Department of Mechanical Engineering
University of Washington

Stevens Way, Box 352600
Seattle, WA 98195 USA
206-543-5090

Reaction Control of Inverted Pendulum

Matt Lester
sargent7@uw.edu

John Davis
jdavis7@uw.edu

Daehyun Kim
dhk0927@uw.edu

Yongsuhn Jo
joy01@uw.edu

Jonathan Keogh
jkeogh@uw.edu

Mechatronics ME495 Design Report —June 10th, 2016

Executive Summary

The inverted pendulum is a classical feedback control problem for an inherently unstable system. This problem is similar to balancing a pen on a finger. It is usually accomplished by mounting pendulum on a moving cart. However, for our capstone project, we decided to stabilize the system with a reaction wheel affixed toward the top of the pendulum. We wanted our system to be able to stabilize from +/- 3 degrees offset with settling time of 1 second and percent overshoot of no more than 25%. To satisfy the functional specifications, it was necessary to implement a phase-lead controller. We also implemented proportional gain controller to produce a voltage opposite in sign to the output of the lead compensator to remove some residual velocity after transient responses. Most of the mechanical components of our system were to be machined from aluminum except for the encoder bracket and the hanger bracket which were 3D printed. Holes were drilled in the walls of a C-channel, which was used as the pendulum rod, with 2.5 cm spacing. This allowed for the motor/flywheel assembly to be relocated up and down the pendulum and compare the response. Along with the motor, tachometer, and encoder, all of the mechanical components were assembled properly. In addition to the mechanical assembly of our system, there was an electrical component as well. We wired the motor, tachometer, and encoder to the current amplifier via the digital to analog converter (DAC) socket, low-pass filter via the analog to digital (ADC) socket, and encoder socket on the Quanser board, respectively. To achieve communication between the mechanical and electrical components, we wrote a C language script for myRIO using the Eclipse development environment. In the end, our inverted pendulum was able to balance itself with a reaction wheel and met our functional requirements.

Introduction

The inverted pendulum system is a standard problem in the area of control systems. Early investigations of the inverted pendulum were geared toward the need to design controllers to balance rockets during vertical take-off [2]. Similar to rocket launch, which is exceptionally unstable, the inverted pendulum needs a controller that continuously manipulates the top to stay upright. Even though widespread approach to the control problem utilizes a moving cart along a horizontal track, our team was tasked with applying inertial reaction forces to stabilize an inverted pendulum rotating about one axis at its base. The system should maintain stability in the steady state as well as when subjected to external disturbances, either through an initial position, initial velocity, or applied disturbance torque.

Functional Requirements

In order to constrain the design for this project, we prescribed the following functional requirements for our system response: –Settling time of 1 seconds.
–Stabilize initial displacements of +/- 3 degrees
–Stabilize initial velocity of 0.50 rad/s
–Steady-state error of 0 degrees for step input of either initial displacement or velocity
–Percent overshoot of 25%
–Realistic estimates for pendulum dimensions were also made to guide the design development process. The pendulum was assumed to be approximately 0.5 m in length with an assumed point mass equivalence of 1 kg.

Approach to the Solution

Given the task of stabilizing an inverted pendulum through inertial reaction forces, several possibilities for control presented themselves. These options included reaction wheels, linear actuators, and thrusters. Reaction wheels and linear actuators appeared to be the most robust possibilities. They also presented clear design possibilities.

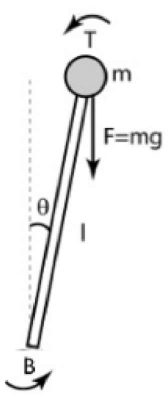
Linear reaction forces presented a unique problem. The application of a reaction force would require a mass at the top of the pendulum as a reactive element, which would move in a linear motion at the top of the pendulum. This would create the problem of a constantly changing center of gravity, which would complicate the system dynamics. Therefore, it was determined that using a rotational reaction control system would be superior.

Before beginning design, we researched existing technology. It was found that rotational reaction control was not a well-documented method of pendulum stabilization. However, reaction wheels are commonly used in space applications such as to control the orientation of satellites. Reaction wheels allow this repositioning without the need for propellants.

Design

Analysis and Model

In order to begin analyzing the problem, a simple dynamic model was required. A free-hanging pendulum with point mass at the system's center of mass was used to create a linearized dynamic model. Zero damping was assumed for both bearing friction and drag. The simple dynamic model shown in Figure 1 was created.



$$\sum T = I\ddot{\theta} = mgl\sin\theta + T_s$$

$$I = ml^2 \Rightarrow ml^2\ddot{\theta} = mgl\sin\theta + T_s$$

$$\sin\theta \approx \theta \Rightarrow ml^2\ddot{\theta} - mgl\theta = T_s$$

$$\Rightarrow \theta(ml^2s^2 - mgl) = T_s$$

$$\frac{\theta(s)}{T(s)} = \frac{1}{ml^2s^2 - mgl}$$

Figure 1: Free body diagram for derivation of equations of motion.

To derive the equations of motion above, an appropriate linearization was required. In particular, we approximated gravitational torque applied to the pendulum with the small angle approximation. The limits of this linearization were analyzed and it was determined that there would be insignificant deviation with the actual nonlinear model until approximately +/- 20 degrees of offset. This meant the use of a linearized model would be more than adequate in developing the control system.

Due to the inherent instability of the inverted pendulum, it was necessary to implement a control law. Examining the root locus of the inverted pendulum Figure 2), it was apparent that the use of a proportional gain controller would be insufficient. A phase-lead controller was investigated instead (Figure 3). The initial block diagram for the system was modeled and the angular position of the pendulum was used as feedback (Figure 4). From this, an initial phase-lead controller was derived by hand calculations, and simulation of this model exhibited stability. This validated the decision to utilize a phase-lead controller. The controller's pole, zero, and gain were chosen based on our percent overshoot and settling time requirements. The computation of controller parameters was automated in a MATLAB script, and made easily adaptable to different pendulum heights, masses, etc.

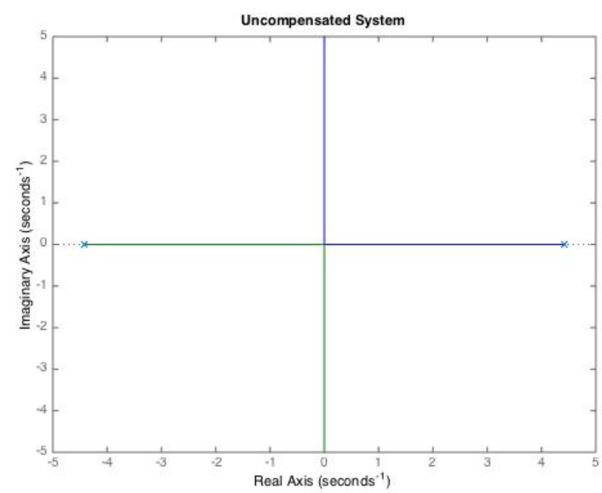


Figure 2: Root-locus of uncompensated system.

Simulation Results

Simulink modeling was used for system stability analysis. It allowed us to easily augment the system model to create a more realistic simulation. Starting with a model of the simplified pendulum and phase-lead controller, an initial condition of 0.05 radian angular position offset (approximately 3 degrees) was set. Torque request and other parameters were simulated (Figure 5). As found through the simulation, the motor torque request did not exceed

1 Nm. This is within a realistic realm for electric motors of practical size and price. The block diagram of the simulated system, including the main feedback loop, can be seen in Figure 6.

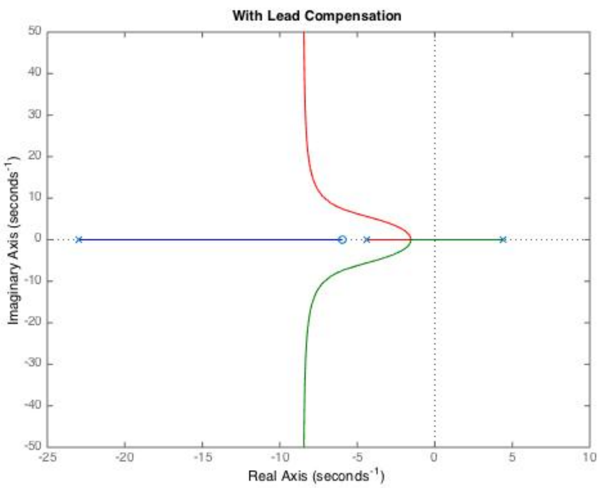


Figure 3: Root-locus of system with lead compensation.

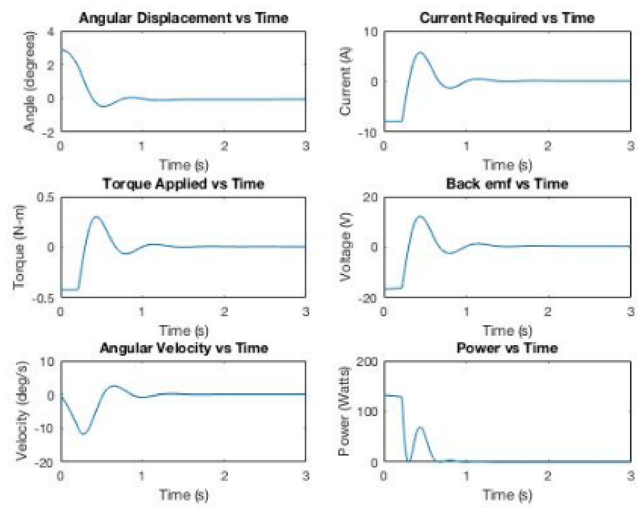


Figure 5: Simulation results with 3 degree initial offset.

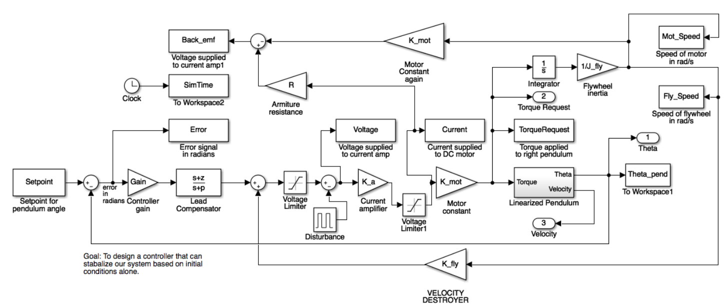


Figure 6: Main feedback loop of system.

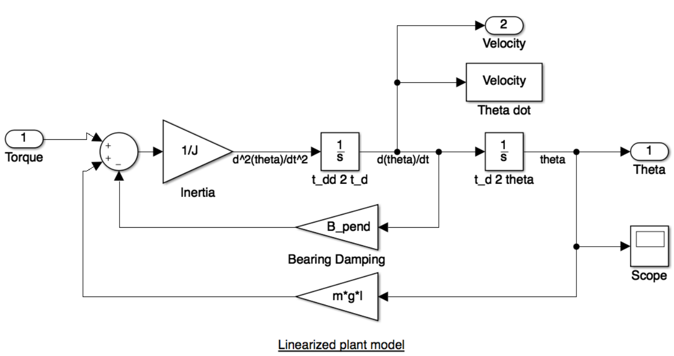


Figure 4: Linearized pendulum model subsystem.

Research was conducted on electric motors suitable for our application. One promising motor, the Maxon RE35 273754, had a motor constant of approximately 0.0525 Nm/A. Our simulation indicated that for this motor, 8 amps would be sufficient to produce the required torque, and fell well within the physical limitations of our system.

The addition of a simple current amplifier source assuming a power supply gain of 0.4 V/A validated the progress of the design to this point, requesting 20 volts which was also well within the limits of realism for this system. For robustness, our final system had a 48 volt power supply.

The model to this point was based on feeding back angular position to the controller. However, this was found to be potentially problematic in situations where the reference surface is not level or in situations where the reference surface is moving. So it was attempted to instead feedback angular velocity to the model which could then be integrated to determine position if necessary. However, it was found that by feeding back angular velocity, a zero at the origin was added to the model's root locus, thereby making a phase-lead controller inherently unstable. In fact, no controller design could fix this without relying on integrating the angular velocity to give angular position which negated any possible benefit of feeding back angular velocity. Thus, it was decided that feeding back angular position would be the better option. We also planned to address the unlevel table problem through the use of an accelerometer/gyroscopic force sensor but later found that it wasn't necessary because it is reasonable to assume that the table will not be moved once the program is initiated.

Diving deeper into the limitations of the electric motor for the design, diminished torque potential of the motor due to back EMF was considered. As the rotational velocity of the motor increases, the maximum torque it can output decreases in a linear fashion until reaching zero torque at the no-load speed of the motor. The maximum torque of the motor likewise occurs at zero rotational velocity and is therefore called the stall torque of the motor. Due to this behavior and the sensitivity of the system to the torque output of the motor, it was most desirable to keep the speed of the motor as close to zero RPM as possible. Therefore, the reaction wheel mechanism of the design was analyzed in more depth. To verify that our system would not demand current or voltages above our limitations, we plotted the torque-speed response to an ambitious initial condition of 10 degrees offset. We overlaid the motor's torque-speed limit line operating at 48 volts (Figure 7).

The effects of back EMF lead to another design consideration - managing the steady-state flywheel velocity. Since steady-state velocity exerts no torque on the system and does not affect the pendulum's stability, there is always some residual velocity after transient responses. It is easy to imagine that this velocity could accumulate and diminish the motor's torque potential, and even saturate at no-load speed. To solve this problem, we used a tachometer to measure motor velocity, and a proportional gain controller to restore steady-state velocity to zero. By deriving the transfer function relating controller voltage to flywheel velocity, it was found that the flywheel velocity time constant is:

Kfly can then be chosen by specifying a time constant tau. Since the velocity controller may well produce a voltage opposite in sign to our lead compensator, the controllers "fight" each other. Small values of tau produce more aggressive velocity controllers than interfere with the lead compensator. Large values of tau have slow response time. The optimal tau was found experimentally by trial and error. Velocity reduction is also discussed by Block et al. [1]. Figure 8 demonstrates the effect of velocity control during an initial angular offset.

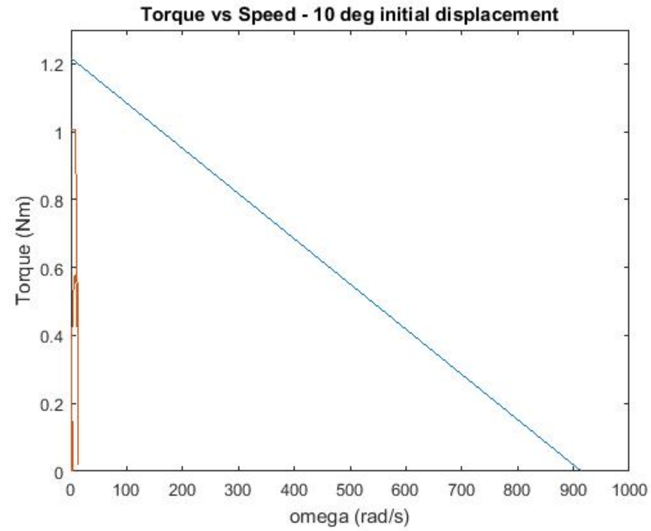


Figure 7: Torque vs. Speed curve for 48 volt power supply.

$$\tau = \frac{J_{fly}}{K_{fly} K_a K_{mot}} \tag{1}$$

Since the flywheel is merely a reactive body in the mechanics of this system, the relationship between the flywheel's inertia and experienced torque is T= J*alpha (alpha=angular acceleration). This can be restated as T/J = alpha. From this, it was evident that if the inertia of the flywheel was increased, the resulting angular acceleration on the flywheel by the torque applied to it by the electric motor would decrease. Therefore, over the same period of time with the same torque applied, a flywheel with greater inertia would have a lower final rotational velocity (Figure 9). This meant that the motor could be kept at a lower overall speed by increasing the inertia of the flywheel. This was found to aid in keeping the electric motor in the regime of maximum torque potential. However, practically speaking, large flywheel inertia requires greater mass. The tradeoff of large flywheel inertia is that the inertia of the whole pendulum, and its susceptibility to disturbances, increases with mass. It is therefore necessary to find an optimal balance of large inertia and low mass.

Mechanical Design

Our model was derived around an inverted pendulum with only one degree of freedom: rotation about the pen-

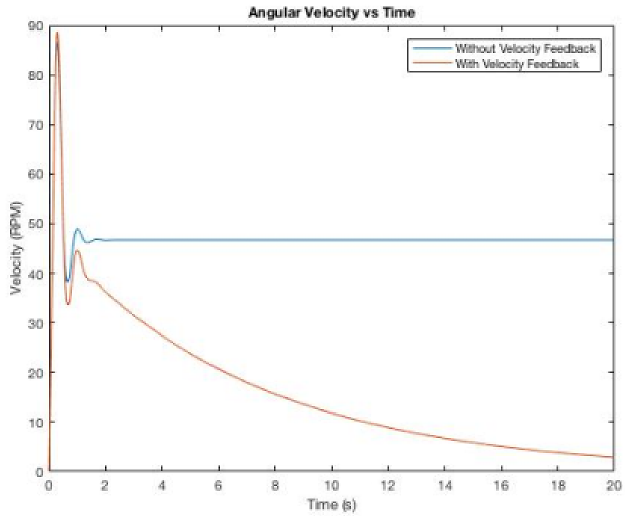


Figure 8: Flywheel speed with and without inclusion of velocity destroyer

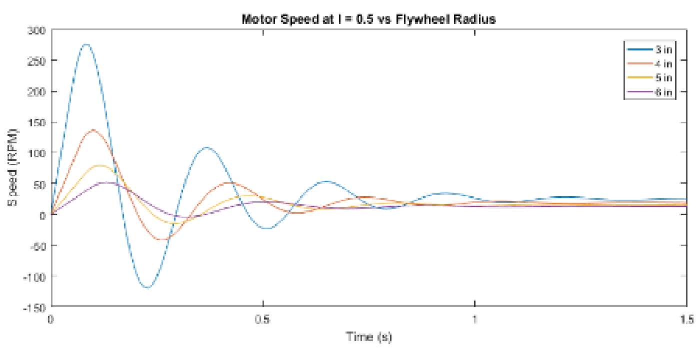


Figure 9: Flywheel speed with varyng flywheel diameter.

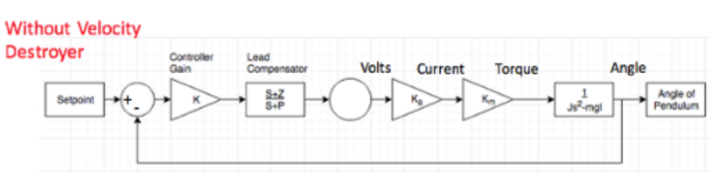


Figure 10: Block diagram without inclusion of velocity destroyer.

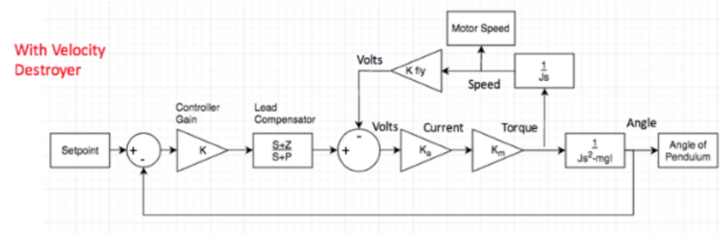


Figure 11: Block diagram with inclusion of velocity destroyer

dulum's desired pivot point. Also required was a way to measure the angle of the pendulum with high angular resolution. In order to meet both of these requirements, it was decided to use a hanger bracket attached to a baseplate on which the pendulum can be mounted. The other end of the hanger bracket shaft was attached to the encoder via a flexible connection in order to prevent damage to the encoder. Medical tubing was well suited for this due to the low inertia of the encoder and therefore minimal effect of any torque exerted upon it, preventing deviation from the analytical model.

The pendulum itself was made using 1"x1" C-channel with 1/8" wall thickness. This would allow for a channel in which the electrical wiring could be run in order to contain the wiring to prevent it from becoming damaged during operation, as any inverted pendulum design problem necessarily involves significant physical movement.

The reaction torque itself was to be provided by an electric motor. Therefore a slide was designed that would mount onto the C-channel rod from which the motor could be mounted. This also had the effect of further constraining the movement of the wiring leading to the motor by the C-channel and slide creating a contained channel for said wiring. In order to power the motor, a Copley 412 current amplifier was utilized, giving a maximum voltage of 48 volts and a maximum current of 10 amps. With our motor's torque constant of 52.5 mNm/A and armature resistance of 2.07 ohms, this would give a maximum reaction torque of 525 mNm which our design charts suggested would be sufficient to stabilize an inverted pendulum with a distance of 0.5m between the motor/flywheel and the pendulum axis of rotation.

Due to the reaction torque coming from the output torque of the electric motor, an inertial load is highly valuable as it increases the amount of time the electric motor can exert a torque before it reaches its no-load speed, therefore allowing the inverted pendulum to be more robust. This inertial load was chosen to come in the form of a flywheel. However, as precision flywheels were not readily available for purchase, it was decided that we would manufacture our own flywheel. Therefore, aluminum was chosen as the flywheel material due to its ease of machinability. A material with a higher density would be better suited which would suggest steel rather than aluminum while having approximately the same final cost, however the increased difficulty of machining steel led to our discounting its use.

Our design charts indicated that if using a spoked ring flywheel design, the larger the diameter the better as the gain in flywheel inertia offset the gain in motor/flywheel mass (and therefore pendulum inertia). Therefore the limiting factor in regards to flywheel diameter was that the mills available to machine the flywheel only had a maximum X travel of +- 30.5 cm, therefore when factoring in tool diameter constrained any possible flywheel we could safely machine to 28 cm in diameter. However with a no-load speed of over 8000 RPM, it was decided that 15-20 cm was the maximum diameter flywheel we would machine to maintain the safety of our design.

We desired including the ability to adjust the distance between the pendulum's axis of rotation and the motor/flywheel combination because this would greatly affect the inertia of the pendulum and therefore the robustness of the system as once the motor and current amplifier were chosen the maximum reaction torque was constrained. Affixed to the motor would be a tachometer in order to track flywheel angular velocity so it could be used within our second control system that would reduce any residual velocity left in the flywheel after reacting to disturbances.

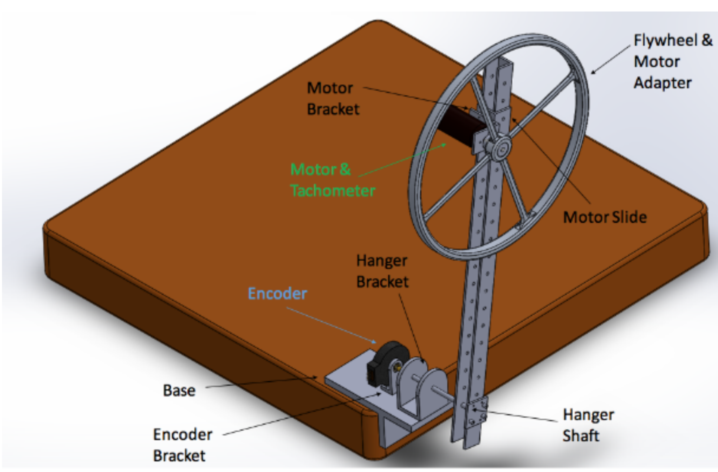


Figure 12: Final assembly of mechanical design.

Electrical Design

In addition to the mechanical design of our model, we needed to wire the motor, tachometer, and encoder properly. The Quanser board has an encoder connector socket, analog-to-digital converter (ADC) socket, and digital-to-analog converter (DAC) socket. National Instruments myRIO was used to connect the Quanser board on the connector C. With the pin-out for our optical encoder, DIN connector was used to connect the encoder to the encoder connector socket on the Quanser board. The motor was connected to the current amplifier, which has a 48V power supply, and to DAC socket. A low-pass filter was used to filter out the noise from the tachometer. Then, the tachometer was connected to ADC socket. The schematic of our electrical system is shown in Figure 13.

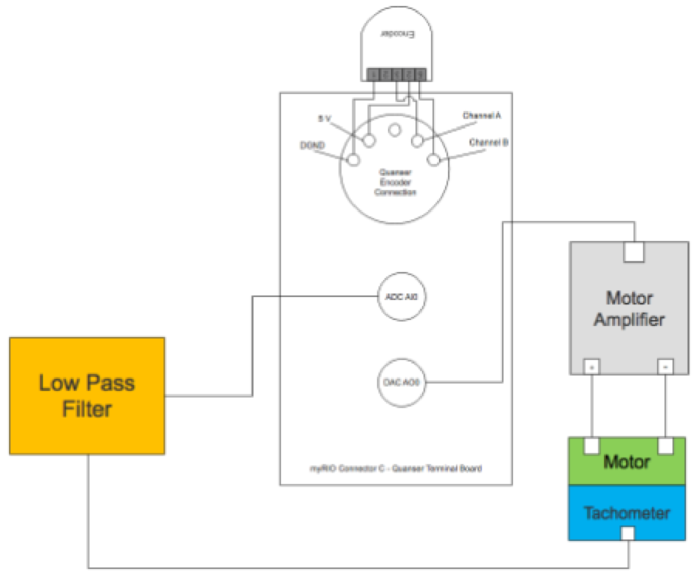


Figure 13: Schematic of electrical system.

Computational Design

To achieve communication between the mechanical and electrical components, we wrote a C language script for myRIO using the Eclipse development environment. First, the code imports a header file from MATLAB that contains the precalculated zero, pole, gain of our compensator, flywheel inertia, number of sections, time interval, and biquad filter. Our code, at a high level, is composed of three threads: main, timer IRQ, and table update. In the main thread, it first reads the tachometer voltage offset value. Since our tachometer inaccurately showed a non-zero voltage even if the flywheel was not spinning at all, it was necessary to read this offset value in order to calculate the correct velocity of the motor later in the timer thread. Then, the code calculates the moving average of encoder counts when the pendulum is at a vertically down position for 5 seconds. This step helps to find the approximate center of gravity of our system. Two thousand encoder pulses (180 degrees) are then subtracted to find our "zero" or upright position. Afterward, we manually bring the pendulum upright and press "Delete" on the keypad to start the other two threads. At this point, the reaction wheel control system starts running to balance the pendulum.

In the timer IRQ thread, we defined the table entries for the ctable on the LCD display. We implemented five variables in the table. The input variables that we could alter during the control process were the reference angle and the flywheel constant. The reference angle is the angle we tried to balance the pendulum at, and the flywheel constant is a constant needed to achieve the steady-state flywheel velocity. The output variables that demonstrated the current state of the system were the current velocity of the flywheel, the current angle of the pendulum, and the output voltage of the system. After setting up the table variables, the code determines the tachometer constant that is needed to calculate the velocity of the motor as well as the flywheel constant, or "velocity destroyer" constant. The code then measures the current angle of the pendulum with the encoder. To do this, it calculates the difference between the reference and the current encoder positions and multiplied by 360 degrees over 4000 counts because our encoder has 4000 counts per one revolution. Then our compensator and current angle determined the controller voltage using the cascade function. The current velocity of the motor and the flywheel constant determined the "velocity destroyer" voltage. The addition of the two terms were used to send the computed control value to DAC. Additionally, the timer thread implements critical safety checks that limit the output voltage, angular displacement, and the velocity of the flywheel to a safe value. These safety limits will be further discussed on the Risk and Liability section. At the end of the timer thread, it also continues to change the reference angle for a better "zero" position and its purpose will be discussed later on the Integration section.

Meanwhile, the table update thread keeps updating the table entries in the LCD screen according to the system status and the user input. The pseudocode of the computational design described above is attached in the software description section of the Appendix. Furthermore, the system hierarchy in Figure 14 summarizes the relationships among the three threads and their functions.

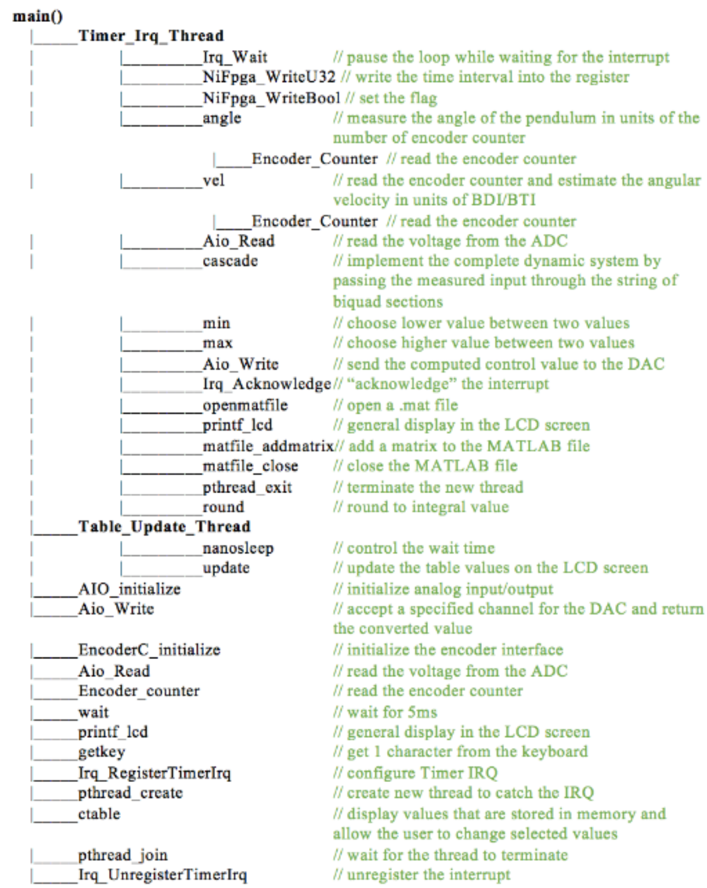


Figure 14: Hierarchical diagram.

Hardware

Most of the mechanical components of our system were to be machined from aluminum with the encoder bracket and the hanger bracket being the exceptions. Holes were drilled in the walls of the C-channel with 2.5 cm spacing. This allowed the distance between the motor/flywheel and the pendulum axis of rotation to be modified in 2.5

cm increments.

The pendulum base, the motor slide, and the motor bracket were machined from 7050 billet aluminum. The pendulum shaft adapter and flywheel spacer were machined from 6061 aluminum round card (1" diameter). The tolerances for all of these components were a non-issue due to the insignificant stresses these components would undergo during operation.

The most difficult component of our assembly to machine was the flywheel which, due to its intended operating speed of up to 1,000 RPM or higher, required extremely tight tolerances to minimize any geometric asymmetries and therefore remain balanced while rotating. The flywheel was CNC milled from 5/8" thick 7075 aluminum plate. The necessary G-code was created using HSMworks to program the milling operations. Initially, a 20 cm diameter flywheel was machined. However, the flywheel spacer was designed to be shrink fit into the flywheel hub but during the shrink fitting process, the spacer heated too quickly and the operation failed. The subsequent attempt to press the spacer the rest of the distance resulted in deformation of the spokes as they had been designed to be thin to maximize the inertia/mass ratio of the flywheel.

Turning was attempted to remove the port spacer and attempt another shrink fit using an interference of 0.0015" inches. However, this operation also failed. Our final flywheel was redesigned to have only three spokes, each of which was much more robust than the first flywheel, and having a diameter of only 15 cm due to limited remaining aluminum plate. The flywheel spacer was press fit on this attempt and the results were satisfactory. The flywheel remain sufficiently balanced even up to the no-load speed of the motor, however the speed of the flywheel would be limited in software to maintain the safety of our design.

The encoder bracket was 3D printed as the only constraints were to rigidly maintain the clocking of the encoder with respect to the pendulum and fix the encoder shaft to a position approximately concentric with the hanger bracket shaft. Due to the low friction between the encoder wheel and the encoder housing, minimal torque was required to fix the clocking of the encoder and therefore the use of a 3D printed bracket was sufficient.

As stated in the mechanical design section, the current amplifier used was a Copley 412 DC motor driver which was capable of a maximum output voltage of 48 volts and a maximum current output of 10 amps. However, the amplifier gain was reduced from 1.0 to 0.40 for debugging to reduce the maximum stresses experienced by the components of the pendulum system.

Integration

After machining and 3D printing mechanical components, the mechanical assembly came together fairly easily. Coupling of the fulcrum shaft with the encoder was accomplished using surgical tubing to allow for some leeway in the tolerance of the fit. The flywheel posed the greatest machining challenge. Due to warpage during a shrink fit, the first flywheel was improperly balanced and had to be adjusted considerably by re-facing surfaces on the lathe. A second flywheel was machined that was much better balanced. Several electrical and software challenges were encountered as well. The tachometer output voltage was noisy and contained a non-zero DC offset. To remedy this, we used an analog low-pass filter and subtracted the DC offset term in our program's calculations. Another challenge was finding the perfect flywheel constant that is needed to determine the "velocity destroyer" voltage. Our flywheel constant, as previously discussed, was determined by dividing the flywheel inertia by the product of amplifier gain, motor constant, and time constant. Since the flywheel inertia, amplifier gain, and motor constant were all fixed values, we tested different time constants to make the flywheel properly "kill" its velocity at the vertical position. 15 seconds was determined to be the best time constant. The last significant challenge we faced was inaccuracy in the "zero" position measured. This lead to the flywheel running at 500-700 rpm while balancing at the position slightly off the real "zero". Ideally the pendulum should balance at the zero position with the flywheel switching directions back and forth at a slow speed of around 50 rpm. This was not initially achieved perhaps due to the low encoder resolution or the imbalanced flywheel constantly altering the center of gravity. To resolve this issue, we manually adjusted the reference angle by a tiny step in the timer thread whenever the flywheel speed was above the cut-off velocity with the pendulum slightly off the "zero".

Prototype

Purpose

The purpose of building this prototype was to tie together the engineering fundamentals we have learned throughout our academic careers and to apply a control algorithm to an inherently unstable system. Additionally, the prototype was built to assess our ability to operate our device under engineering control (i.e. adjust system parameters and anticipate output behavior).

Testing and Results

Our main basis for assessing our prototype was comparing the experimental response to the simulated response.

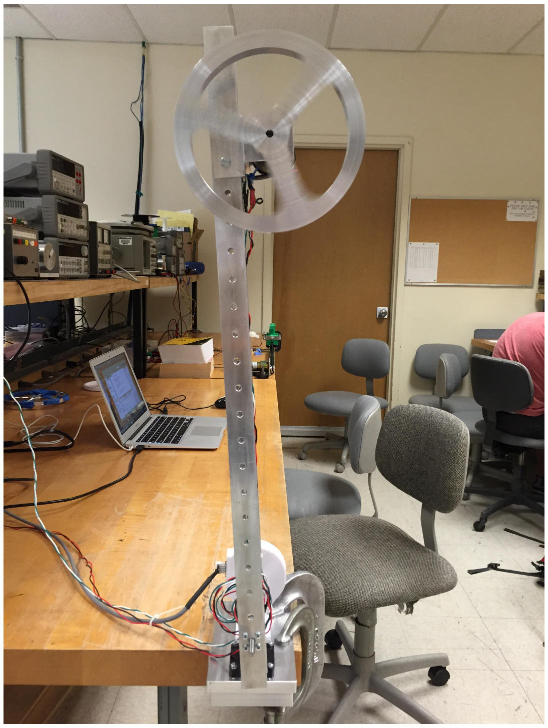


Figure 15: Final assembly of prototype.

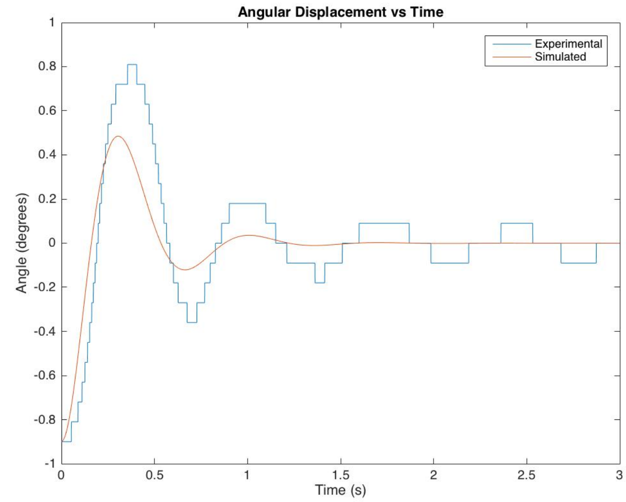


Figure 16: Angular displacement of pendulum vs. time shown for initial displacement of 1 degree.

Below are two figures comparing the measured and experimental angular displacement vs. time and flywheel velocity vs. time (Figures 16 and 17). Furthermore, our design was driven by our functional requirements laid out in the design section. We designed the system to have a settling time of 1 second, and a percent overshoot of 25%. In reality, our percent overshoot was higher than expected. This is likely due to the simplified model used to design our control loop which neglected damping at the pivot point and modeled the pendulum as a point mass on a massless rod. Furthermore, our Simulink model didn't take motor dynamics into account. In reality, there is likely an inductance in the motor which can slow response, and a back-emf problem that wasn't fully represented in the model. In spite of this, our requirement for a 1 second settling time was achieved and we are satisfied with the results. The step-like oscillation seen in Figure 16 is due to the resolution limits of our encoder.

Response to disturbances was also assessed. Just like for the initial displacement response, settling time after a disturbance matched closely with our model. Percent overshoot was larger than expected, particularly on the down-swing. The steady-state angle offset immediately after the disturbance in Figure 18 is due to the "modify zero" algorithm, as discussed in the integration section, which moves the zero reference position if the angular speed of the flywheel gets too high when the controller thinks the pendulum is at zero reference. Because fly-

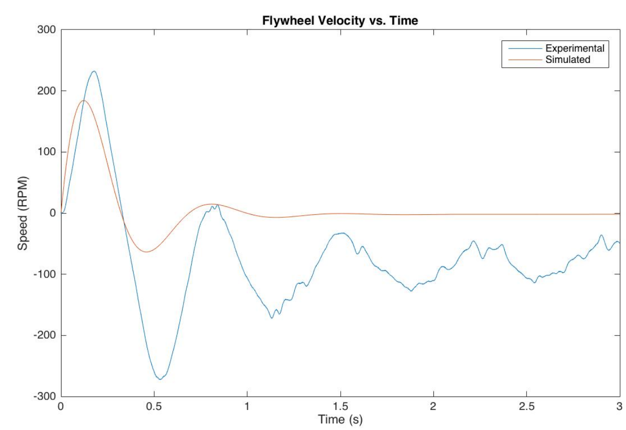


Figure 17: Flywheel velocity vs. time for an initial offset of 1 degree.

wheel velocity exceeds 200 rpm, our control loop modifies the zero setpoint and attempts to stabilize there.

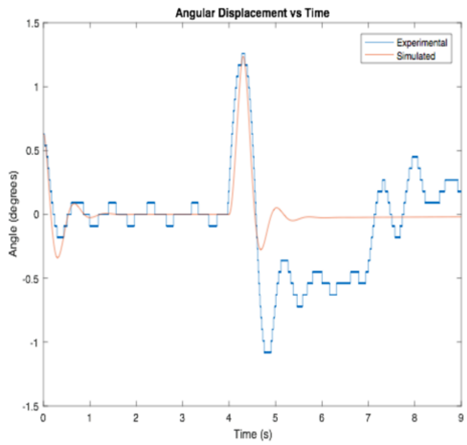


Figure 18: Angular displacement of pendulum vs. time shown for response to disturbance.

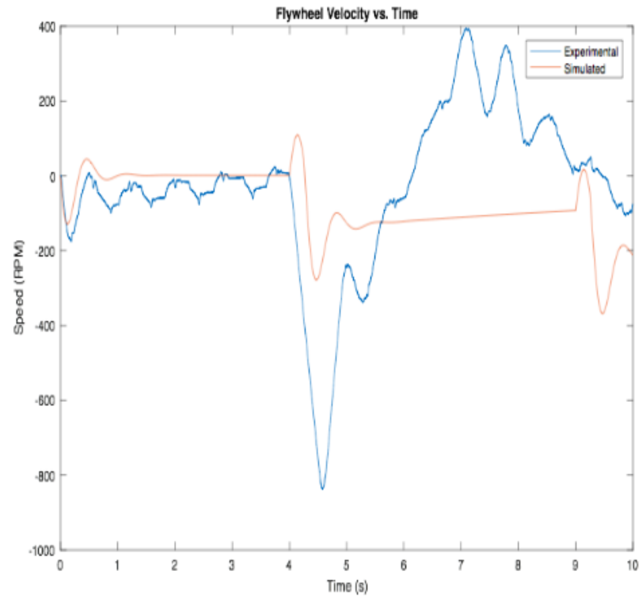


Figure 19: Flywheel velocity vs. time for response to disturbance.

Risk and Liability

Our prototype posed a small risk of injury to people standing nearby. Due to the tolerance we were able to achieve from the tools used to manufacture our prototype, small imbalances were present in our system. These imbalances lead to small vibrations within the pendulum when operated at high RPM. If these vibrations became too severe, the flywheel could potentially detach from the pendulum and become a projectile. To mitigate this, a limit on speed of 1000 RPM was initially imposed via software. If the flywheel speed exceeded 1000 RPM, the output voltage to the motor would be set to zero until the speed reduced. Through fine tuning and remanufacturing of the flywheel, we were able to get to system to be more balanced and were able to operate at higher speeds without experiencing vibrations. We still have a limit on speed in the code of 1500 RPM to ensure that the flywheel doesn't speed up without limit. To maintain stability, our system shouldn't need much more speed than 1500 RPM anyways.

Another thing one must consider when working with an inherently unstable system such as an inverted pendulum is what happens when it falls over. Since the stability of our system is dependent on our flywheel applying a torque to maintain a vertical position, it needs to be able to accelerate whenever a non-vertical position is detected. However, if the pendulum tips beyond an angle that it can recover from, it will continue to try to accelerate forever and certainly pose risk of injury. We mitigated this by setting a hard limit on angular displacement in code. If the pendulum tips beyond 90 degrees in either direction, the voltage supply to the motor turns off and the program stops running. This ensures that the motor velocity will be cut off and safely return to zero.

Ethical Issues

Inverted pendulum technology is essential in the design of Segways and the recently popularized "Hoverboards." It is well documented that labor rights and human health issues are consequential in the production and disposal of consumer electronics. Labor conditions in precious metal mining operations are often very poor. The metal mining industry exploits inexpensive labor and abundant resources in areas of Africa, Asia, and South America. In addition, post-consumer electronic waste is frequently sold to landfills and recycling stations in Asia and Africa. Here, materials are salvaged under scarce regulation and oversight. This creates hazardous human health conditions, and contributes to a poverty cycle perpetuated by global trade and production. To mitigate these ethical issues, devices should be designed to minimize the use of precious metals. Careful consideration should also be given to the supply chain.

Impact on the Environment

In addition to labor issues, consumer electronics are well known to cause environmental damage through their production and disposal. Electronic components like batteries and lead solder can leach harmful chemicals into the soil and groundwater near their disposal site. They can also release toxins into the air if they are burned in waste piles. To minimize risk to public and environmental health, devices should be designed to be power efficient. Consideration should also be given to battery selection and alternative power sources.

Impact on Society

The applications of inverted pendulums and reaction wheels have a number of impacts on society. Since Hoverboards are so compact, they increase personal mobility over short-range travel. This could improve efficiency in home and work life, where walking to meetings, errands, or lunch breaks cuts productive time out of the day. Segways are popular for guided tours at tourist destinations. This technology can bolster local economies by attracting tourists. Reaction wheels are most commonly used for satellite attitude control. Satellite research and technology enables scientists to advance human understanding of the universe.

Cost and Engineering Economics

Due to the relative simplicity of our reaction wheel stabilized inverted pendulum, the primary costs of the assembly are found within only four components: the motor used to exert the reaction torque, the flywheel used to react this torque, the encoder used to measure the angle of the pendulum, and the current amplifier used to drive the motor. As our prototype was of limited size, the selection of motors with sufficient manufacturer documentation was limited. As a result, we utilized a Maxon DC motor with a maximum torque output of 525 mN-m. The specific motor we used is now discontinued however equivalent motors from Maxon cost approximately $600. However, much cheaper motors could be used so long as their lack of documentation can be overcome through controller design and experimental data.

While the encoder used was capable of an angular resolution of only 0.09 degrees, this was found to be a limiting factor in the performance of our final design. Therefore, either a higher resolution encoder or a gearing system would be required to improve angular resolution, both of which increase the costs of the system. If our design were to be applied to larger scale problems, the angular resolution would become even more important as absolute displacement of the tip of the stabilized pendulum increases linearly with increase in length of the pendulum.

High performance current amplifiers are widely and readily available, with economies of scale applying at increased power capacities. Therefore the most cost sensitive component of our design is the flywheel itself. Due to the very tight tolerances required, the primary cost of the flywheel is in the manufacturing of it; the raw material costs are comparatively insignificant. However, the manufacture of the flywheel could be optimized for both larger scales and increased production rates, possibly using water-jetting to quickly and affordably produce round blanks that can then be finished using CNC milling, since the majority of machining time was due to material removal.

Codes and Standards

Due to the relatively simple and safe design of our reaction wheel stabilized inverted pendulum, and applicable codes and standards would concern the electric motor, encoder, current amplifier, and/or tachometer. These should be sufficiently met by the manufacturer of said component and therefore not of concern to us.

However, due to the operating speeds of the flywheel, depending on the scale and application of the reaction wheel stabilized inverted pendulum, there may be applicable codes and standards with regards to tolerance of the flywheel at intended operating speeds or vibrations resulting from slight geometric asymmetries of the flywheel during operating since the pendulum will be subjected to these vibrations. If this design is used at a high enough speed or at a large enough scale for these to be applicable, the appropriate codes and standards should be consulted to ensure compliance.

Conclusions

Continued Development

If we were to keep working on this project, the first thing we would implement would be a swing up routine in which the pendulum rights itself prior balancing in the vertical position. This has been shown to be possible by previous inverted pendulum designs. We began coding this routine, but ran out of time before the quarter ended. Another stretch goal we set but were unable to complete was the implementation of a UM7 sensor for sensing angular position of the pendulum. The UM7 chip contains an accelerometer and gyroscope and communicates with the microcomputer via UART connection. Our plan was to get the system up and running using the encoder attached to the pendulum pivot and eventually implement the UM7, but we ran out of time.

Final Product Configuration

We are happy with the final product and if needed, would reconstruct the device in the same way. However, if we were able to source components again, we would select a higher resolution encoder to improve the accuracy of the zero (vertical) position, and look for a tachometer with a better transient response and less noise. Additionally, the motor exhibited a slight squeaking noise when rotating in the counterclockwise direction. This is indicative of damping being present in the motor and isn't ideal. However, the system was able to balance, which minimized the urgency of this issue.

Bibliography

References

[1] K.J. Block D.J. AÌŁstroÌĹm and Spong M.W. *Reaction Wheel Pendulum.* Morgan & Claypool., San Rafael, CA, 2007.

[2] K.K. Mathew N.J. Rao and Sivakumaran N. *Swing Up and Stabilization Control of a Rotary Inverted Pendulum.*, volume Preprints of the 10th IFAC International Symposium on Dynamics and Control of Process Systems. Mumbai, India., 2013.

Appendices

— Drawings

1. Mechanical

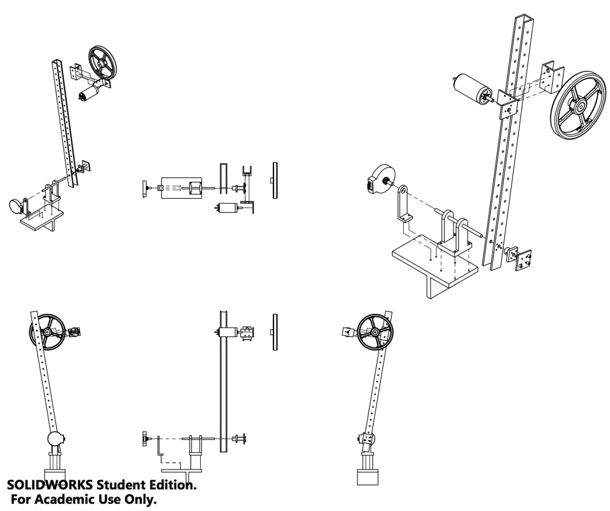


Figure 20: Inverted pendulum assembly with flywheel

2. Electrical (individual components and connections)

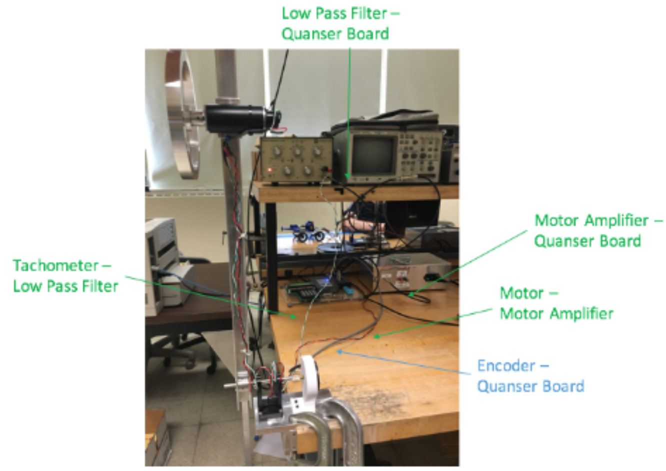


Figure 21: Electrical subsystem overview.

Figure 22: National Instruments myRIO shown with Quanser board.

Figure 23: Current Amplifier with 48 V Power Supply (left) and Low-Pass Filter (right).

3. Software Descriptions (flow charts, hierarchical diagrams, etc.)

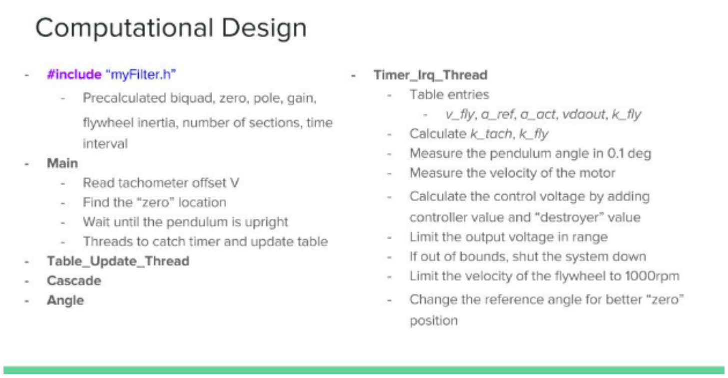


Figure 24: Pseudocode

— Code

1. MATLAB analysis and design
   –Submitted electronically to Prof. Garbini 6/10/16
2. C-Code
   –Submitted electronically to Prof. Garbini 6/10/16

— Major components list for prototype

1. Electric motor
   (a) Manufacturer Maxon Motor
   (b) Model number 273754
   (c) Cost Discontinued (equivalent motor costs $588.50)
2. Current amplifier
   (a) Manufacturer Copley Controls
   (b) Model number 412
   (c) Cost Highly variable (approximately $150+)
3. Encoder
   (a) Manufacturer US Digital
   (b) Model number S2 Optical Shaft Encoder
   (c) Cost $93.50
4. Tachometer
   (a) Manufacturer Maxon Motor
   (b) Model number DCT 22 118908
   (c) Cost $59.80
5. MyRIO
   (a) Manufacturer National Instruments
   (b) Model number NI MyRio
   (c) Cost $1,000