair-climbing process are derived and the stair-climbing is compared against experimental results with a prototype. It was shown that the model closely follows the trend of the experimental results and provides a basis for future studies on the concept.

Keywords: robot; stair; climbing; balancing; standing.

DÉVELOPPEMENT D’UN ROBOT COMPACT S’AUTO-BALANÇANT AVEC HABILETÉ DE S’AUTO ÉRIGER ET DE MONTER DES ESCALIERS

RÉSUMÉ

Les robots qui s’auto-balançent peuvent tourner sur place en utilisant la traction différentielle avec une meilleure efficacité que les robots à chenilles ou à quatre roues. Ils peuvent être plusieurs fois leur largeur en hauteur. Cependant, ils ont deux limitations majeures : ils ne peuvent pas se lever eux-mêmes pour se balancer et ne peuvent pas monter des escaliers. Dans cet article, une nouvelle méthode est démontrée pour résoudre ces deux limitations : ajouter un bras avec un seul degré de liberté au centre d’un robot à quatre roues motrices. Ce bras permet au robot de monter des escaliers et de se lever soi-même. Un modèle et une simulation du balancement puis un modèle de la séquence pour monter une marche est développé. Le second modèle est comparé avec des résultats expérimentaux sur un prototype. Il a été montré que le modèle suit de près la tendance des résultats expérimentaux et fournit une base pour de futures études sur le concept.

Mots-clés : robot ; escaliers ; grimpe ; balance ; debout.

1. INTRODUCTION

Self-balancing robots are a special type of robotic platform that have a number of advantages. They have simple mechanical designs, where they do not require a steering mechanism which also makes them energy efficient. They can also turn on the spot and they have the ability to be taller without requiring a large base, which is closer to the human morphology. Those key advantages would make them perfect as surveillance robots if it was not for their two major limitations: they cannot stand up on their own and they cannot climb stairs. The objective of this research is to develop a simple and effective method to solve these two limitations. On the other hand, stair climbing robots tend to be bulky, expensive, and large which limits their ability to navigate door openings. They are often tracked drive robots which make them inefficient on flat ground. The robot proposed in this article is not only solving the two major limitations of self-balancing robots, but also proving that a very simple and compact design for stair-climbing is possible.

The most common configuration of stair-climbing robot is the tracked drive. Providing the threads are sufficiently long to overlap at least two steps at a time, the robot sees the staircase as a slope and can easily climb stairs. However the difficulty is in transitioning from a flat surface onto the first step. For this, several methods have been proposed. An example is the four-tracked robot that can move its front and rear tracks relative to each other to effectively climb the first step. Once the robot is on the first step, the robot can re-align its tracks and climb the remaining steps [1]. Another method is to use a deformable/reconfigurable track such as the LMA robot shown in [2]. Tracked robots have good outdoor applications but are not ideal for indoor robots because they consume a lot of energy, are slow, and are rather large.

The Whegs robots developed by the Case Western University [3] shows a good example of a biologically-inspired robot that is small, compact, and very good at climbing obstacles such as steps. These robots may however have trouble climbing a flight of stairs (more than one step) and because they do not have wheels, are less efficient on flat ground.

Self-balancing robots are in the category of differential drive robots. They have two wheels, but instead of a caster wheel they use a dynamic control loop to keep their body upright. Perhaps the best example of a commercially successful self-balancing platform is the Segway vehicle [4]. It is popular with tourists where they can rent them to visit local attractions.

The concept of self-balancing robots is not new [5], but the recent popularity of the Segway, the availability of low cost gyroscopic sensors, and powerful low-cost microcontrollers have inspired many versions of self-balancing robots both from research environments and amateurs. Good examples are given in [6–11].

Little work has been done on giving a self-balancing robot the ability to stand-up on its own. There is a lot of research done in the stand-up routine of humanoid robots [12] but they are not applicable since these platforms have legs with many degrees of freedom. Self-erecting an inverted pendulum has been examined, but the methods used (swing the pendulum back and forth from the hanging position until enough energy is provided to raise high) cannot be applied to self-balancing robots. One method is shown in [13] where an internal flywheel is forced to turn away from its spinning axis. The resulting precession force generated tilts the robot upwards until it is standing. The concept however requires at least two actuators (one to spin the disk, and the other to push the flywheel).

2. ROBOT OVERVIEW

The proposed solution is to have a four-wheel drive differential steering mobile platform. The left and right sides of the robot each have their own drive motors and both wheels on the same side are mechanically coupled. This platform provides good mobility on flat ground and is suitable for a self-balancing robot because the opposite wheel provides protection when the robot falls from balancing.

This platform is augmented with an arm attached at the middle of the body. This arm is driven by a powerful actuator and can indefinitely rotate around its joint. It is about twice as long as the body. The body

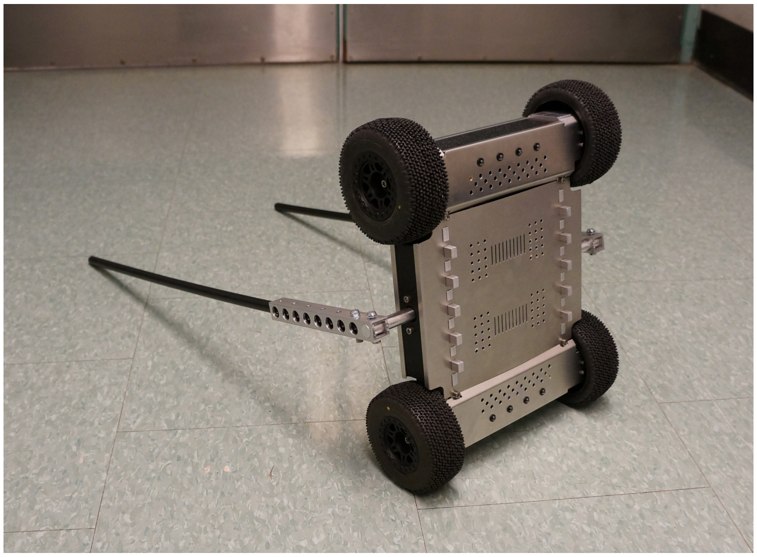


Fig. 1. Photograph of the robot developed for climbing stairs and self-balancing. Including its wheels, the standing robot is 14 inches high. The arm is 26 inches long.

carries all the heavy components while the arm is made as light as possible to keep the center of gravity of the robot predictable. Figure 1 shows a photograph of the custom made robot.

To stand-up from a resting position (four wheels in contact with floor), the arm rotates and lifts the body of the robot until close to equilibrium. At which point the balancing can start.

To climb stairs the robot aligns itself in four-wheel mode with the first step and the arm starts to rotate against the floor on the opposite side from the stairs. Once the arm touches the floor, the whole robot's body starts to turn. After ¼ turn, the body falls on the nose of the first step while the arm keeps rotating.

This process goes on for every step while the whole body of the robot continuously rotates. This is a simple effective method. It does have the drawback of making it more difficult to use sensors and cameras while the robot is climbing stairs. However because of the long arm towards the bottom of the stairs the climbing sequence is reliable. If the robot misses a step, it is almost impossible for the robot to tumble all the way down. The robot does not rely on the balancing ability while climbing steps.

In terms of the robot external dimensions, there are some guidelines that had to be followed. The wheel radius should not be more than the thread of a step and it should be at least large enough to grab the nose of a step with a vertical reaction force. This includes steps that have a fillet or chamfer to them. The dimension parameters such as the robot's body length, the robot's body width, and the rise of a step are all related. Essentially, the body of the robot must be kept smaller than the wheel radius to give a chance to properly grab the nose of a step.

The robot is also able to navigate down stairs, on slopes, and over random obstacles but the analysis of these manoeuvres is out of the scope of this paper.

3. MODEL

Because the design was an iterative process especially for the selection of motors and batteries, the model values below were first estimated before constructing the robot, and they were updated to reflect the real robot after it was finished. The robot parameters used in the models were measured from the prototype and are given by:

- $M = 5.012$ kg, total mass of robot (with arm and wheels).

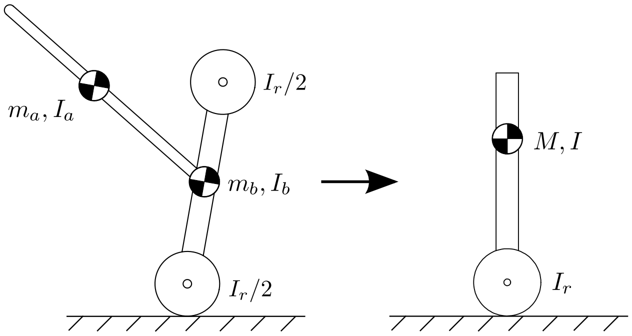
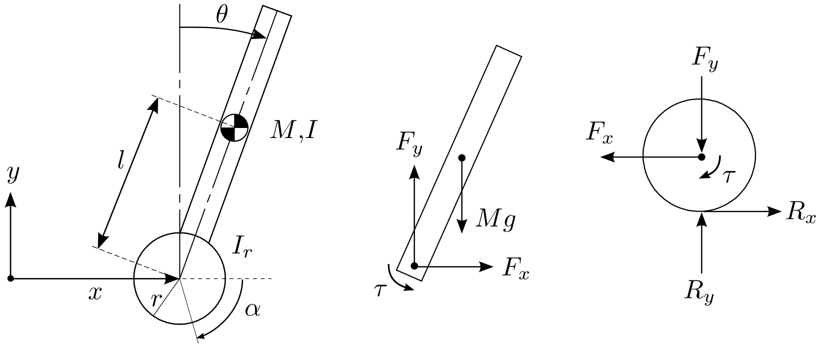


Fig. 2. The arm and the body are both combined into a single mass for the balance analysis (given that the arm is not rotating).



(a) (b) (c)

Fig. 3. (a) Nomenclature of the robot balancing; (b) forces acting on the robot's body; (c) forces acting on the robot's wheel. τ is the control input (torque between wheel and body)

- $r = 0.050$ m, wheel radius.
- $l = l_b = 0.127$ m, half of the distance wheel-to-wheel.
- $I = 0.056$ kg·m^2, robot's moment of inertia.
- $I_r = 0.001$ kg·m^2, rotational components's moment of inertia.

In Sections 3.1 and 3.2 we will derive the model for balancing and for stair-climbing respectively. These two models are unrelated as they are different scenarios. As such, the model derived in one scenario does not apply in the other. In Section 3.1 we will study the model and control of the balancing robot while it balances on flat ground. In Section 3.2 we will develop a model of the forces involved during the stair climbing process. The robot is not balancing while climbing stairs.

3.1. Model for Balancing and Controller Design

For the analysis of the balance, it is assumed that the arm is lumped with the body for simplification purposes. The top wheels are mechanically coupled to the bottom ones so their moment of intertia must be taken into account. As long as the arm does not rotate while balancing, the effect will be the same. If the arm is moving, it is simply considered as a disturbance. For the purpose of analyzing the balancing, the assumption is valid because the arm will not be moving. See Fig. 2.

The analysis will be made following the coordinate system shown in Fig. 3. The robot is broken down into two parts: the body and the wheel.

The position of the wheel is x. In the following equations the wheel is considered massless as its weight is included as part of the body weight M but it still has a mass moment of inertia I_r. This simplification is valid because the wheel is always at the same position relative to the body, but it can rotate relative to it. Applying Newton's second law of motion along x and y coordinates, and with moments along z on Fig. 3 we obtain:

$$I_r\ddot{\alpha} - \tau + Mr(\ddot{\alpha}r + \ddot{\theta}l\cos\theta - \dot{\theta}^2 l\sin\theta) = 0 \tag{1}$$

$$Mgl\sin\theta - \tau = I\ddot{\theta} + Ml^2\ddot{\theta} + Ml^2\dot{\theta}^2\sin^2\theta + Mlr\ddot{\alpha}\cos\theta - Ml^2\dot{\theta}\cos\theta\sin\theta \tag{2}$$

Because the robot is balancing, we assume its body angle θ is stable and is close to zero. Applying small angle approximations $\sin\theta \approx \theta$, $\cos\theta \approx 1$, and $\theta\dot{\theta}^2 \approx 0$ and re-arranging the equations we have:

$$C\ddot{\alpha} = (I + Ml^2 + Mlr)\tau - M^2gl^2r\theta \tag{3}$$

$$C\ddot{\theta} = Mgl(I_r + Mr^2)\theta - (I_r + Mlr + Mr^2)\tau \tag{4}$$

with: $C = IMr^2 + I_rMl^2 + II_r$

From where we write the state space representation as:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u \tag{5}$$

with:

$$\mathbf{x} = \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{Bmatrix} = \begin{Bmatrix} \theta \\ \dot{\theta} \\ \alpha \\ \dot{\alpha} \end{Bmatrix}, \quad \mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \dfrac{Mgl(I_r + Mr^2)}{C} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -\dfrac{M^2gl^2r}{C} & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 \\ -\dfrac{I_r + Mlr + Mr^2}{C} \\ 0 \\ \dfrac{I + Ml^2 + Mlr}{C} \end{bmatrix}, \quad u = \tau$$

The control law for the robot's balancing is defined as:

$$u = -\mathbf{K}\mathbf{x} \tag{6}$$

where **K** needs to be determined. Here we will use the Linear Quadratic Regulator (LQR) method [14] to design an optimal controller. To do so, we define the performance index to be minimized as:

$$J = \int_0^\infty (\mathbf{x}\mathbf{Q}\mathbf{x} + Ru^2)dt \tag{7}$$

with **Q** and R being optimization parameters that need to be chosen. These values need to found by trial and error in the simulation, although as starting point, the Bryson's rule [15] can be applied. To minimize J we use the MATLAB® "lqr" function that takes as parameters the state space model and the **Q** and R constants. The "lqr" function returns **K** for the control law such that J is minimized. After several trials, the **Q** and R parameters that gave a satisfactory response are:

$$\mathbf{Q} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 20 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \text{ and } R = 0.5 \tag{8}$$

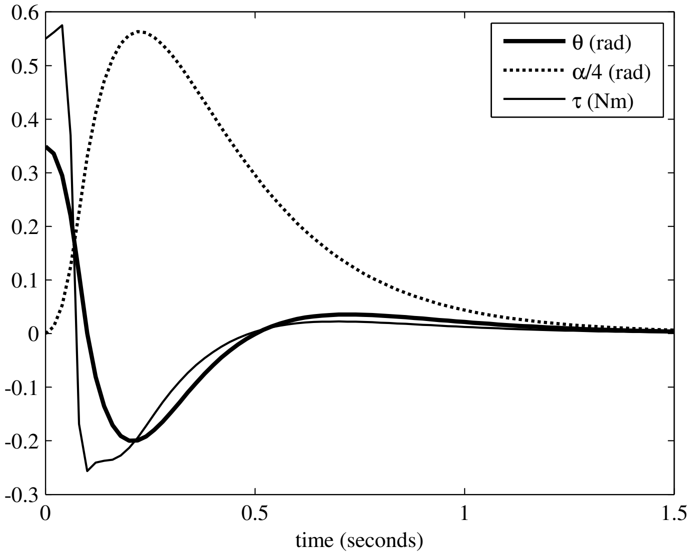


Fig. 4. Simulation of the robot recovering from a 20º disturbance while balancing. The simulation is run with the discrete controller and the nonlinear model of the robot. Sampling rate is 50Hz.

for which the “lqr” function returns:

$$\mathbf{K} = \begin{bmatrix} -101.2 & -15.11 & -6.325 & -3.343 \end{bmatrix} \tag{9}$$

Since the robot is controlled with a microcontroller, the model needs to be converted to a discrete model. The continuous state space model was converted to a discrete state space model using the MATLAB® “c2d” function and a new discrete $\mathbf{K}_d$ was computed using the “dlqr” function. We obtained for a sample rate of 50Hz:

$$\mathbf{K}_d = \begin{bmatrix} -11.07 & -1.473 & -0.5405 & -0.2960 \end{bmatrix} \tag{10}$$

It is also desirable for testing to have a nonlinear model of the robot. Some control systems will work with a linear model but are unstable with the nonlinear version. The control system cannot easily be designed from the nonlinear version, but offers a good validation tool. Re-arranging Eqs. (1) and (2) by extracting $\ddot{\alpha}$ and $\ddot{\theta}$ we obtain the nonlinear equations:

$$\ddot{\alpha} = \frac{\sin\theta(rM^2l^3\dot{\theta}(\dot{\theta}\cos\theta\sin\theta + \dot{\theta} - \cos^2\theta) - grM^2l^2\cos\theta + lrMl\dot{\theta}^2) + (Ml^2 + rMl\cos\theta + I)\tau}{M^2l^2r^2\sin^2\theta + I_rMl^2 + lMr^2 + II_r} \tag{11}$$

$$\ddot{\theta} = \frac{\begin{array}{c}\sin\theta(M^2lr^2\dot{\theta}(l\dot{\theta}\sin\theta - g - l\dot{\theta}\cos\theta + l\cos\theta) + MlI_r(l\dot{\theta}^2\sin\theta + l\dot{\theta}\cos\theta - g)) \\ +(Mr^2 - Mlr\cos\theta + I_r)\tau\end{array}}{M^2l^2r^2\sin^2\theta + I_rMl^2 + lMr^2 + II_r} \tag{12}$$

A discrete model was written and the nonlinear model was augmented with the Pittman GM9236 motor model. It ensures that the current control law is compatible with the real torque limited motors. Figure 4 shows the simulation results for the pendulum subjected to a 20° disturbance. The system successfully recovers in about one second. The settling time could be shorter but then vibrations started to appear on the real model.

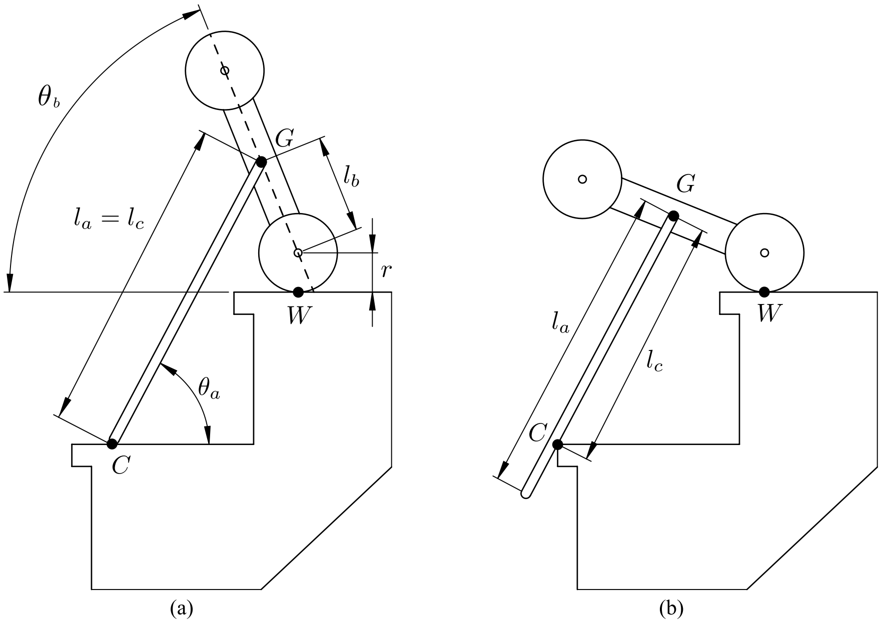


Fig. 5. Schematic of the two cases encountered during the stair-climbing process. (a) Case A: the tip of the arm is in contact with a horizontal surface; (b) case B: the arm is in contact with the nose of a step.

3.2. Model for Stair Climbing

In this section a new model for the robot climbing stairs will be developed. The robot is not balancing while climbing stairs, thus the model developed in Section 3.1 is not applicable here.

During the stair climbing sequence, the robot relies on the friction of the wheel to overcome the normal forces generated by the arm against the steps, and the sliding friction force of the arm. The stair-climbing process can be simplified to have only two different configurations that we call case A and case B. This is a simplification as there exists other configurations (such as when the body of the robot is touching the step instead of the wheels), however these cases are not considered critical due to lower forces involved. Figure 5 shows the two configurations that were analyzed. Figure 6 shows the forces for both cases on the arm and the robot's body.

It is assumed that the robot is currently going upstairs. There is a sliding motion of the arm's contact point and the wheels of the robot are not slipping. A slipping condition would occur whenever force W_x is higher

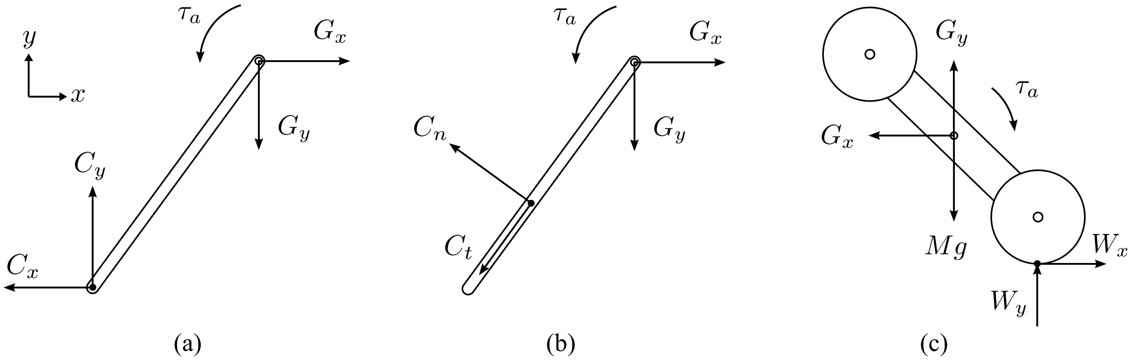


Fig. 6. Free body diagrams of the arm and the robot's body. (a) Forces applied on the arm under case A; (b) forces applied on the arm under case B; (c) forces applied on the robot's body for both case A and case B.

than the friction force the wheel can provide. The arm length was determined such that this never occurs in real stair conditions with a safety factor of at least 2. The condition is:

$$W_x < \mu_w W_y \tag{13}$$

where μ_w is the coefficient of friction between the wheel and the step surface. Also for case A:

$$C_x = \mu_c C_y \tag{14}$$

where μ_c is the coefficient of friction of the arm at its contact point. And for case B:

$$C_t = \mu_c C_n \tag{15}$$

Case B can be re-written to be compatible with the equations of case A by transforming C_n and C_t into C_x and C_y. Then we have:

$$C_x = \begin{cases} \mu_x C_y & \text{if case A} \\ C_n(\sin\theta_a + \mu_c \cos\theta_a) & \text{if case B} \end{cases} \tag{16}$$

$$C_y = \begin{cases} C_y & \text{if case A} \\ C_n(\cos\theta_a - \mu_c \sin\theta_a) & \text{if case B} \end{cases} \tag{17}$$

Applying sum of forces on the arm and body, we obtain a set of equations that can be solved to obtain the arm and wheel torques (the wheel torque is defined as $\tau_w = rW_x$), as a function of the arm angle θ_a, body angle θ_b, and contact length l_c. For case A we obtain:

$$\tau_a = \frac{Mgl_b l_c \cos\theta_b(\cos\theta_a + \mu_c \sin\theta_a)}{D} \tag{18}$$

$$\tau_w = \frac{Mgl_b \mu_c r \cos\theta_b}{D} \tag{19}$$

with: $D = l_b \cos\theta_b - \mu_c r + l_c \cos\theta_A - l_b \mu_c \sin\theta_b + l_c \mu_c \sin\theta_a$.

For case B we obtain:

$$\tau_a = \frac{Mgl_b l_c \cos\theta_b}{E} \tag{20}$$

$$\tau_w = \frac{Mgl_b r(\sin(\theta_a - \theta_b) + \mu_c \cos(\theta_a - \theta_b) + \sin(\theta_a + \theta_b) + \mu_c \cos(\theta_a + \theta_b))}{2E} \tag{21}$$

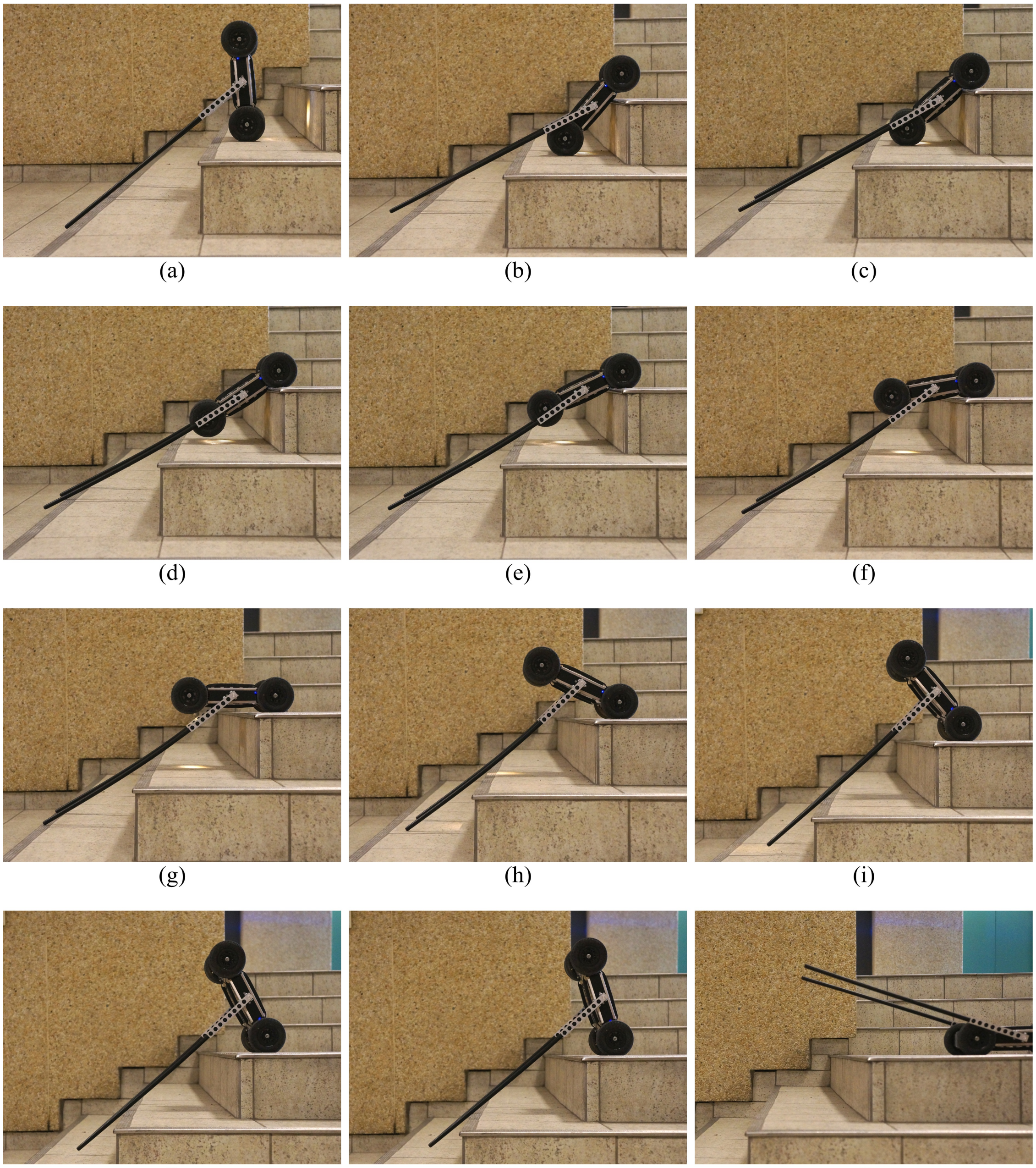
with: $E = l_b \cos(\theta_a + \theta_b) - l_b \mu_c \sin(\theta_a + \theta_b) - r \sin\theta_a + l_c - \mu_c r \cos\theta_a$

4. STAIR-CLIMBING RESULTS

To validate the theory of Section 3.2, the experimental results of the robot climbing a complete step were obtained. Figure 7 shows a sequence of photographs taken during the experiment.

To avoid dynamic effects, the experiment was conducted at a slow rate where the arm was rotating at a speed of about one degree per second. While the experiment was performed, the robot's internal circuitry recorded a number of parameters into a memory card which could then be analyzed offline. For this experiment, the robot logged the following parameters:

- Current time: time was synchronized with the camera's internal time. This allows matching the photographs with the readings in the memory card.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j) (k) (l)

Fig. 7. Sequenced photographs of the robot climbing a single step. The robot starts with its body in a vertical position (a) and finishes at the same position (upside down) on the next step after (k). The same sequence would be repeated for more steps.

- Inertial measurements: the gyroscope and accelerometer readings were recorded to ensure the events on the pictures are well synchronized.
- Motor currents: drive motors and arm motor.
- Battery voltage.

The arm angle θ_a, body angle θ_b, and arm contact length $l_c$were measured using pixel coordinates from the photographs. Whether the robot was in case A or case B was also determined from the picture. The arm an body angles could have also been determined by the logged measurements in the memory card, but were less accurate then measuring on the picture. Referring to Fig. 7, the contact mode is as follow:

- (a): Just before being vertical, the robot is finishing its previous cycle.
- (b): The robot has fallen down with body against step. The arm is raised.
- (c): Arm pushes lightly on the bottom step. Body slipped until the wheel grabs the step run.
- (d) to (f): Case B: the bottom wheel has lifted from the step.
- (g) to (h): Case A: the arm is now pushing against the run of the step
- (i) to (k): Case B.
- (l): The robot passes its equilibrium point and falls (ready for another step).

Once θ_a, θ_b, l_c, and the mode were determined using the photographs, the theoretical torque values were calculated using Eqs. (18–21). The arm coefficient of friction μ_a was assumed to be 0.25 (arm is made of acetal). The logged motor currents were converted to torques using the motor's models. For the arm drivetrain, an estimated efficiency of 0.8 was assumed to account for the triple stage greased planetary gearbox and the chain drive in the arm mechanism. Figures 8 and 9 show a graphical representation of the results for the arm and wheel torques. The annotations refer to the pictures in Fig. 7.

For both figures, we see that the trend is closely followed between both theoretical and experimental data. The torque values are about zero when the body is vertical. This is expected because the center of gravity of the robot is directly above its wheels and the arm does not need to apply any torque. There is discrepancy between the model and the real data for point (b) for the arm torque. It is because the model assumes the bottom wheel is not touching the ground while in reality it is still in contact with it, thus the lower arm torque required. For the wheel torque, point (c) is peaking to almost one Nm, which can happen if the robot tries to move forward but is blocked by the body in contact with the riser (see Fig. 7 (c)).

There are a number of sources of errors. For one, the arm gearbox efficiency and the arm coefficient of friction were assumed based on literature examples but were not explicitly available. Other modelling errors could be possible. Dynamic effects were neglected, although the experiment was conducted at very slow speed. Angle and distance measurements on the photographs are subjected to resolution, visual, and lens distortion errors. The current sensors have an error of ±1.5% plus the error due to the digital-to-analog converter resolution and noise. There are also unmodeled physical properties such as friction, arm bending, wheel deformation, motor dead-band, and backlash. Nonetheless with all these sources of error, it is clear that the model is closely following the experimental data.

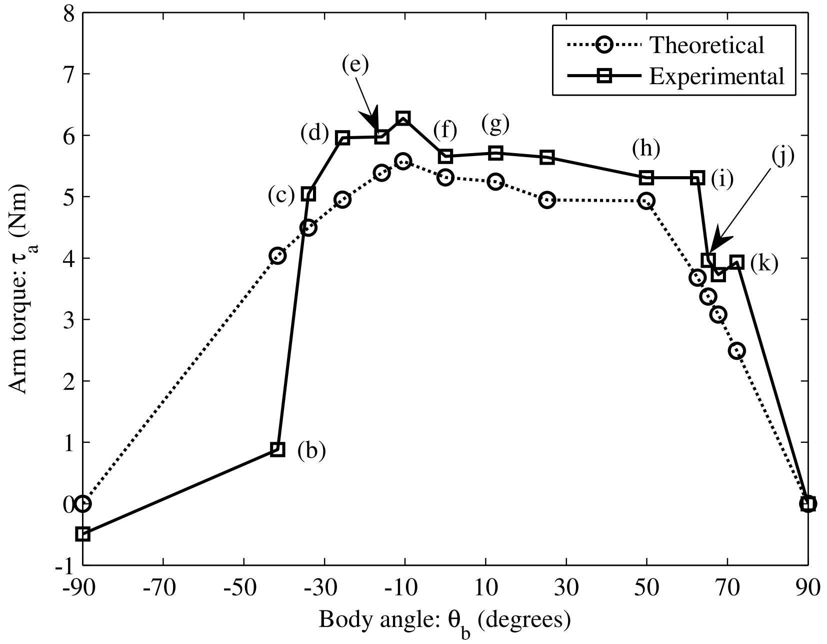


Fig. 8. Torque exerted on the arm of the robot during the climbing sequence shown in Fig. 7.

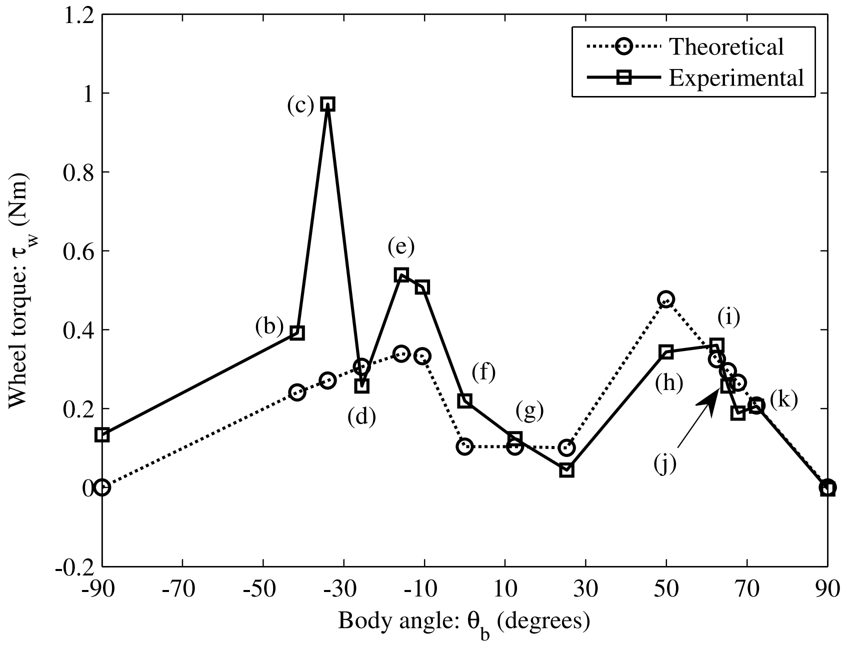


Fig. 9. Torque exerted on the wheels of the robot during the climbing sequence shown in Fig. 7.

5. CONCLUSIONS

In this paper, a simple and effective design that gives self-balancing robots the ability to stand-up and climb obstacles such as stairs was proposed, developed, and tested. An overview of the concept and the design of the balancing control loop were covered, and a model for the analysis of the stair-climbing torques in both the arm and the wheels was developed. This new stair-climbing model was then validated using experimental data on the real robot. We showed that the addition of two long arms in the center of a four-wheel drive robot allows it to stand-up for balancing and navigate stairs. The single extra degree of freedom (for a total of three counting the drive train) increases the field capabilities of self-balancing robots.

REFERENCES

1. Vu, Q.H., Kim, B.S. and Song, J.B. "Autonomous stair climbing algorithm for a small four-tracked robot." In "International Conference on Control, Automation and Systems, 2008. ICCAS 2008," pp. 2356–2360. doi: 10.1109/ICCAS.2008.4694199, October 2008.
2. Ben-Tzvi, P., Ito, S. and Goldenberg, A. "Autonomous stair climbing with reconfigurable tracked mobile robot." In "International Workshop on Robotic and Sensors Environments, 2007. ROSE 2007," pp. 1–6. doi: 10.1109/ROSE.2007.4373976, October 2007.
3. Taylor, B.K., Balakirsky, S., Messina, E. and Quinn, R.D. "Modeling, validation and analysis of a whegs robot in the USARSim environment." In "Proceedings of SPIE 6962, Unmanned Systems Technology X, 69621B," pp. 69621B–69621B. doi:10.1117/12.777604, April 2008.
4. "Segway – the leader in personal, green transportation." `http://www.segway.com/`, November 2012.
5. Yamafuji K., Feng M. Q., K.T. "Robots driven by parallel bicycles." *International Journal of Mechanics and Control*, Vol. 9, No. 2, pp. 3–11, 2008.
6. Jeong, S. and Takahashi, T. "Wheeled inverted pendulum type assistant robot: design concept and mobile control." *Intelligent Service Robotics*, Vol. 1, No. 4, pp. 313–320. ISSN 1861-2776, 1861-2784. doi:10.1007/s11370-008-0024-5, October 2008.
7. Tang, R. and Green, R. "Obstacle avoidance on a mobile inverted pendulum robot." In "Image and Vision Computing New Zealand, 2009. IVCNZ '09. 24th International Conference," pp. 254 –259. doi: 10.1109/IVCNZ.2009.5378400, November 2009.
8. Anderson, D.P. "nBot, a two wheel balancing robot." `http://www.geology.smu.edu/dpa-www/robo/nbot/`, November 2012.
9. Feng, T., Liu, T., Wang, X., Xu, Z., Zhang, M. and Han, S.c. "Modeling and implementation of two-wheel self-balancing robot equipped with supporting arms." In "2011 6th IEEE Conference on Industrial Electronics and Applications (ICIEA)," pp. 713 –718. doi:10.1109/ICIEA.2011.5975678, June 2011.
10. Takei, T., Imamura, R. and Yuta, S. "Baggage transportation and navigation by a wheeled inverted pendulum mobile robot." *IEEE Transactions on Industrial Electronics*, Vol. 56, No. 10, pp. 3985 –3994. ISSN 0278-0046. doi:10.1109/TIE.2009.2027252, October 2009.
11. Kumagai, M. and Ochiai, T. "Development of a robot balancing on a ball." In "International Conference on Control, Automation and Systems, 2008. ICCAS 2008," pp. 433 –438. doi:10.1109/ICCAS.2008.4694680, October 2008.
12. Chen, C.Y., Shih, B.Y., Shih, C.H. and Wang, L.H. "Enhancing robust and stability control of a humanoid biped robot: system identification approach." *Journal of Vibration and Control*. ISSN 1077-5463, 1741-2986. doi:10.1177/1077546312442947, April 2012.
13. Miao, S. and Cao, Q. "Modeling of self-tilt-up motion for a two-wheeled inverted pendulum." *Industrial Robot: An International Journal*, Vol. 38, No. 1, pp. 76–85. ISSN 0143-991X. doi:10.1108/01439911111097878, January 2011.
14. Stefani, R.T., Shahian, B., Savant, C.J. and Hostetter, G.H. *Design of Feedback Control Systems*. 4 ed.. Oxford University Press, USA. ISBN 0195142497, August 2001.
15. Jr, A.E.B. and Ho, Y.C. *Applied Optimal Control: Optimization, Estimation, and Control*. Hemisphere Publishing Company. ISBN 9780891162285, 1975.

Appendix H

Simulink model of the hybrid stabilization system

This appendix shows the implementation details of the Simulink model used in Section 4.1.4.1. The main view for the Simulink model is shown in Fig. H.1. All following figures (Fig. H.2 to Fig. H.7) show the content in the various blocks in the model. The blocks called "Switch" are all implemented the same way and is only shown for the feedback switch in Fig. H.2. These switches are selected before the simulation is run. The observer initial condition block in Fig. H.3 is a custom function that takes the outputs $\boldsymbol{y}$ and places them in their right position in $\boldsymbol{x}_0$. The code below is the MATLAB code inside the MATLAB embedded function implementing the nonlinear system in Fig. H.6:

```
function x_dot = calc_x_dot(x_,u)
%Load the system properties
[...] %code to load system parameters from workspace

%Extract the current state
theta = x_(1);
theta_d = x_(2);
x_d = x_(4);
phi = x_(5);
phi_d = x_(6);
x = x_(3);

%Control input
F = u(1);
tau = u(2);

%Intermediary calculations:
V=theta_d^2*sin(2*phi)*(JY-JZ)/2;
W=2*sin(phi)*phi_d*theta_d*(JY-Jw_p);
X=Jrz+JZ*cos(phi)^2+JY*sin(phi)^2+l^2*m;
Y=m*g*l*sin(theta)-cos(phi)*(W+Jw*phi_d*psi_d);
Z=M*X+l^2*m^2*cos(theta)*(sin(theta)-cos(theta));

%Update the state
x_dot=x_;
x_dot(1)=theta_d;
x_dot(2)=(F*m*l*cos(theta)+M*Y)/Z;
x_dot(3)=x_d;
x_dot(4)=(F*X+m*l*(cos(theta)-sin(theta))*Y)/Z;
x_dot(5)=phi_d;
x_dot(6)=(tau-b*phi_d-k*phi+V+Jw*psi_d*theta_d*cos(phi))/JX;
```

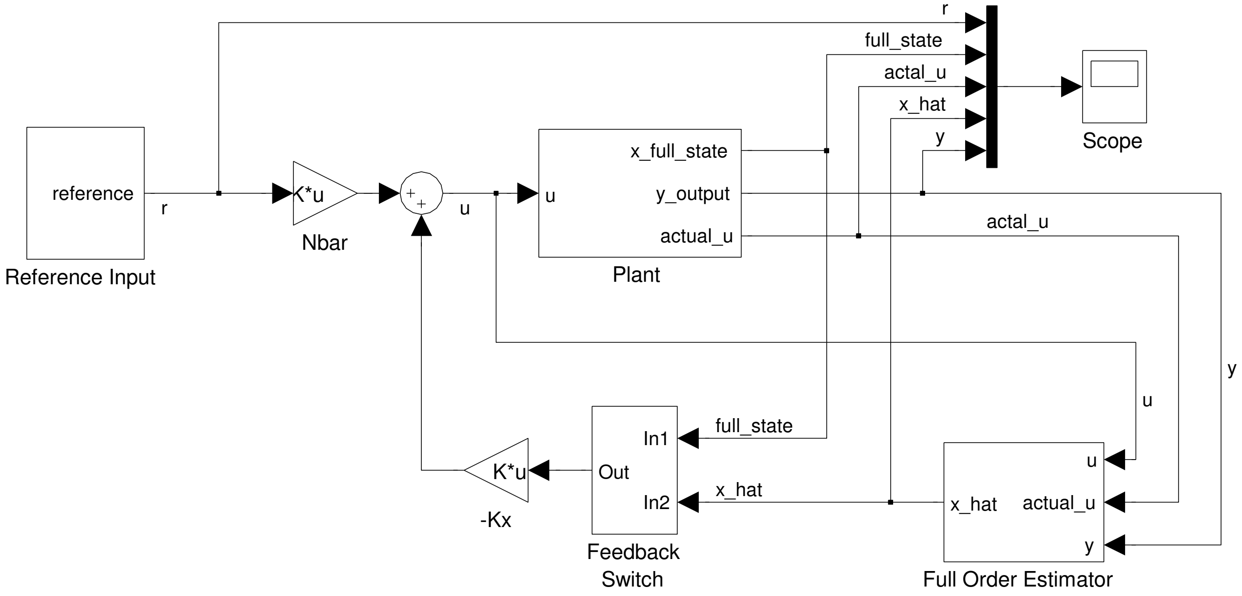


Figure H.1: Full Simulink model (main block).

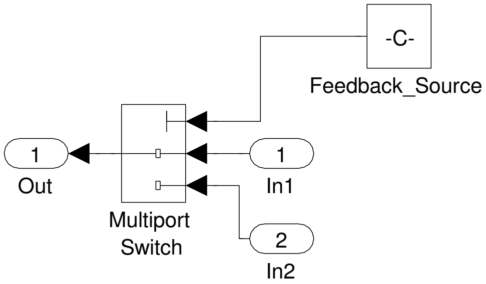


Figure H.2: Feedback switch block of the Simulink model. All the "switch" blocks are made the same way.

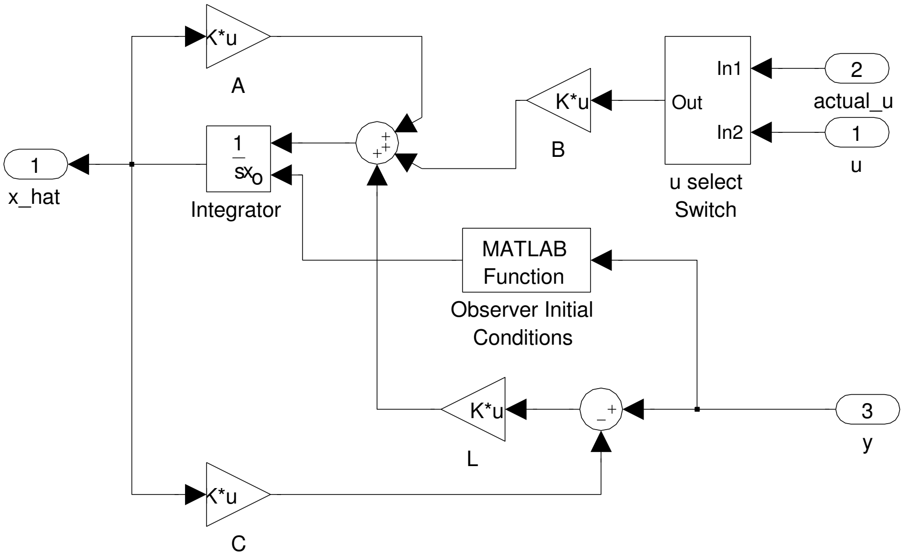


Figure H.3: Full order estimator block of the Simulink model.

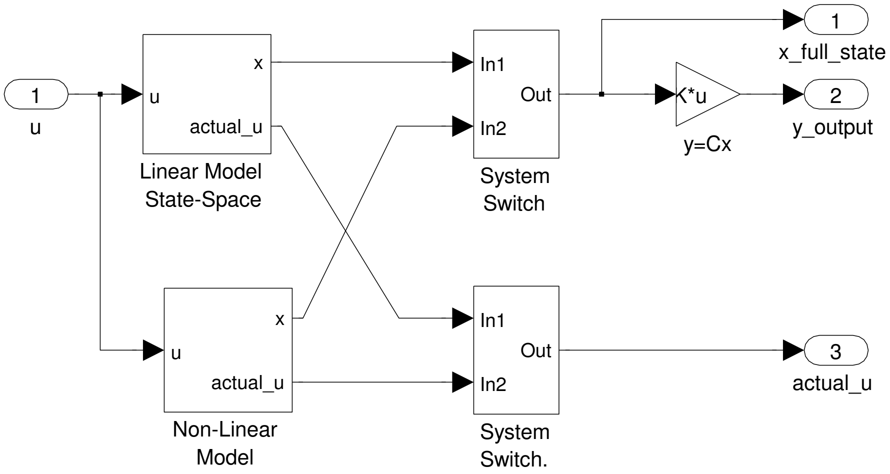


Figure H.4: Plant block of the Simulink model.

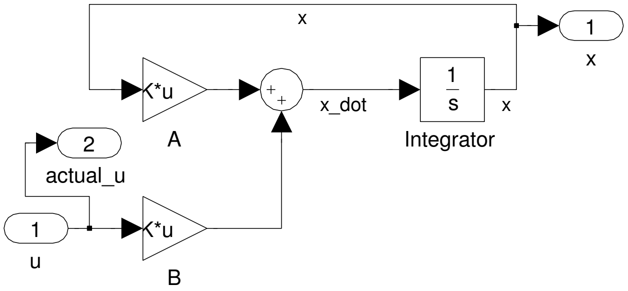


Figure H.5: Linear model state-space block of the Simulink model.

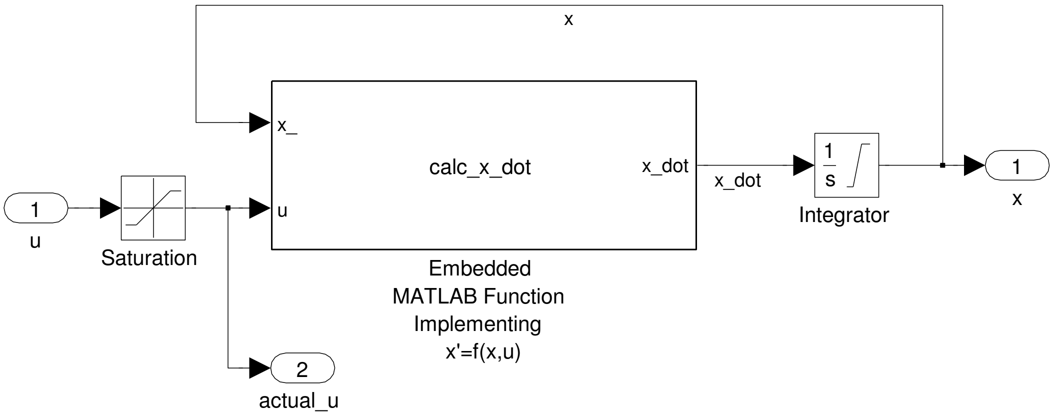


Figure H.6: Nonlinear model block of the Simulink model.

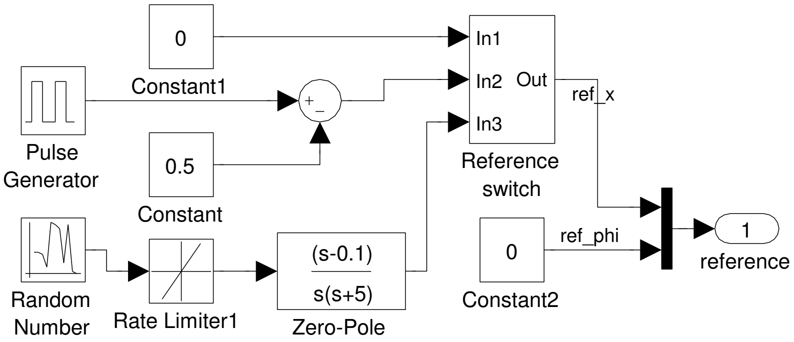
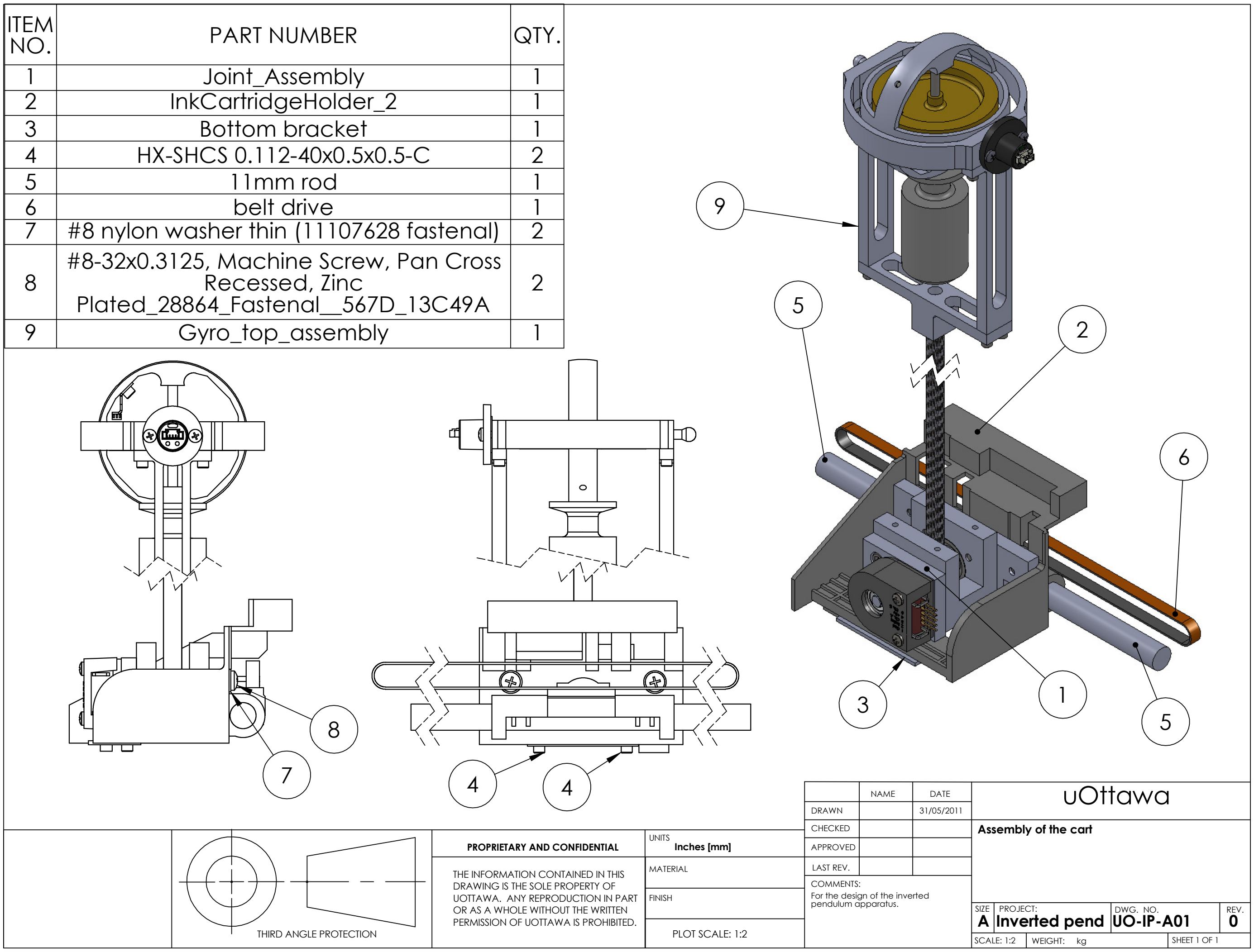


Figure H.7: Reference input block of the Simulink model.

Appendix I

Design drawings for the hybrid inverted pendulum

The digital CAD files are also available for download here [26].



ITEM NO.	PART NUMBER	QTY.
1	Joint_Assembly	1
2	InkCartridgeHolder_2	1
3	Bottom bracket	1
4	HX-SHCS 0.112-40x0.5x0.5-C	2
5	11mm rod	1
6	belt drive	1
7	#8 nylon washer thin (11107628 fastenal)	2
8	#8-32x0.3125, Machine Screw, Pan Cross Recessed, Zinc Plated_28864_Fastenal__567D_13C49A	2
9	Gyro_top_assembly	1

	NAME	DATE
DRAWN		31/05/2011
CHECKED		
APPROVED		
LAST REV.		

uOttawa

Assembly of the cart

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS Inches [mm]

MATERIAL

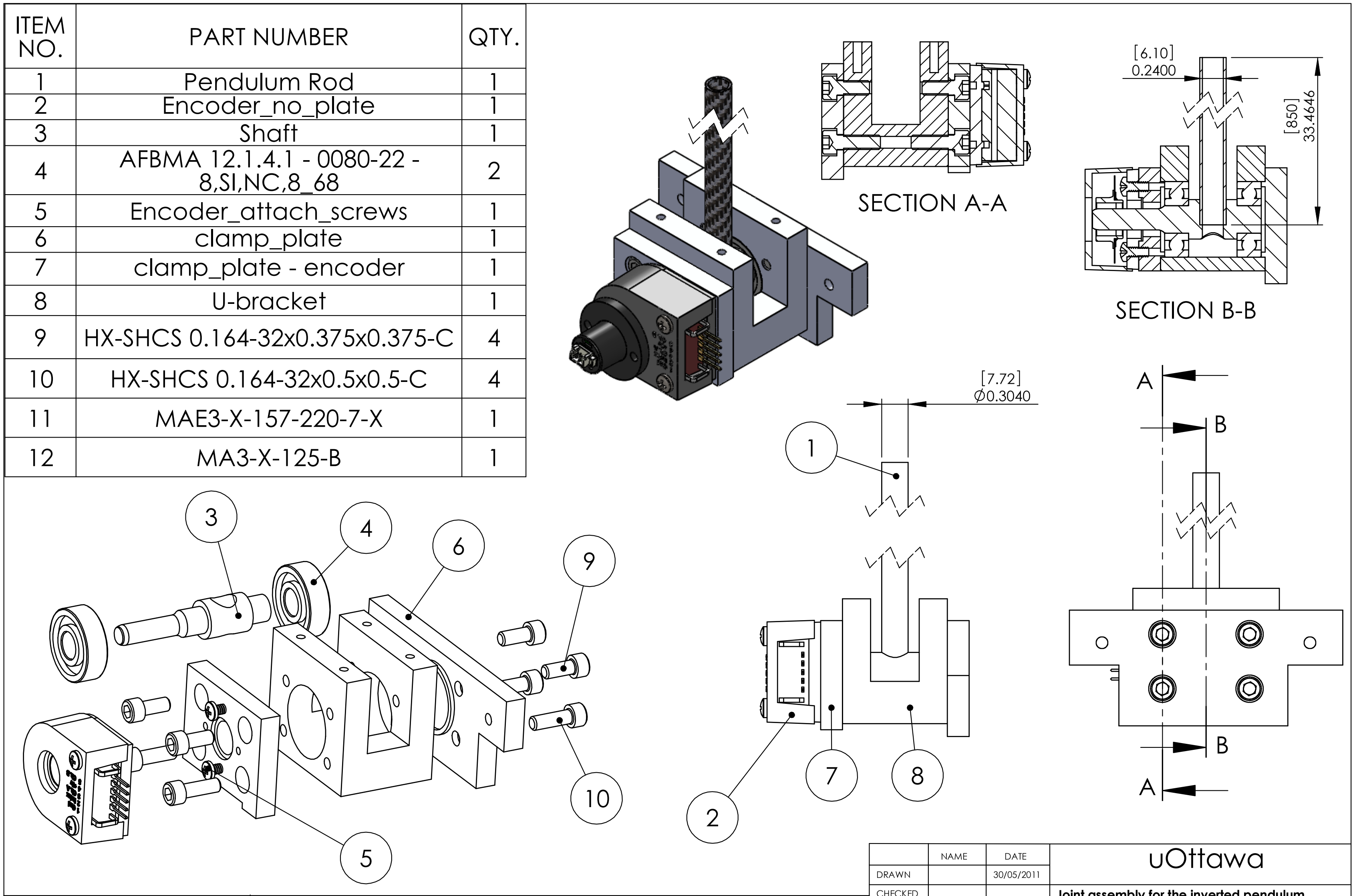
FINISH

PLOT SCALE: 1:2

COMMENTS:
For the design of the inverted pendulum apparatus.

SIZE A
PROJECT: Inverted pend
DWG. NO. UO-IP-A01
REV. 0
SCALE: 1:2
WEIGHT: kg
SHEET 1 OF 1

ITEM NO.	PART NUMBER	QTY.
1	Pendulum Rod	1
2	Encoder_no_plate	1
3	Shaft	1
4	AFBMA 12.1.4.1 - 0080-22 - 8,SI,NC,8_68	2
5	Encoder_attach_screws	1
6	clamp_plate	1
7	clamp_plate - encoder	1
8	U-bracket	1
9	HX-SHCS 0.164-32x0.375x0.375-C	4
10	HX-SHCS 0.164-32x0.5x0.5-C	4
11	MAE3-X-157-220-7-X	1
12	MA3-X-125-B	1



	NAME	DATE
DRAWN		30/05/2011
CHECKED		
APPROVED		
LAST REV.		

COMMENTS:
For the design of the inverted pendulum apparatus.

uOttawa

Joint assembly for the inverted pendulum

SIZE **A** PROJECT: **Inverted pend** DWG. NO. **UO-IP-A02** REV. **0**

SCALE: 2:3 WEIGHT: kg SHEET 1 OF 1

UNITS **Inches [mm]**

MATERIAL

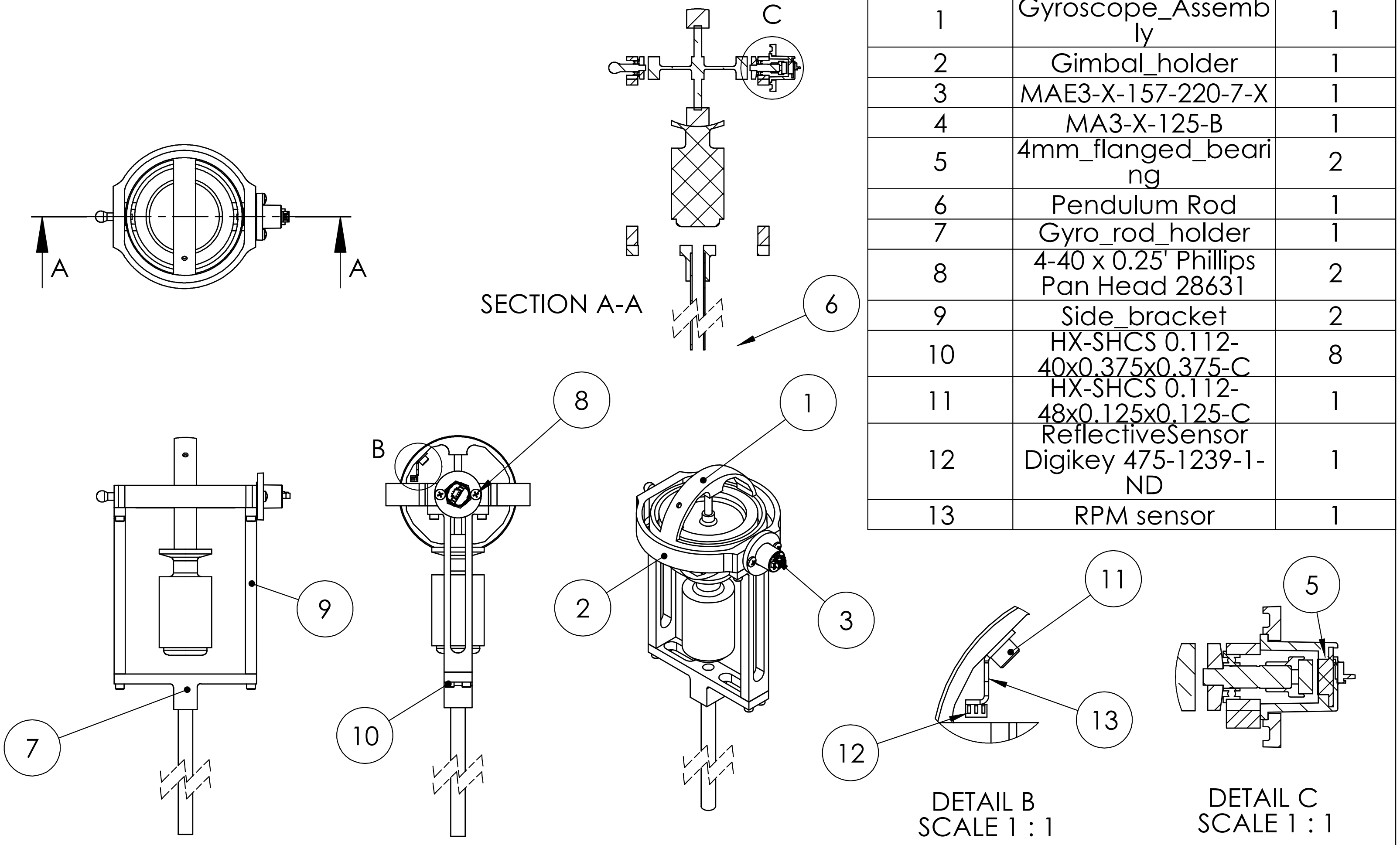
FINISH

PLOT SCALE: 2:3

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

THIRD ANGLE PROTECTION



ITEM NO.	PART NUMBER	QTY.
1	Gyroscope_Assembly	1
2	Gimbal_holder	1
3	MAE3-X-157-220-7-X	1
4	MA3-X-125-B	1
5	4mm_flanged_bearing	2
6	Pendulum Rod	1
7	Gyro_rod_holder	1
8	4-40 x 0.25' Phillips Pan Head 28631	2
9	Side_bracket	2
10	HX-SHCS 0.112-40x0.375x0.375-C	8
11	HX-SHCS 0.112-48x0.125x0.125-C	1
12	ReflectiveSensor Digikey 475-1239-1-ND	1
13	RPM sensor	1

	NAME	DATE
DRAWN		30/05/2011
CHECKED		
APPROVED		
LAST REV.		

COMMENTS:
For the design of the inverted pendulum apparatus.

uOttawa

Top portion of the inverted pendulum assembly. The precision gyroscope is incorporated on a custom gimbal and rotation sensors.

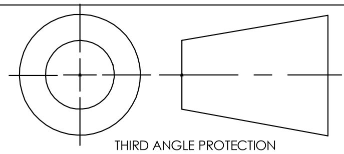
SIZE	PROJECT:	DWG. NO.	REV.
A	**Inverted pend**	**UO-IP-A03**	**0**

SCALE: 1:3
WEIGHT: kg
SHEET 1 OF 1

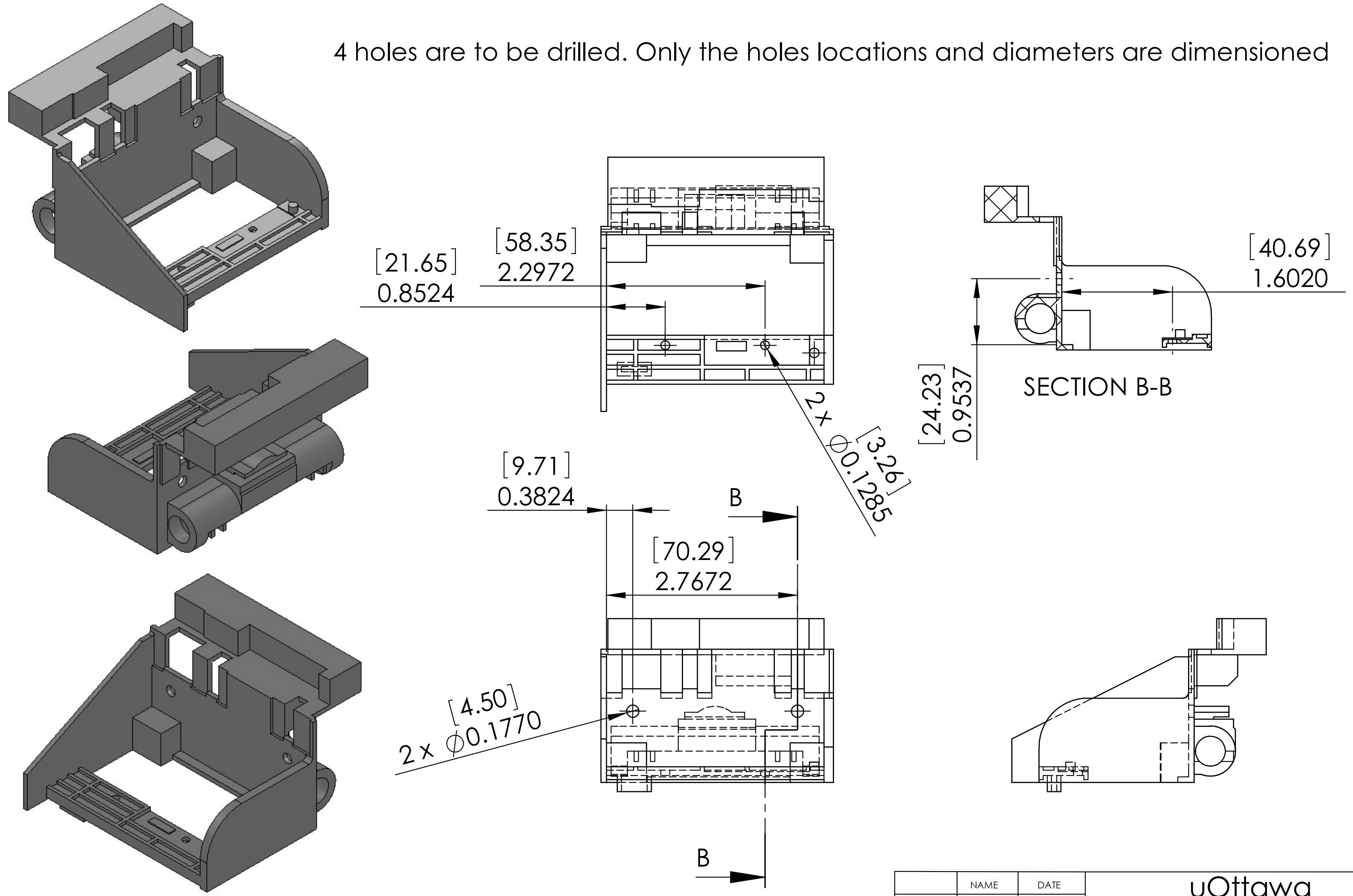
UNITS
Inches [mm]
MATERIAL
FINISH
PLOT SCALE: 1:3

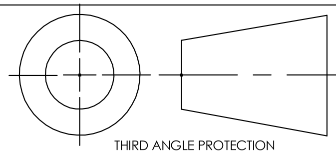
PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.


THIRD ANGLE PROTECTION

4 holes are to be drilled. Only the holes locations and diameters are dimensioned




THIRD ANGLE PROTECTION

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**
MATERIAL
FINISH
PLOT SCALE: 1:2

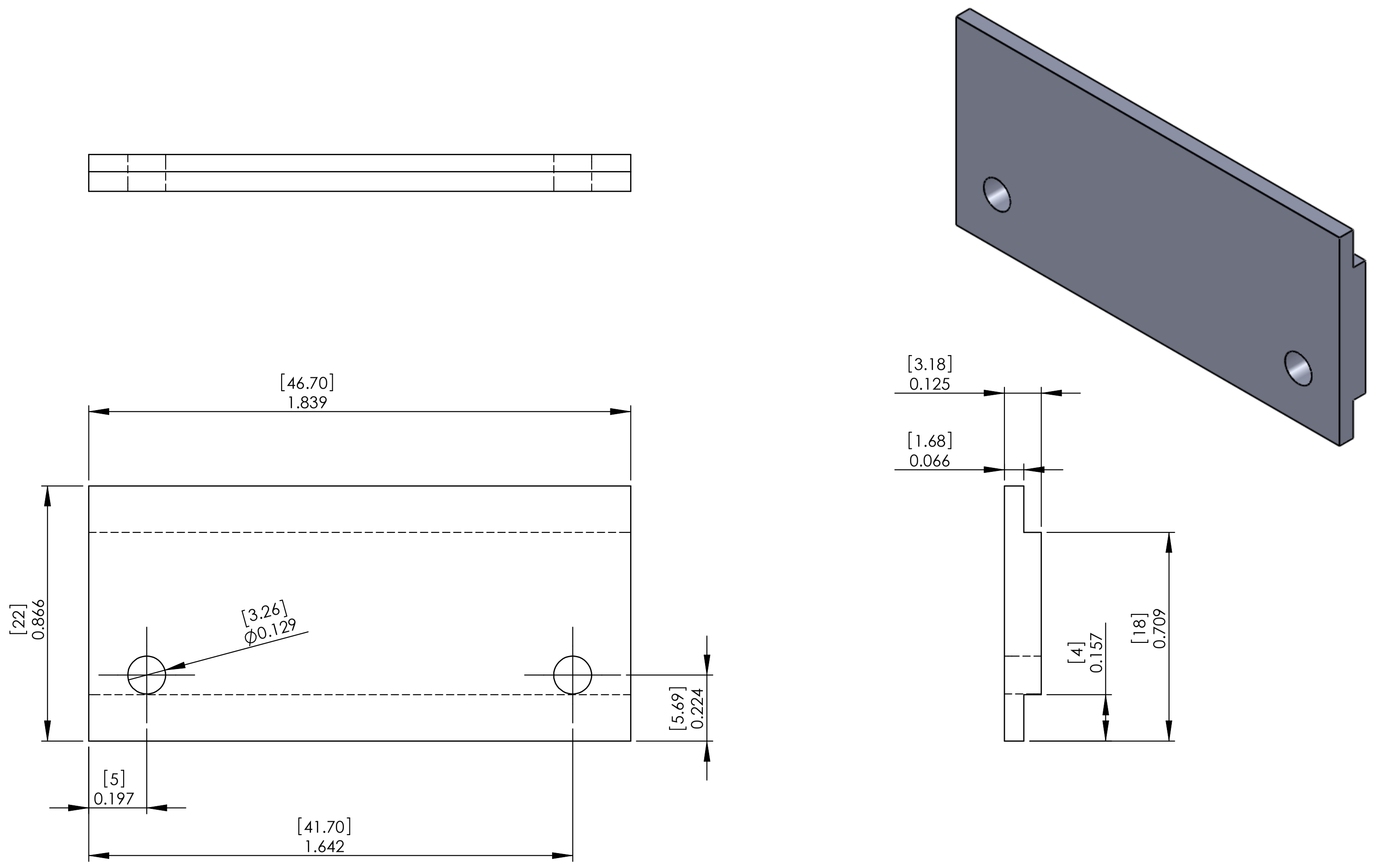
	NAME	DATE
DRAWN		31/05/2011
CHECKED		
APPROVED		
LAST REV.		

COMMENTS:
For the design of the inverted pendulum apparatus.

uOttawa

Original printer cartridge holder modified to hold the joint assembly of the inverted pendulum.

SIZE **A**
PROJECT: **Inverted pend**
DWG. NO. **UO-IP-001**
REV. **0**
SCALE: 1:2
WEIGHT: kg
SHEET 1 OF 1



THIRD ANGLE PROTECTION

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**

MATERIAL

FINISH

PLOT SCALE: 2:1

	NAME	DATE
DRAWN		26/07/2011
CHECKED		
APPROVED		
LAST REV.		

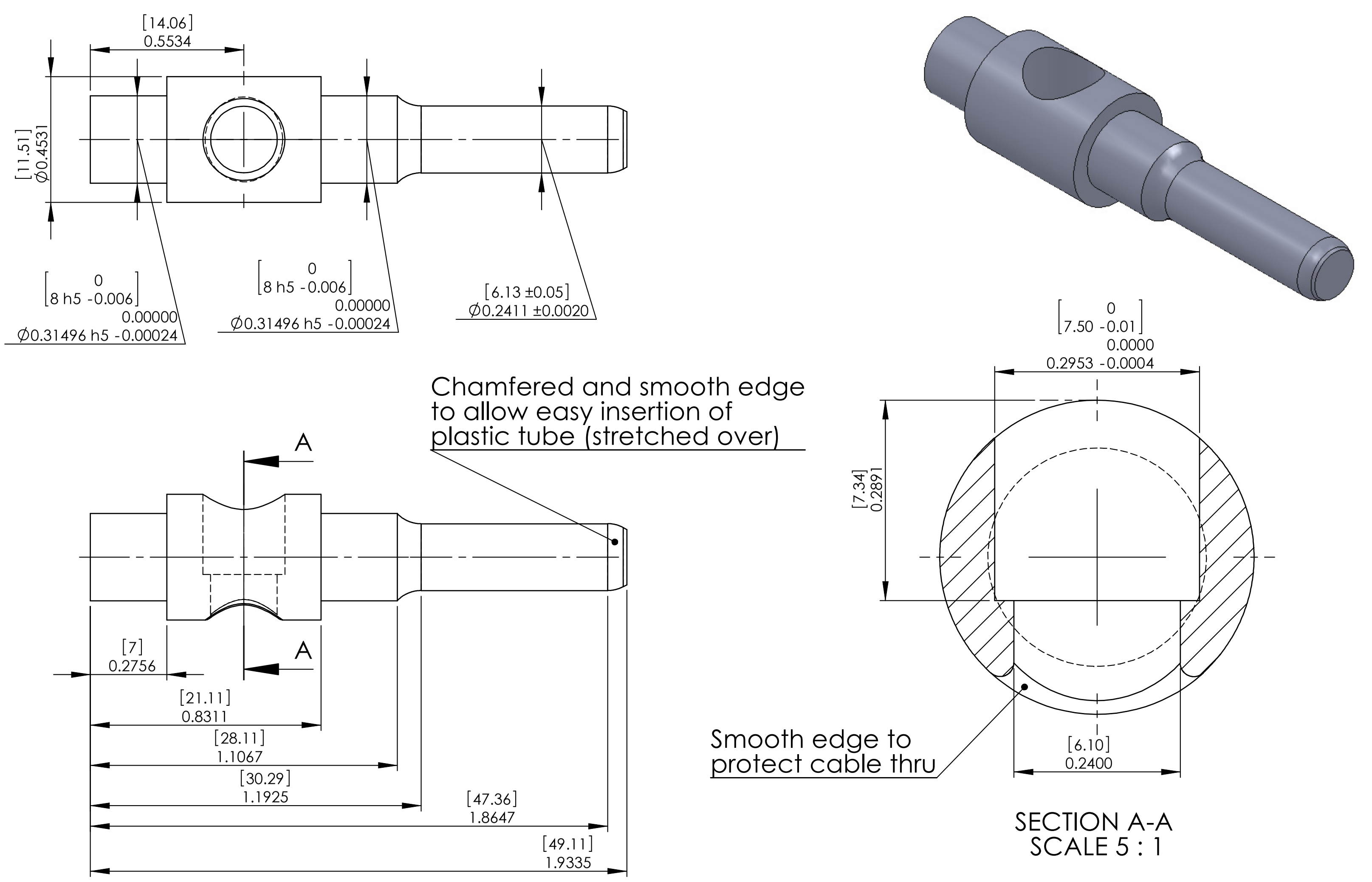
COMMENTS:
For the design of the inverted pendulum apparatus.

uOttawa

Bracket to clamp the joint assembly on the printer catridge holder.

SIZE	PROJECT:	DWG. NO.	REV.
A	**Inverted pend**	**UO-IP-002**	**0**

SCALE: 2:1 WEIGHT: kg SHEET 1 OF 1



Chamfered and smooth edge to allow easy insertion of plastic tube (stretched over)

Smooth edge to protect cable thru

SECTION A-A
SCALE 5 : 1

uOttawa

	NAME	DATE
DRAWN		31/05/2011
CHECKED		
APPROVED		
LAST REV.		

Shaft

COMMENTS:
For the design of the inverted pendulum apparatus.

UNITS **Inches [mm]**

MATERIAL

FINISH

PLOT SCALE: 2:1

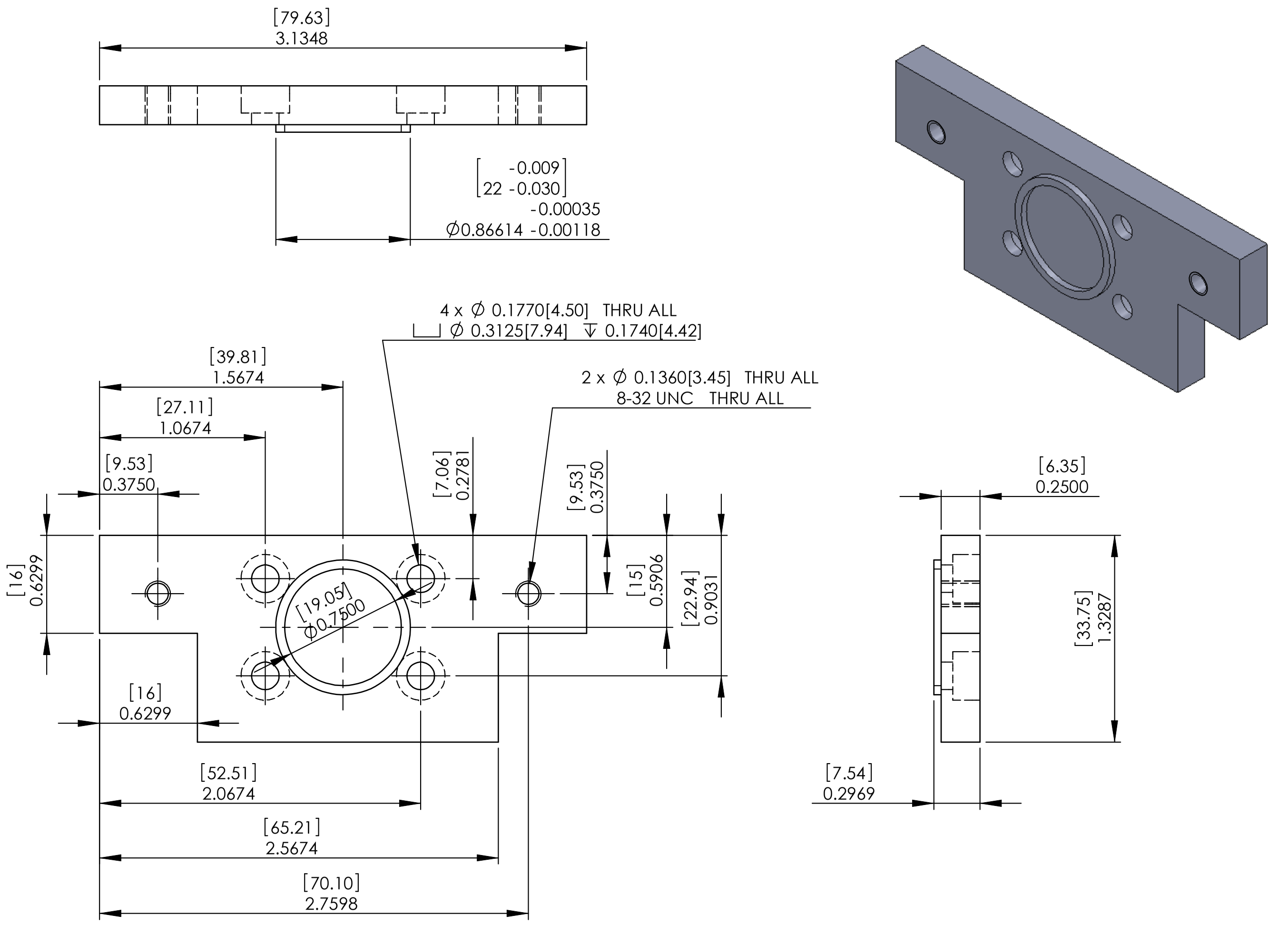
PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

THIRD ANGLE PROTECTION

SIZE	PROJECT:	DWG. NO.	REV.
A	**Inverted pend**	**UO-IP-003**	**0**

SCALE: 2:1 WEIGHT: kg SHEET 1 OF 1



	NAME	DATE
DRAWN		31/05/2011
CHECKED		
APPROVED		
LAST REV.		

COMMENTS:
For the design of the inverted pendulum apparatus.

uOttawa

Holder plate on opposite side from the encoder

SIZE **A** PROJECT: **Inverted pend** DWG. NO. **UO-IP-004** REV. **0**

SCALE: 1:1 WEIGHT: kg SHEET 1 OF 1

UNITS **Inches [mm]**

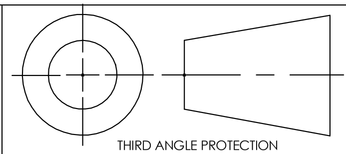
MATERIAL

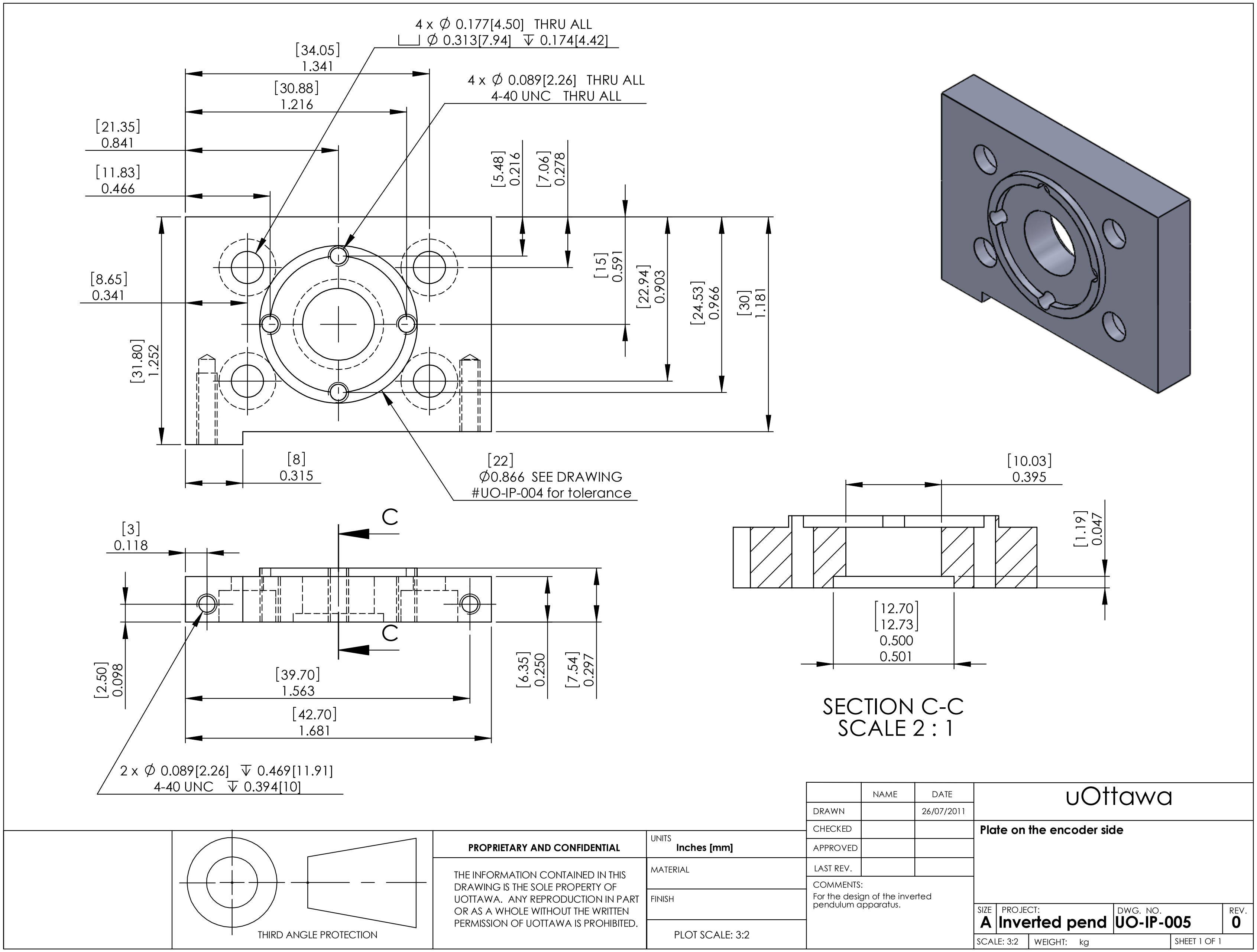
FINISH

PLOT SCALE: 1:1

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.


THIRD ANGLE PROTECTION



	NAME	DATE
DRAWN		26/07/2011
CHECKED		
APPROVED		
LAST REV.		

COMMENTS:
For the design of the inverted pendulum apparatus.

uOttawa

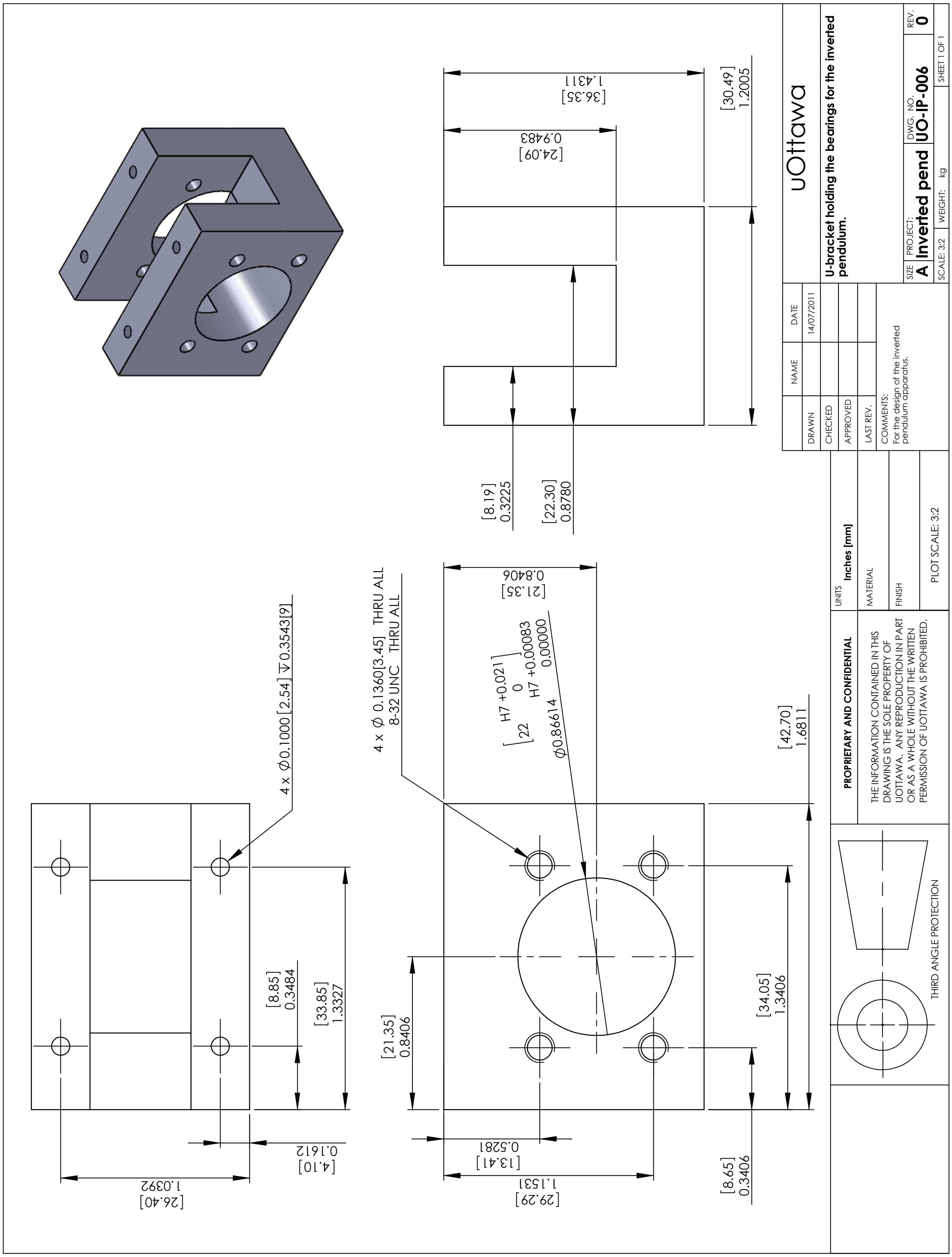
Plate on the encoder side

SIZE A
PROJECT: **Inverted pend**
DWG. NO. **UO-IP-005**
REV. **0**
SCALE: 3:2
WEIGHT: kg
SHEET 1 OF 1

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**
MATERIAL
FINISH
PLOT SCALE: 3:2



uOttawa

U-bracket holding the bearings for the inverted pendulum.

	NAME	DATE
DRAWN		14/07/2011
CHECKED		
APPROVED		
LAST REV.		

COMMENTS:
For the design of the inverted pendulum apparatus.

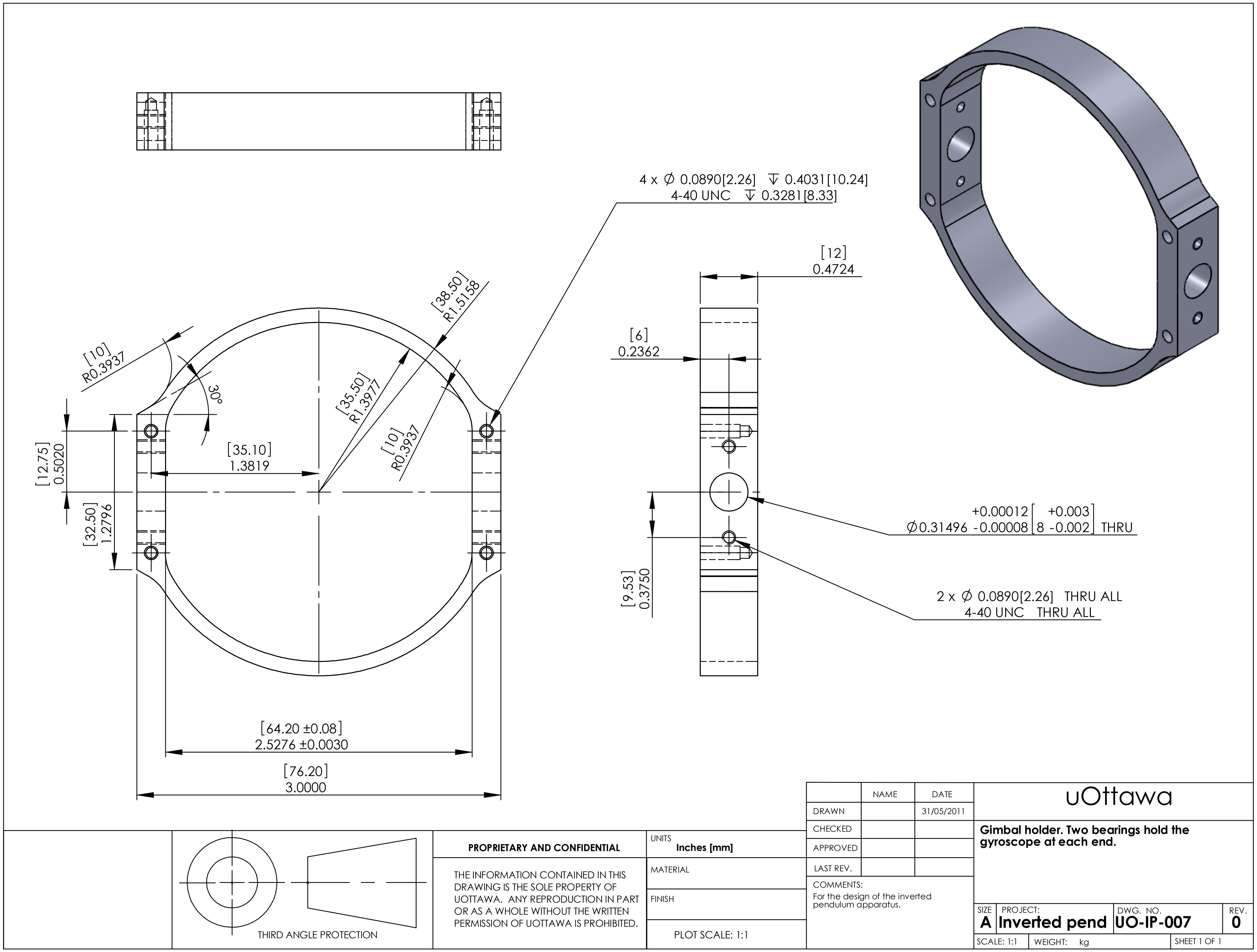
UNITS **Inches [mm]**
MATERIAL
FINISH

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

SIZE **A** PROJECT: **Inverted pend** DWG. NO. **UO-IP-006** REV. **0**

SCALE: 3:2 WEIGHT: kg SHEET 1 OF 1



PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**

MATERIAL

FINISH

PLOT SCALE: 1:1

	NAME	DATE
DRAWN		31/05/2011
CHECKED		
APPROVED		
LAST REV.		

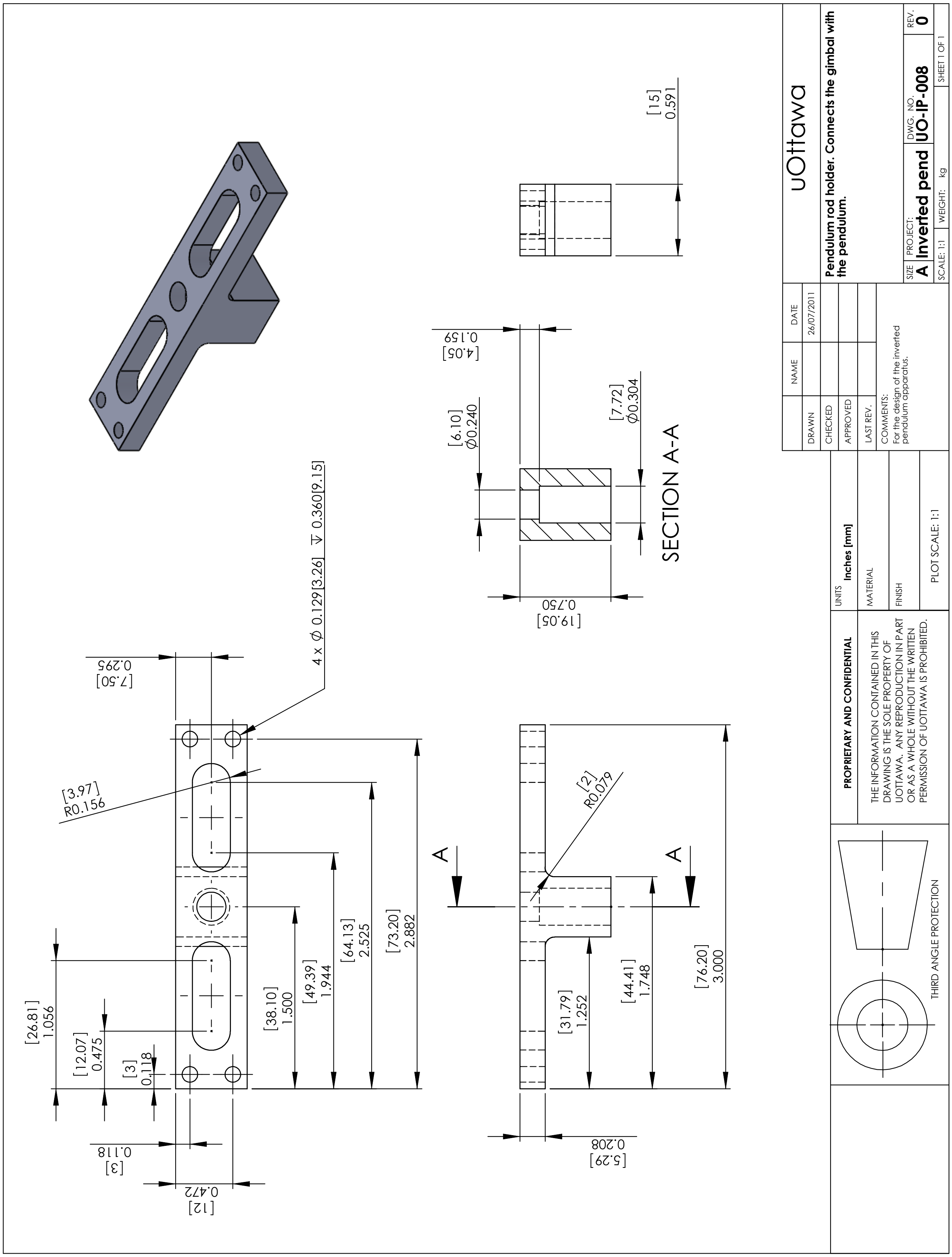
COMMENTS:
For the design of the inverted pendulum apparatus.

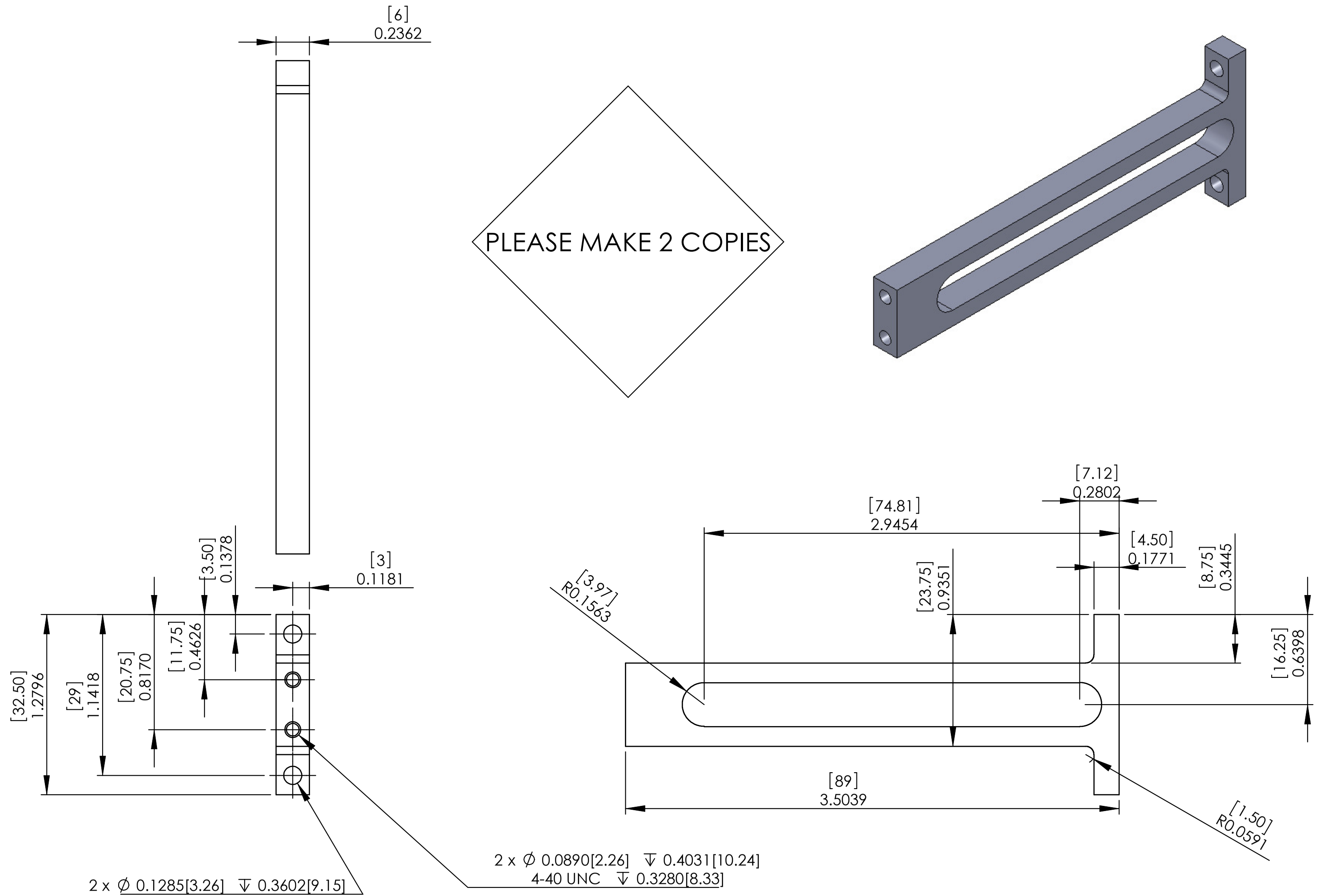
uOttawa

Gimbal holder. Two bearings hold the gyroscope at each end.

SIZE **A** PROJECT: **Inverted pend** DWG. NO. **UO-IP-007** REV. **0**

SCALE: 1:1 WEIGHT: kg SHEET 1 OF 1


uOttawa
Pendulum rod holder. Connects the gimbal with the pendulum.
NAME
DATE
DRAWN
26/07/2011
CHECKED
APPROVED
LAST REV.
COMMENTS:
For the design of the inverted pendulum apparatus.
SIZE
A
PROJECT:
Inverted pend
DWG. NO.
UO-IP-008
REV.
0
SCALE: 1:1
WEIGHT: kg
SHEET 1 OF 1
UNITS
Inches [mm]
MATERIAL
FINISH
PLOT SCALE: 1:1
PROPRIETARY AND CONFIDENTIAL
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.
THIRD ANGLE PROTECTION
SECTION A-A
A
A
4 x Ø 0.129[3.26] ↧ 0.360[9.15]
[3.97]
R0.156
[7.50]
0.295
[26.81]
1.056
[12.07]
0.475
[3]
0.118
[3]
0.118
[12]
0.472
[38.10]
1.500
[49.39]
1.944
[64.13]
2.525
[73.20]
2.882
[2]
R0.079
[31.79]
1.252
[44.41]
1.748
[76.20]
3.000
[5.29]
0.208
[19.05]
0.750
[6.10]
Ø0.240
[7.72]
Ø0.304
[4.05]
0.159
[15]
0.591



PLEASE MAKE 2 COPIES

THIRD ANGLE PROTECTION

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

UNITS **Inches [mm]**

MATERIAL

FINISH

PLOT SCALE: 1:1

	NAME	DATE
DRAWN		31/05/2011
CHECKED		
APPROVED		
LAST REV.		

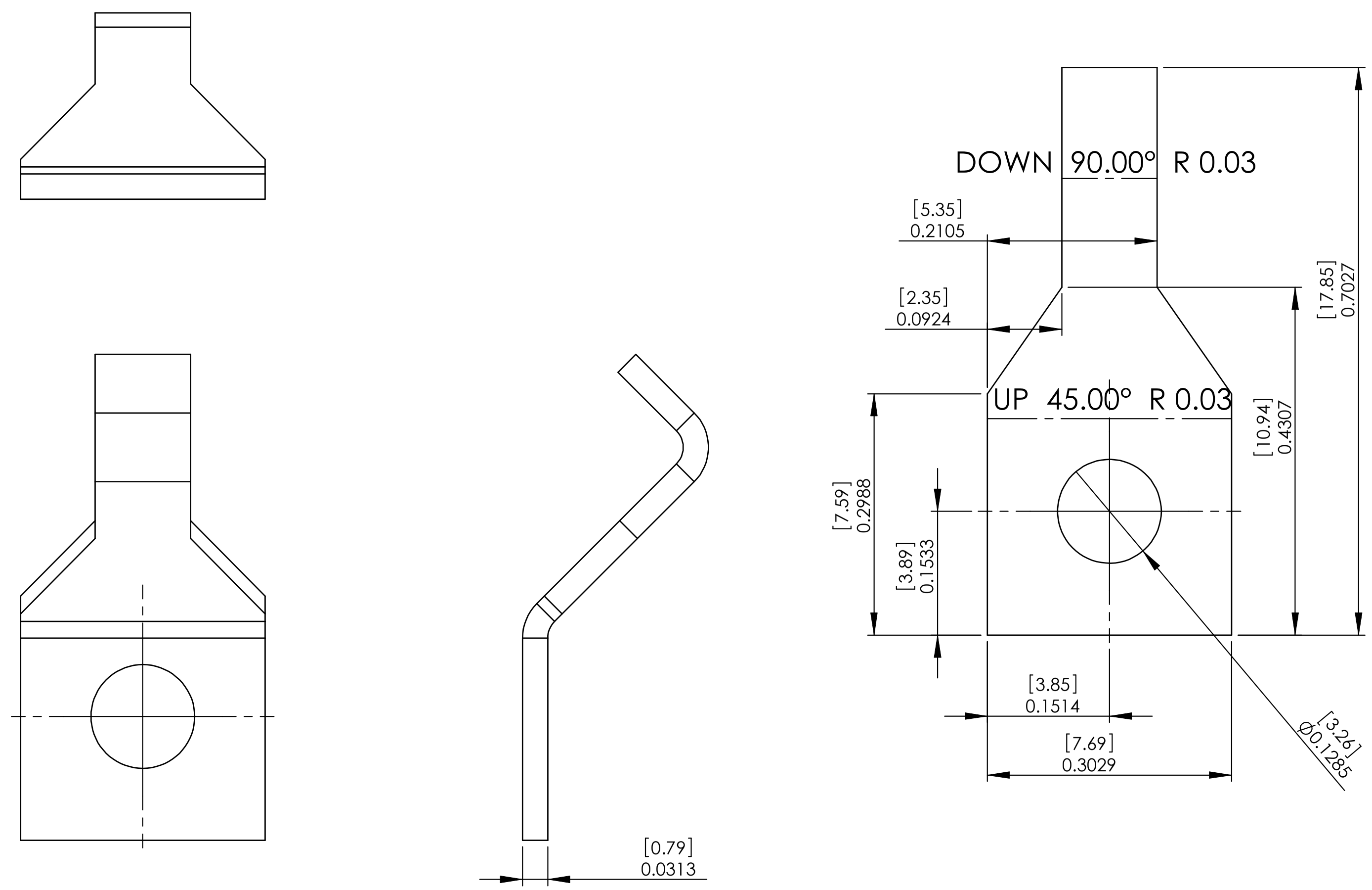
COMMENTS:
For the design of the inverted pendulum apparatus.

uOttawa

Side holders for the gimbal assembly

SIZE **A** PROJECT: **Inverted pend** DWG. NO. **UO-IP-009** REV. **0**

SCALE: 1:1 WEIGHT: kg SHEET 1 OF 1



	NAME	DATE
DRAWN		31/05/2011
CHECKED		
APPROVED		
LAST REV.		

uOttawa

Small sheet metal sensor holder.

COMMENTS:
For the design of the inverted pendulum apparatus.

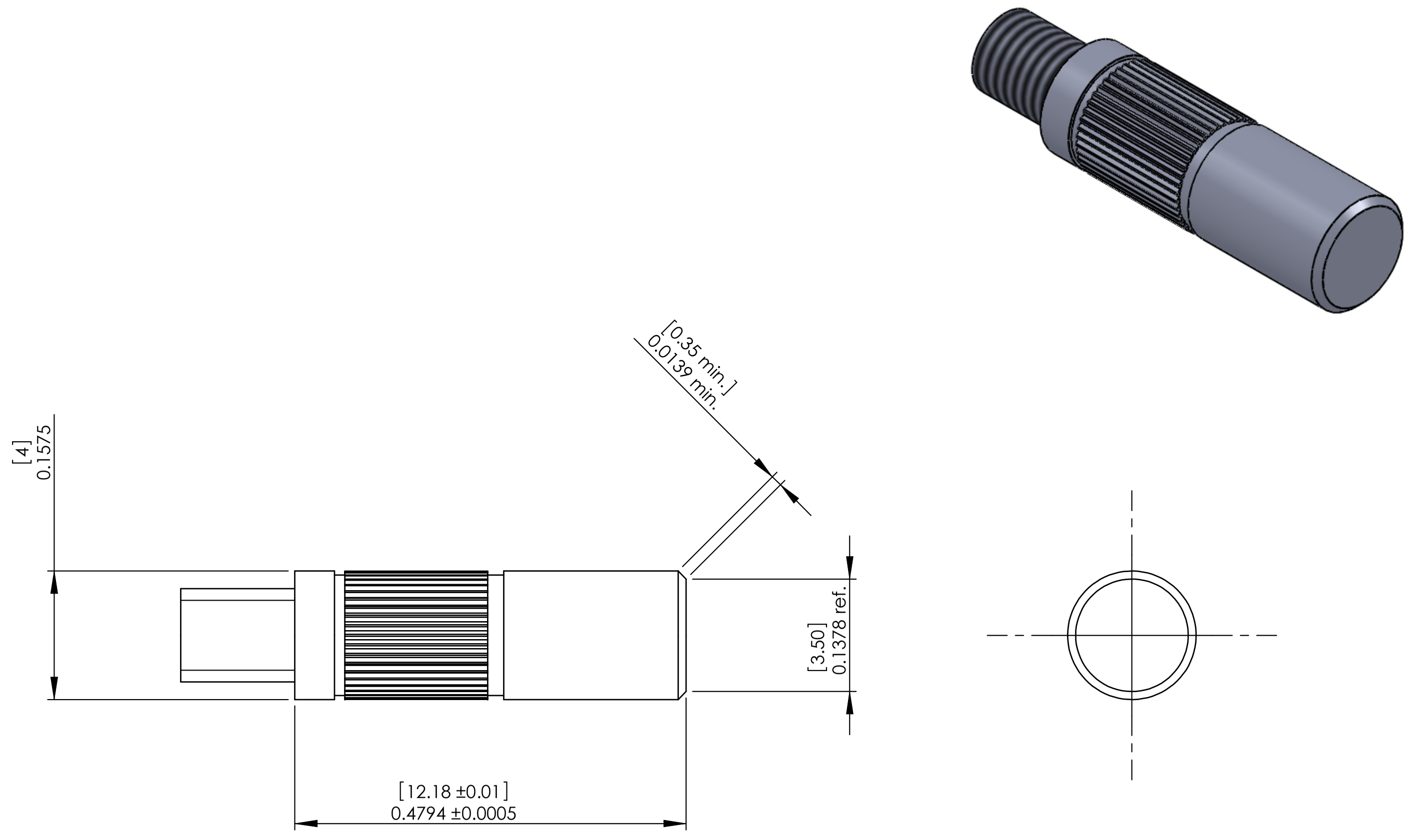
SIZE **A**
PROJECT: **Inverted pend**
DWG. NO. **UO-IP-010**
REV. **0**
SCALE: 1:1
WEIGHT: kg
SHEET 1 OF 1

UNITS **Inches [mm]**
MATERIAL
FINISH
PLOT SCALE: 1:1

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

THIRD ANGLE PROTECTION



	NAME	DATE
DRAWN		31/05/2011
CHECKED		
APPROVED		
LAST REV.		

COMMENTS:
For the design of the inverted pendulum apparatus.

uOttawa

Gimbal shaft modification. Cut to lenght following specified tolerance.

SIZE **A** PROJECT: **Inverted pend** DWG. NO. **UO-IP-011** REV. **0**

SCALE: 5:1 WEIGHT: kg SHEET 1 OF 1

UNITS **Inches [mm]**

MATERIAL

FINISH

PLOT SCALE: 5:1

PROPRIETARY AND CONFIDENTIAL

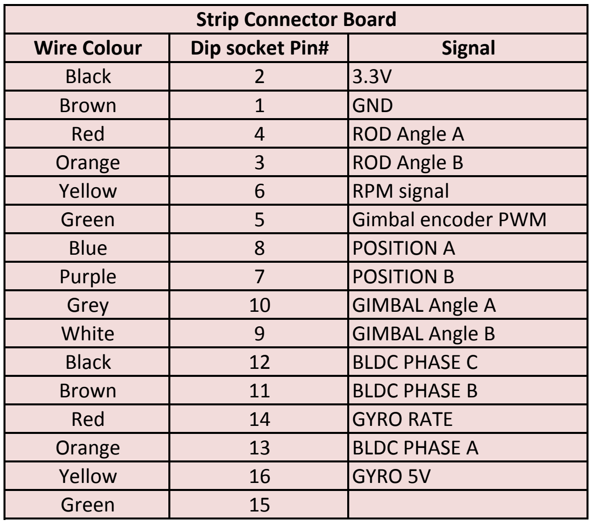
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF UOTTAWA. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF UOTTAWA IS PROHIBITED.

THIRD ANGLE PROTECTION

Appendix J

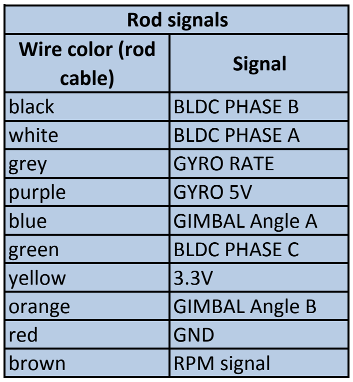
Circuit schematic and pinout for the hybrid inverted pendulum apparatus

The digital files are also available for download here [26].



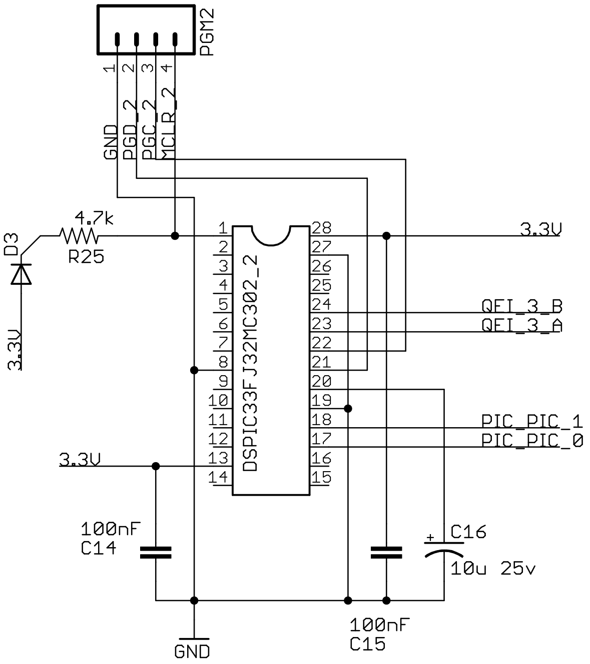
Strip Connector Board

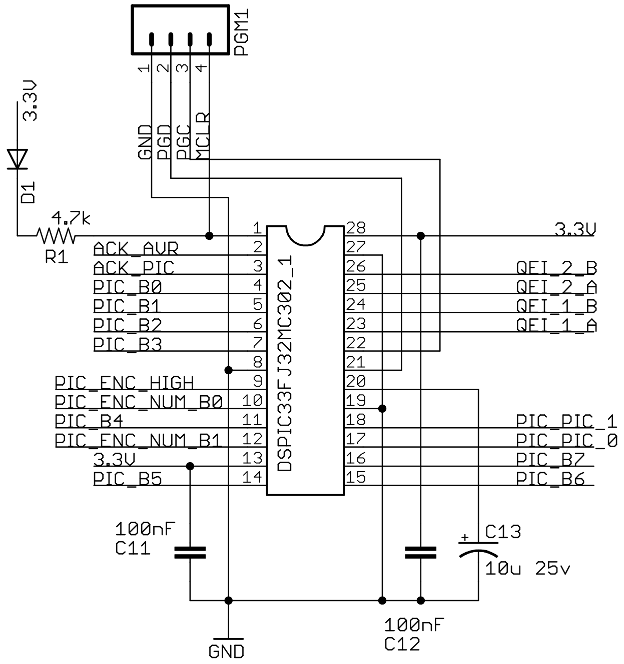
Wire Colour	Dip socket Pin#	Signal
Black	2	3.3V
Brown	1	GND
Red	4	ROD Angle A
Orange	3	ROD Angle B
Yellow	6	RPM signal
Green	5	Gimbal encoder PWM
Blue	8	POSITION A
Purple	7	POSITION B
Grey	10	GIMBAL Angle A
White	9	GIMBAL Angle B
Black	12	BLDC PHASE C
Brown	11	BLDC PHASE B
Red	14	GYRO RATE
Orange	13	BLDC PHASE A
Yellow	16	GYRO 5V
Green	15	

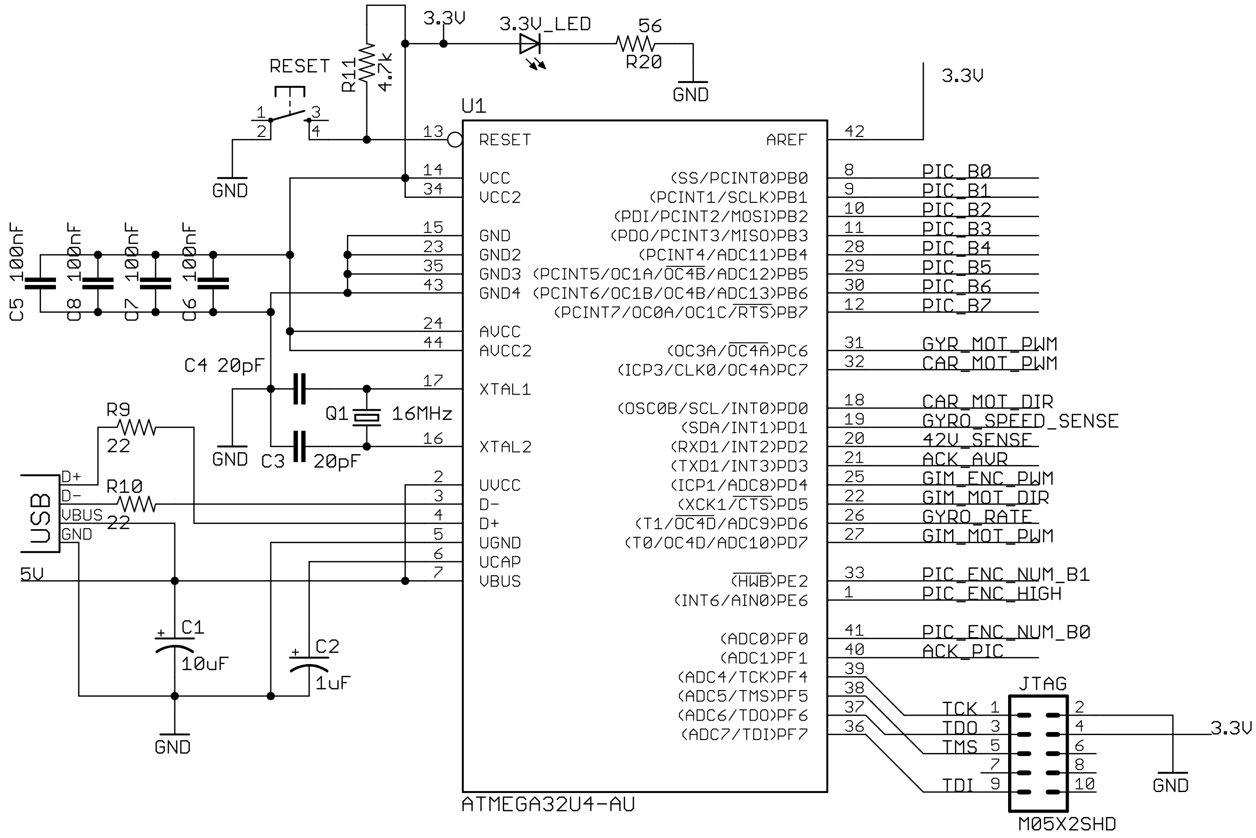


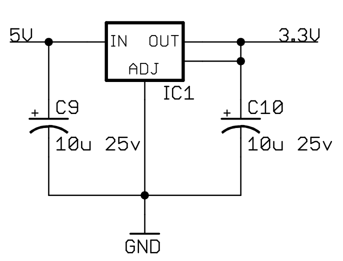
Rod signals

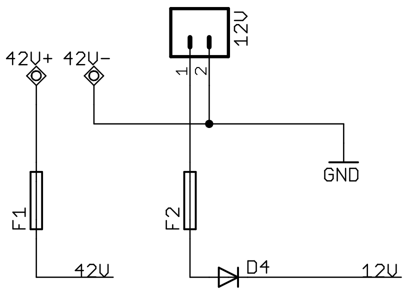
Wire color (rod cable)	Signal
black	BLDC PHASE B
white	BLDC PHASE A
grey	GYRO RATE
purple	GYRO 5V
blue	GIMBAL Angle A
green	BLDC PHASE C
yellow	3.3V
orange	GIMBAL Angle B
red	GND
brown	RPM signal

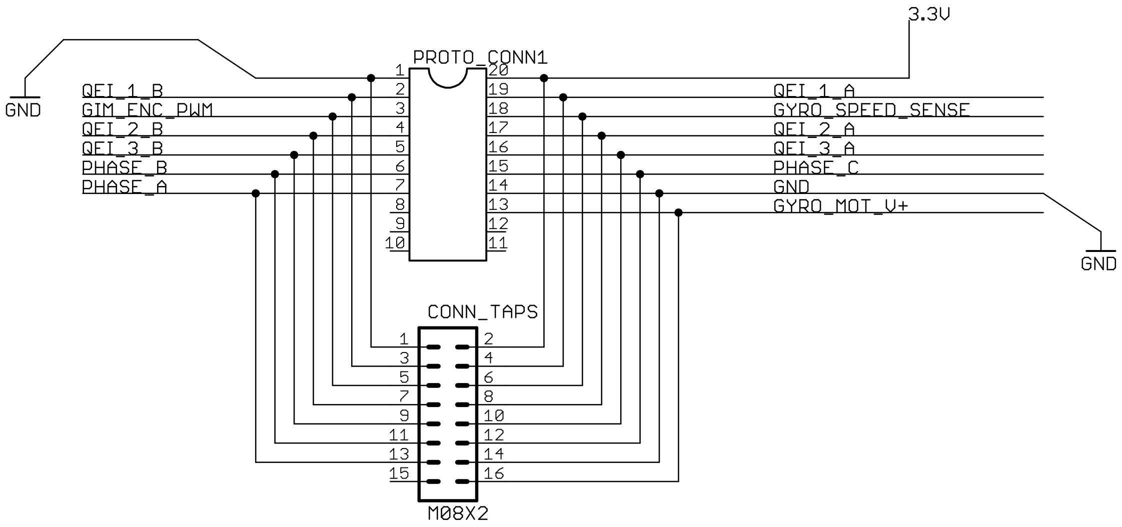

PGM2
1 2 3 4
GND
PGD_2
PGC_2
MCLR_2
4.7k
R25
D3
3.3V
DSPIC33FJ32MC302_2
3.3V
QEI_3_B
QEI_3_A
PIC_PIC_1
PIC_PIC_0
3.3V
100nF
C14
C16
10u 25v
100nF
C15
GND

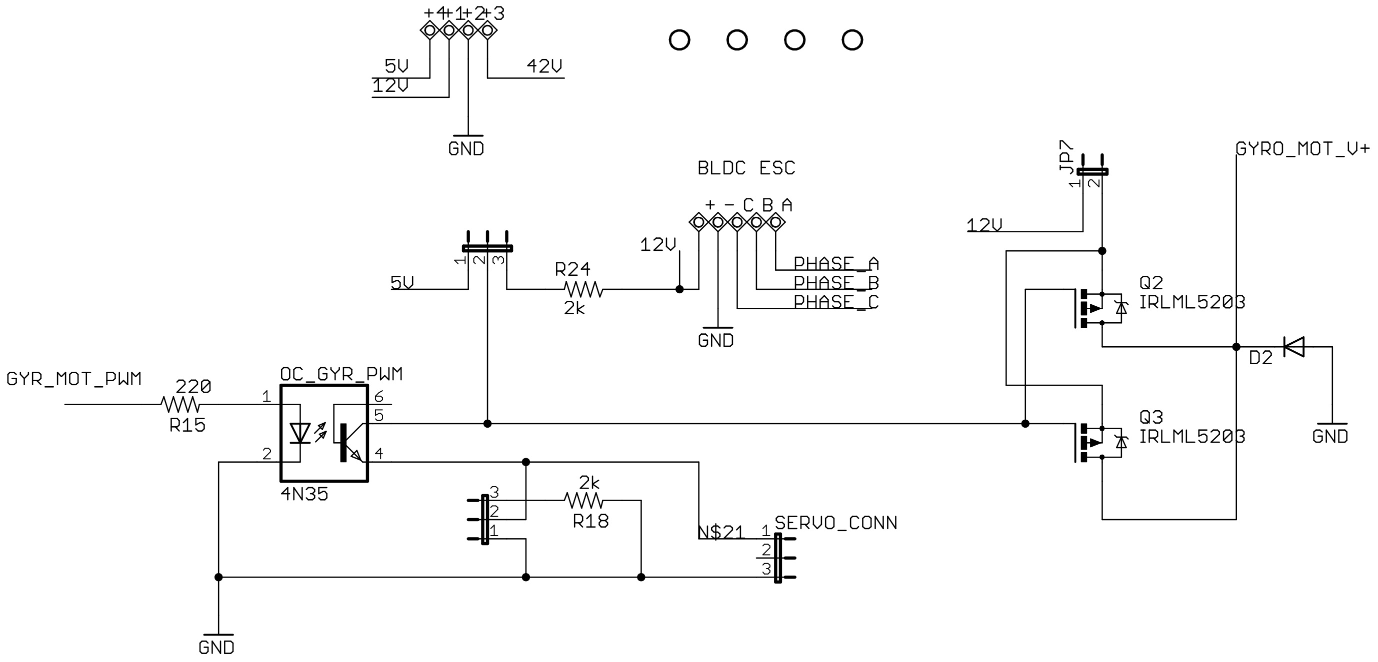

PGM1
1 2 3 4
GND
PGD
PGC
MCLR
3.3V
D1
4.7k
R1
ACK_AVR
ACK_PIC
PIC_B0
PIC_B1
PIC_B2
PIC_B3
PIC_ENC_HIGH
PIC_ENC_NUM_B0
PIC_B4
PIC_ENC_NUM_B1
3.3V
PIC_B5
DSPIC33FJ32MC302_1
3.3V
QEI_2_B
QEI_2_A
QEI_1_B
QEI_1_A
PIC_PIC_1
PIC_PIC_0
PIC_B7
PIC_B6
100nF
C11
C13
10u 25v
100nF
C12
GND

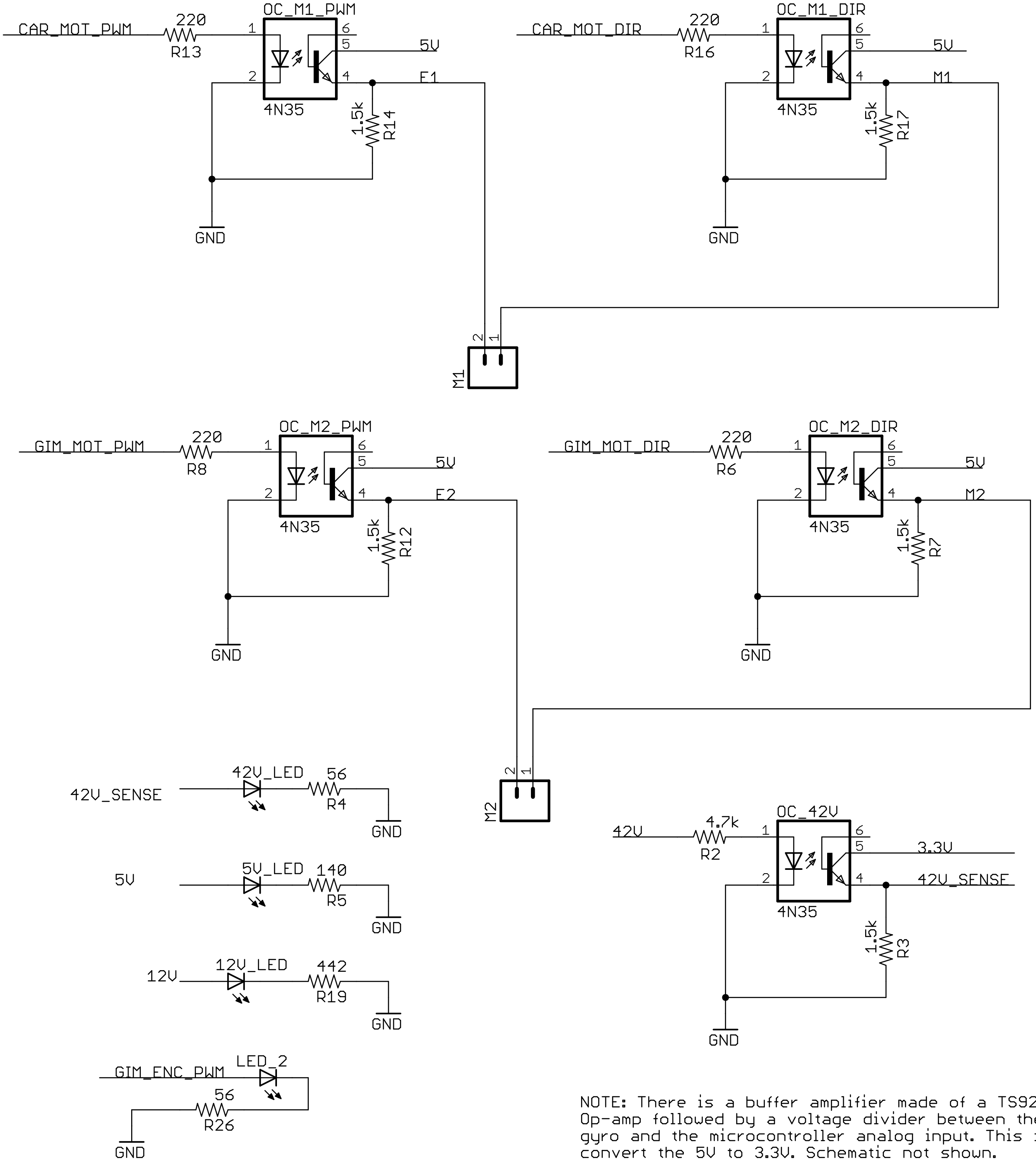

3.3V
3.3V_LED
56
R20
GND
R11
4.7k
RESET
3.3V
U1
GND
RESET
AREF
VCC
VCC2
GND
GND2
GND3
GND4
AVCC
AVCC2
XTAL1
XTAL2
UVCC
D-
D+
UGND
UCAP
VBUS
(SS/PCINT0)PB0
(PCINT1/SCLK)PB1
(PDI/PCINT2/MOSI)PB2
(PDO/PCINT3/MISO)PB3
(PCINT4/ADC11)PB4
(PCINT5/OC1A/OC4B/ADC12)PB5
(PCINT6/OC1B/OC4B/ADC13)PB6
(PCINT7/OC0A/OC1C/RTS)PB7
(OC3A/OC4A)PC6
(ICP3/CLK0/OC4A)PC7
(OSC0B/SCL/INT0)PD0
(SDA/INT1)PD1
(RXD1/INT2)PD2
(TXD1/INT3)PD3
(ICP1/ADC8)PD4
(XCK1/CTS)PD5
(T1/OC4D/ADC9)PD6
(T0/OC4D/ADC10)PD7
(HWB)PE2
(INT6/AIN0)PE6
(ADC0)PF0
(ADC1)PF1
(ADC4/TCK)PF4
(ADC5/TMS)PF5
(ADC6/TDO)PF6
(ADC7/TDI)PF7
ATMEGA32U4-AU
PIC_B0
PIC_B1
PIC_B2
PIC_B3
PIC_B4
PIC_B5
PIC_B6
PIC_B7
GYR_MOT_PWM
CAR_MOT_PWM
CAR_MOT_DIR
GYRO_SPEED_SENSE
42V_SENSE
ACK_AVR
GIM_ENC_PWM
GIM_MOT_DIR
GYRO_RATE
GIM_MOT_PWM
PIC_ENC_NUM_B1
PIC_ENC_HIGH
PIC_ENC_NUM_B0
ACK_PIC
JTAG
TCK
TDO
TMS
TDI
3.3V
GND
M05X2SHD
100nF
C5
C8
C7
C6
C4 20pF
Q1
16MHz
GND
C3
20pF
R9
22
R10
22
USB
D+
D-
VBUS
GND
5V
C1
10uF
C2
1uF
GND


5V
IN
OUT
ADJ
3.3V
IC1
C9
10u 25v
C10
10u 25v
GND


42V+
42V-
12V
1
2
GND
F1
F2
42V
D4
12V


3.3V
PROTO_CONN1
GND
QEI_1_B
GIM_ENC_PWM
QEI_2_B
QEI_3_B
PHASE_B
PHASE_A
1
2
3
4
5
6
7
8
9
10
20
19
18
17
16
15
14
13
12
11
QEI_1_A
GYRO_SPEED_SENSE
QEI_2_A
QEI_3_A
PHASE_C
GND
GYRO_MOT_V+
GND
CONN_TAPS
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
M08X2


+4
+1
+2
+3
5V
12V
42V
GND
BLDC ESC
+ - C B A
12V
R24
2k
5V
1
2
3
PHASE_A
PHASE_B
PHASE_C
GND
JP7
1
2
12V
GYRO_MOT_V+
Q2
IRLML5203
D2
Q3
IRLML5203
GND
GYR_MOT_PWM
220
R15
OC_GYR_PWM
1
2
6
5
4
4N35
3
2
1
2k
R18
N$21
SERVO_CONN
1
2
3
GND



NOTE: There is a buffer amplifier made of a TS922 Op-amp followed by a voltage divider between the gyro and the microcontroller analog input. This is to convert the 5V to 3.3V. Schematic not shown.