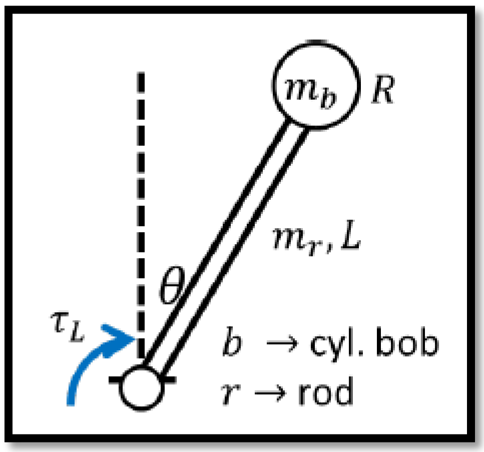


Final Project 4: Inverted Pendulum Controls Demonstration

Applied Mechatronics (MECH 5970-019)

April 29, 2022

Walter Livingston   Matthew Covington

wbl0009   mac0139

Table of Contents

Table of Figures

Team Photo

Figure 1: Team Photo

Project Photos

Figure 2: Front Panel 1

Figure 3: Front Panel 2

Figure 4: Back Panel

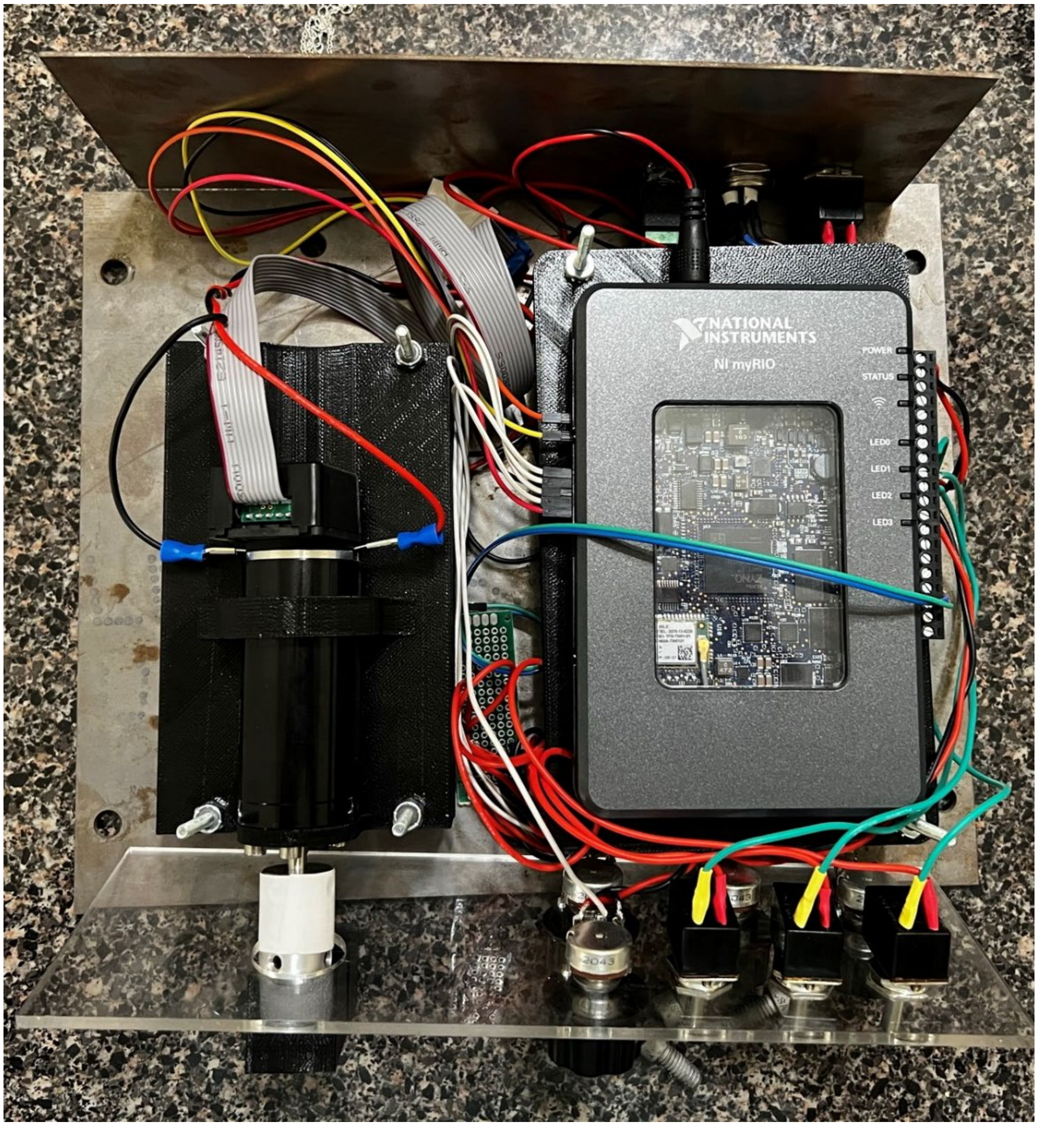


Figure 5: Internals

Project Demonstration Video

Functional Summary

The IPCD's (Inverted Pendulum Control's Demonstration) main purpose it to demonstrate the effects of PID control on a dynamic system. This was accomplished by integrating the four founding concepts of this project: Sensing, Actuation, Computation, and Mechanical Design.

Sensing

The primary sensing implementation in this project was a digital encoder. We mounted a HEDL 5540 encoder to the back of the Maxon RE30 24V that will be discussed further in the Actuation section. This encoder had 500 counts per revolution, meaning that the counts had to be read and converted to degrees (or radians). This was accomplished utilizing the NI myRIO 1900 discussed further in Computation. This encoder's reading acts as the feedback to close the control loop. Every loop of the program read the current angular position of the encoder, processed it based on user-defined gains, and output a control effort to the servo amp.

Another form of sensing in this project comes from the use of potentiometers as part of the user interface. Four potentiometers were used in all: one for the control setpoint, and the other three for the PID gains. A difficulty in this, is given that a potentiometer is an analog device, it is very susceptible to noise. Because the pendulum is meant to be reliable, noise in the user input is unacceptable. To improve reliability within the pendulum's control, software-based Butterworth filters were used on each input to filter out most noise.

The last form of sensing utilized for this process would be the use of switches. This form of sensing is simplistic as switches are either on or off, and their functionality was written accordingly. If they were on, X happens and if not, Y happens.

Actuation

The actuation implementation within this project was a Maxon RE30 24V motor used in tandem with an AMC Brushless Servo Amp. The motor acts as the torque/control input to the inverted pendulum. The pendulum was mounted about its end to the motor shaft using a coupler. This allowed for sample calculations to be simple given the moment of inertia was comparable to a thin rod rotating about its end.

The servo amp is what allowed the motor to be driven with a -10V/10V analog output from the NI myRIO. The current gain for the amplifier was set at approximately 0.3A/V as this had been tested in Project 3 and provided good results. This gain also simplified the sample calculations, as some of them had already been completed for Project 3.

Computation

The computation in this project was performed by the NI myRIO 1900. This microcontroller uses the LabVIEW programming environment which contains a lot of useful, prebuilt functions for various tasks; some of which are specific to reading and writing from the myRIO. For this project, the National Instruments PID VI (Virtual Instrument) was originally implemented but was eventually replaced by custom PID calculations. The reason for this, is that for a reliable controller design, a full understanding of all the workings of the controller was desired as opposed to a "black box" of controls calculations.

One particular benefit of the myRIO that stands to be noted is its extremely high sampling rate and achievable loop rates. While the generic While Loop within LabVIEW is limited to a 1kHz loop rate, the Real-Time While Loop can achieve rates of up to 1MHz. The importance here is the fact that since digital sensing is used for the feedback loop of the inverted pendulum's controller. This project is meant to be a demonstration for Dr. Rose's Systems & Dynamic Control's course, which deals primarily with the continuous domain. The encoder, and its digital readings, are within the discrete domain. To try and close the gap between these two forms of control, a high loop rate was desirable. The higher the loop rate, the smaller the discretization of readings, and the closer overall the pendulum's response is to its continuous counterpart.

Within the myRIO programming, many functions and features are established. A general overview of this process is as follows: The myRIO takes in a counter input from the encoder. This counter input is converted to degrees and is compared to the setpoint, as measured by a potentiometer input, and error is calculated. This error is multiplied by a proportional gain, also set by a potentiometer. The derivative of this error is taken and multiplied by a derivative gain set by a potentiometer. The integral of this error is also taken and multiplied by an integral gain set by a potentiometer. With these three gains applied, they're outputs are all summed together and provided to the Servo Amp as a control effort, or command voltage.

While the above process describes the PID controller used in this project, other features were implemented as well, and controlled by toggle switch inputs. Three additional functions were implemented. The first of which was a frequency input. When this is enabled, the controller outputs a sinusoid with an amplitude of 10 and a frequency set by the setpoint potentiometer. This allows the user to see how various PID gains would affect the frequency response at frequencies between 0 and 2 deg/s. This small range was used to prevent instabilities at all PID gains.

The next function implemented was anti-windup. An integral controller takes the area between the current position and setpoint and uses it to create a control effort to "close the gap." In cases of large error, this can cause what is called windup. When the integral becomes so large, even when the error is closed, the integral controller is still exhibiting a control effort, which is now no longer decreasing error but increasing it. The anti-windup function reinitializes the integral to zero when it reaches saturation; for this project, saturation is 10/-10 as that is the output of the analog pins utilized on the myRIO.

The last function implemented was controls hold. This switch allows the user to pause the PID gains at their current values and turn the knobs without affect. For instance, if the IPCD is setup with only proportional gain, and the hold switch is turned on. The user can change all of the gains, and the pendulum will still only exhibit a proportional response. But as soon as the hold is turned off, the new gains will take effect and a new response can be observed.

Mechanical Design

This project implements a wide variety of concepts lending to the Mechanical Design portion of this project. The first of which would be the mounting method for the motor.

The electrical components were mounted to the base plate of the control module. The base plate is made of 1/8" A36 steel. A 3D printed part was used to secure the DC motor atop the AMC Servo Amp. The part has a hole to allow for the shaft of the motor to pass through so it may be connected to a pendulum. It also has holes through which screws may be placed to ensure the motor does not move. To fasten this part to the base plate, holes in both parts are present, with four 3" bolts passing through both. The AMC Servo Amp is sandwiched between the motor mount and the baseplate, with the 3" bolts passing through the mounting cutouts on the motor driver. The pendulum arm and dowel are 3D printed parts; they are press fit together to allow for a secure connection. A cavity within the pendulum arm allows pennies to be press fit inside it as to raise the moment of inertia, which allows for easier control of the motor position. This cavity allows more weight in the pendulum arm, without making it longer. A coupler is used to connect the pendulum to the shaft of the DC motor.

A similar method is used to mount the myRIO and the power supply. A part was also printed for the myRIO to interface with, with two 3" bolts used to secure this to the mounting plate. The power supply is sandwiched between the motor mount and the baseplate, with the 3" bolts passing through mounting holes on the power supply. Power for the myRIO and the power supply are provided by two power ports mounted to the back plate of the IPCD. The back plate also has an on/off switch mounted to it. The backplate is made of 1/16" aluminum sheet metal.

The front plate is made of 0.117" clear acrylic. This material was used as a means of allowing users to view the inner workings of the IPCD. This front plate houses four potentiometers used as control gain and the position inputs. These are used by the myRIO to determine voltage output (control effort) to the DC motor. Along with potentiometers, toggle switches are also mounted to the front plate. These switches are used as the feature controls mentioned in Computation.

To encapsulate the device, 3/4" AC radiata pine is used for the side walls and top. This material was chosen for aesthetic purposes as the IPCD was meant to look like a vintage stereo. The side pieces are mounted to the base plate via rivet nuts and screws. The rivet nuts were pressure fit into the wood after drilling appropriately sized holes. The screws pass through the baseplate and thread into the rivet nuts, as well as they serve as feet for the device. The top is secure to the side walls by use of wood glue and finishing nails. Each of these pieces have 1/8" wide cutouts which the front and back pieces slide into, ensuring they will not move without having to mount them. This method allows the front and back pieces to be replaced or altered easily.

Build Documentation

All files and models associated with this project can be found in the following GitHub repository: https://github.com/walterlivingston/Inverted-Pendulum-PID-Demonstration

Schematics and Drawings

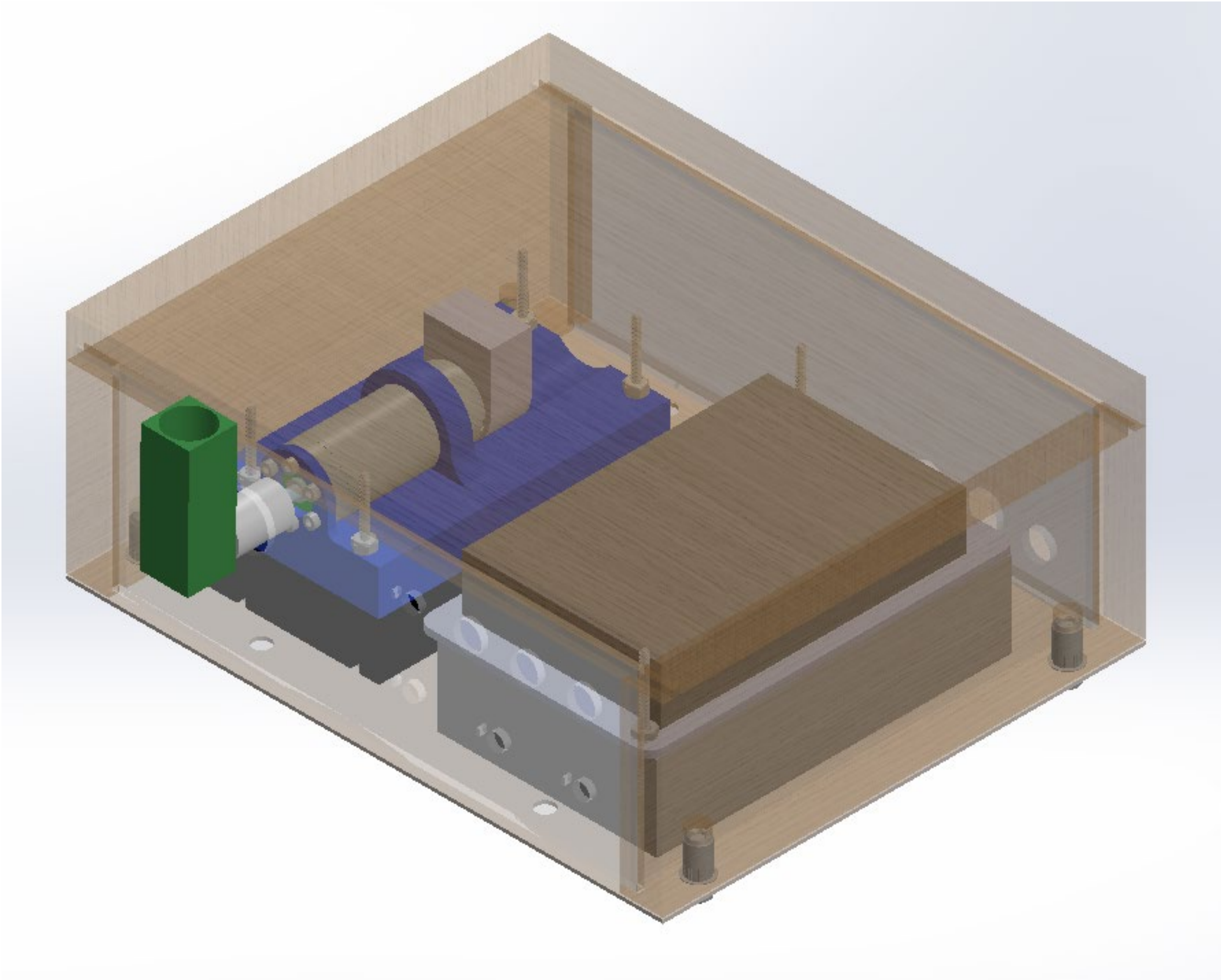


Figure 6: IPCD CAD Model

Assembly and Part Drawings

The drawing for each assembly (control module, motor mounting assembly, and pendulum) as well as part drawings designed for this project can be found in Appendix A.

LabVIEW Code & Explanation

The code implemented on the myRIO can be seen below in Figure 2. This visual code was written in the LabVIEW Environment. The box around everything here is known as a Real Time While Loop and in this project, was run at a loop rate of 1kHz.

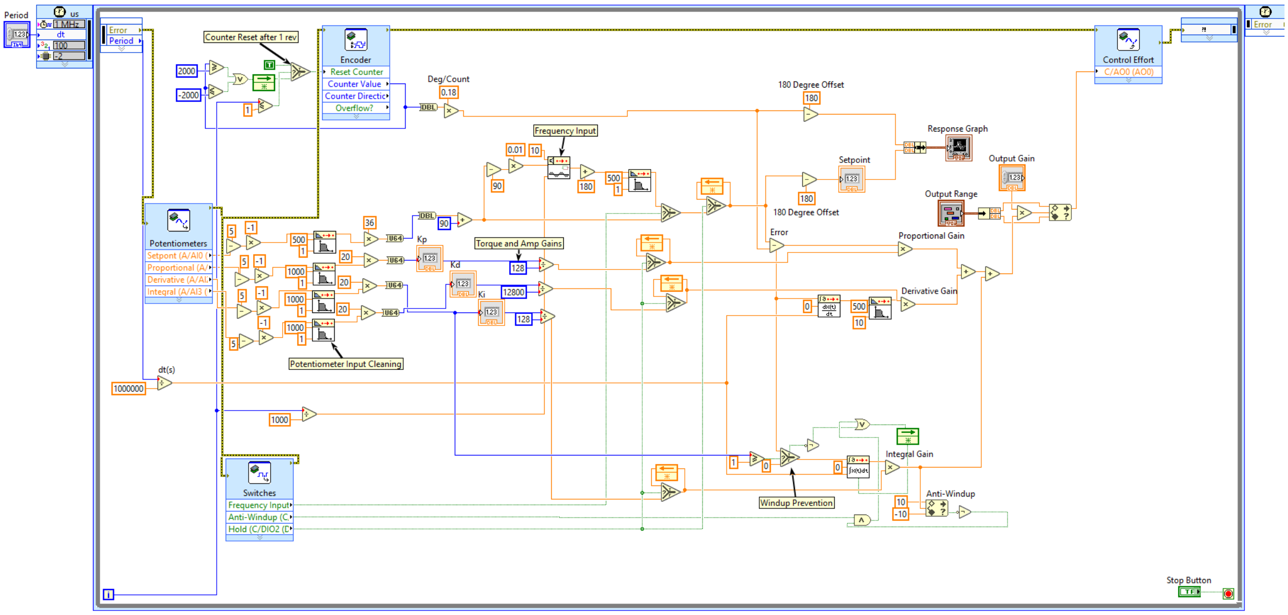


Figure 7: LabVIEW Code

The UI used to troubleshoot this project can also be seen in Figure 3. This includes a response graph showing the setpoint and the pendulum's position. It also contains indicators for the setpoint and PID gains. There are controls for an optional output gain (not used in the final version), the period of the loop, and inputs for an output range. These were also useful in troubleshooting the code shown above in Figure 2.

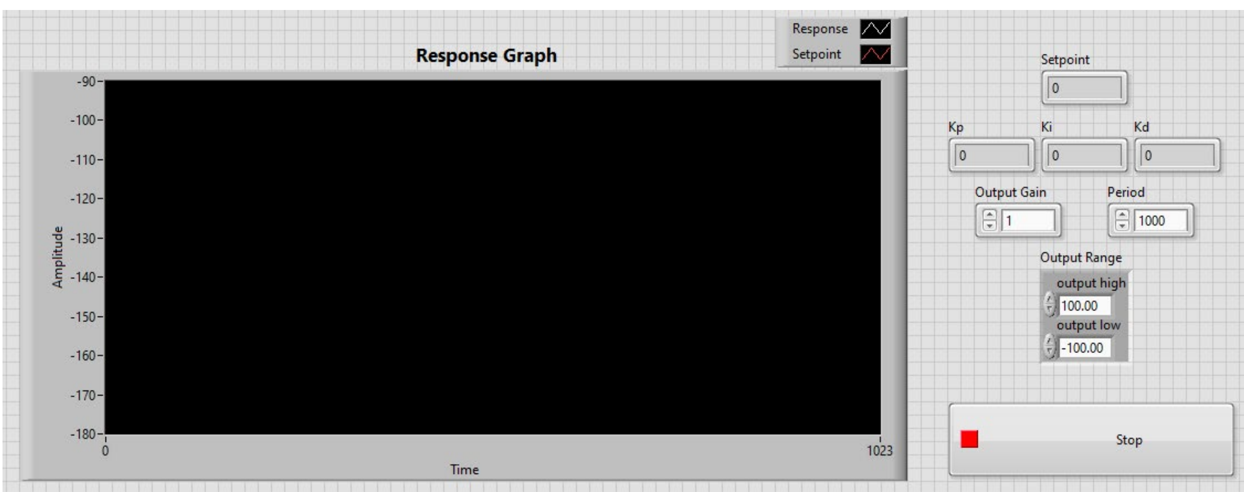


Figure 8: LabVIEW UI

The encoder readings were handled by the code snippet shown in Figure 4. This takes the encoder value and compares it to -/+ 2000. The reason for this is the HEDL 5400 encoder used in this project has a counts per revolution of 500. This being a quadrature encoder, that means that the myRIO is reading 4 times the counts per revolution, that being 2000. This comparison tells the myRIO to reset the counter if the encoder has gone a full revolution. It can also be seen her that something is being compared to the number 1. This is the index of the while loop. This bit of code tells the myRIO to reset the encoder counter if this is the first iteration of the while loop. Lastly, the encoder counts are also multiplied by 0.18 which is a conversion from counts to degrees.

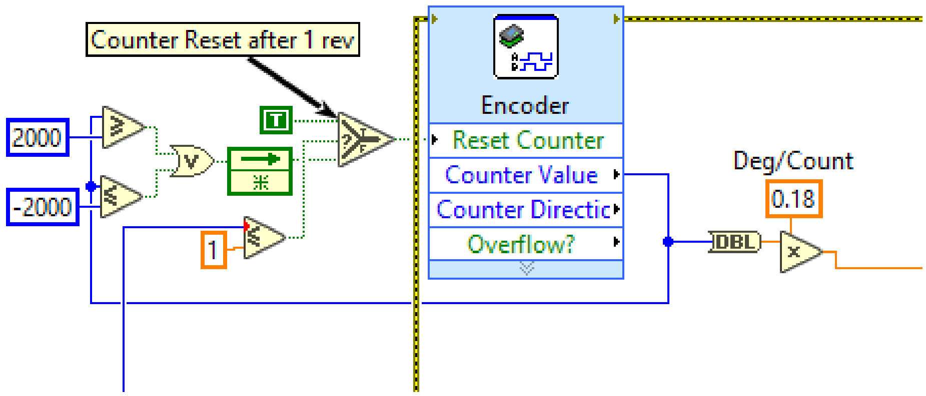


Figure 9: Encoder Processing Snippet

The PID controls calculations were handled by the code snippet shown in Figure 5. At the top of this figure, error is calculated from the setpoint and the encoder input, and passed into the P, I, and D components of controller. The proportional component is labeled in Figure 5, and is a simple multiplication of error by a gain. The derivative component is labeled and is shown in the middle of the figure. This uses the LabVIEW PtByPt Derivate VI as well as the PtByPt Butterworth Filter VI. The reason for this, is a PtByPt Derivative contains a lot of noise which would lead to instabilities in the control effort. It then multiplies this filtered derivative by a gain. Lastly the integral component is labeled and is shown at the bottom of the figure. It takes in the error and utilizes the PtByPt Integral VI to get the integral of the error. This is then multiplied by a gain and all three components are summed and passed to the analog out of the myRIO.

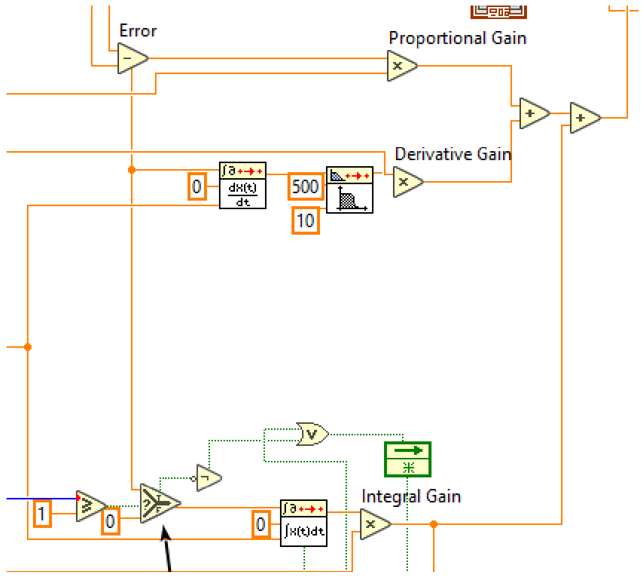


Figure 10: PID Calculations Snippet

The filtering and processing of the potentiometer inputs are shown in Figure 6. First, all inputs had 5 subtracted from them and were multiplied by 1. This was because the potentiometers inputs were between 0 and 5V, but because of their orientation, this range was reversed. This subtraction and multiplication rectified that. All the inputs are then filtered by the PtByPt Butterworth Filter, and they are multiplied by the conversion factors. For the setpoint, a range of -90 to 90 was desired, while for the gains, a range of 0 to 100 was desired. These values are then cast to integers to attain clean, non-floating-point values and the gains are further processed with another conversion factor. They are divided by the number 128 as this is a product of the motor's torque constant and the servo amps voltage gain. The derivative gain's conversion is actual 12800 as the pendulum requires very small derivative gains to remain stable.

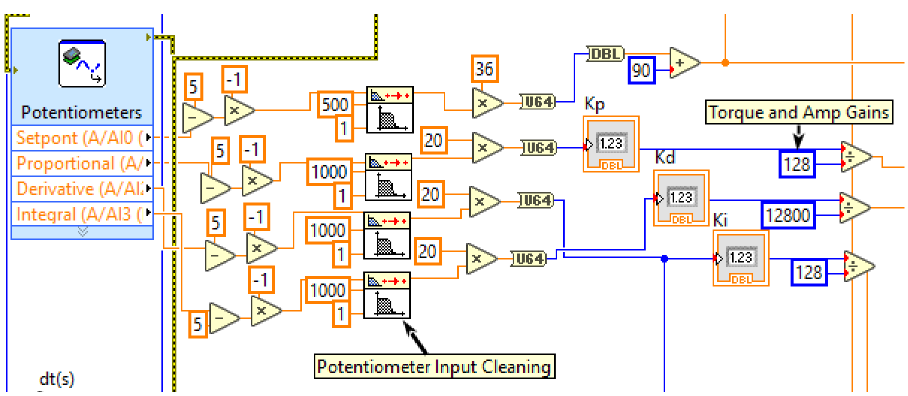


Figure 11: Input Filtering Snippet

Improvements

Structural

The structure of the IPCD is rigid and compact, although this does bring up some issues. The device would have been easier to work with if the tolerances between the internal components and the front and back plates were larger. Due to these tight tolerances, wires were jumbled and would often get in the way. Another issue with the front and back plate is with the depth of the slots in the wood which these parts are inserted. The slots are slightly too deep, part of this can be contributed to the plywood not being quite 3/4" thick. If the actual size were considered this issue would have been avoided. Because the back piece is thinner than the slots, it is able to move. To minimize this movement, the wood side pieces were slanted inwards slightly.

Taking the wood exterior pieces off is simple, however the nut/bolt connection could be improved. The use of river nuts is because initially the plan was to use more sheet metal parts so these nuts would allow for loosening of the bolts without having access to the nut, as it would be inside the IPCD. A better alternative would be threaded inserts, as they have external threads for mounting. Currently, the rivet nuts used do not have a way to solidify their connection to the wood, the use of the proposed inserts would contain a solid connection.

The pendulum arm is short enough that the device can sit on the table without having to hang the pendulum over the edge, however, it still may be too long. At high angular, the pendulum has the potential to come into contact with the set point knob. If the system were given gains that would make it unstable, the user's hand could cross the pendulum's path, resulting in some pain as this motor is powerful. The issue with the length of the pendulum could be solved in one of two ways: make the pendulum shorter or shift the motor mounting assembly much closer to the wall, there is up to 5/8" of room between the motor mounting assembly and the wooden side wall.

Functional

The IPCD works really well and was an overall successful project. There were many aspects, that in hindsight, the team would've changed. The first of these changes would be the use of an analog position sensor rather than a digital encoder. There are a few reasons for this, the primary of which being that the pendulum's zero is reset every time the myRIO is power cycled. If the pendulum is not point straight down, when the myRIO is powered on, the zero position is offset. An analog sensor would not have this issue. An analog position sensor would also make the IPCD more analogous to a continuous sensor, where the digital is by nature discrete.

Another functional improvement would be the addition of a screen to display the response of the pendulum as it happens. The frequency input switch is useful to display the frequency response, but it is clearer to see how different gains affect the response using a scope such as the one shown in Figure 3. The team had plans to do this through a serial connection with an Arduino Nano, and the screen Dr. Rose had purchased, but there was not enough time to implement this.

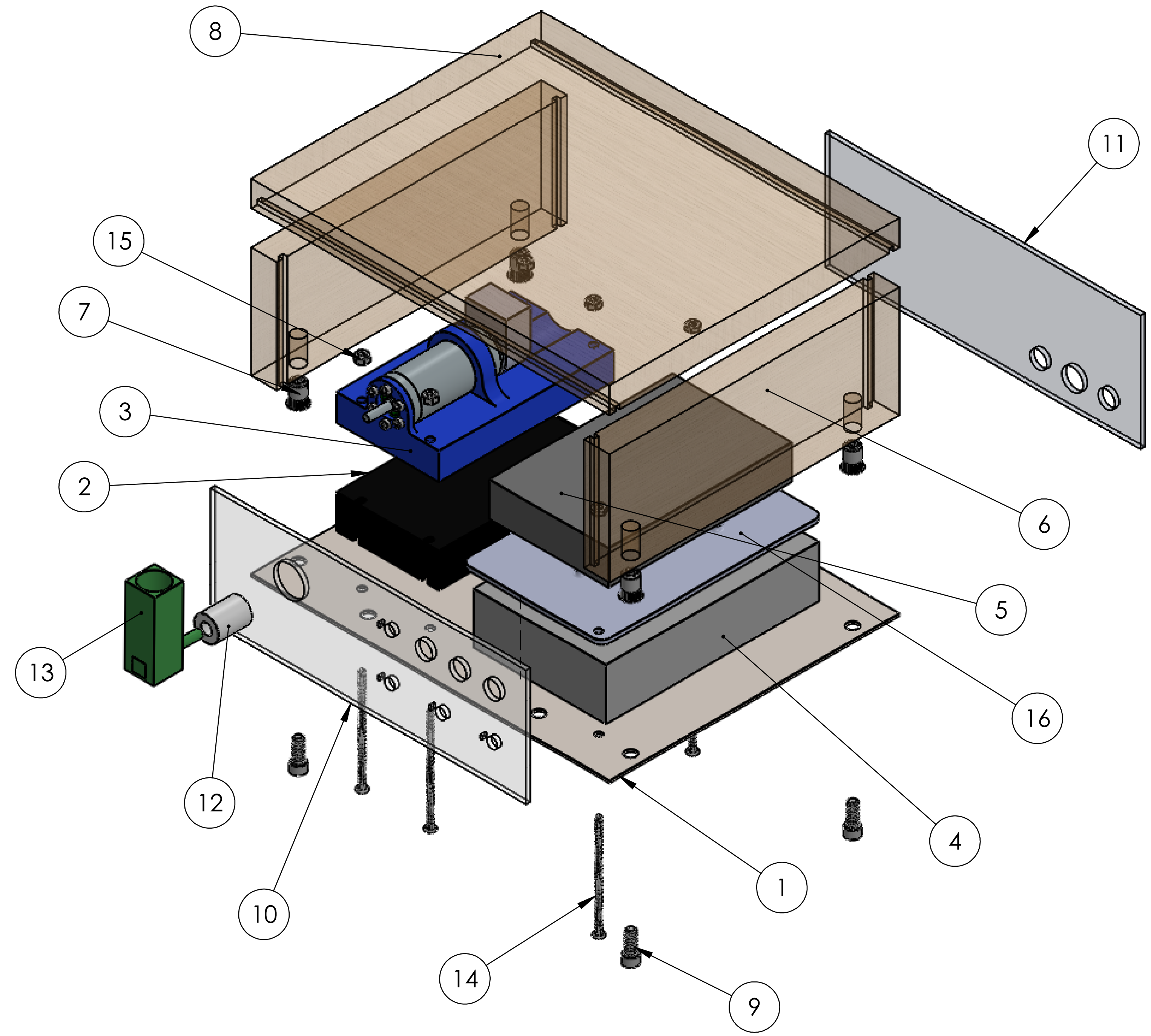
The last functional improvement would be a manual controls mode. By this, I mean a pin header on the rear of the pendulum and a switch to toggle this controls mode. This would allow students to develop their own PID controller and test it on this pendulum. Aside from the analog position sensor, this would be the largest improvement to the project. With this addition, not only would the IPCD be useful for Systems & Dynamic Controls as a demonstration, but would also be useful for Applied Mechatronics projects 2 and 3.

Appendix A: Assembly and Part Drawings

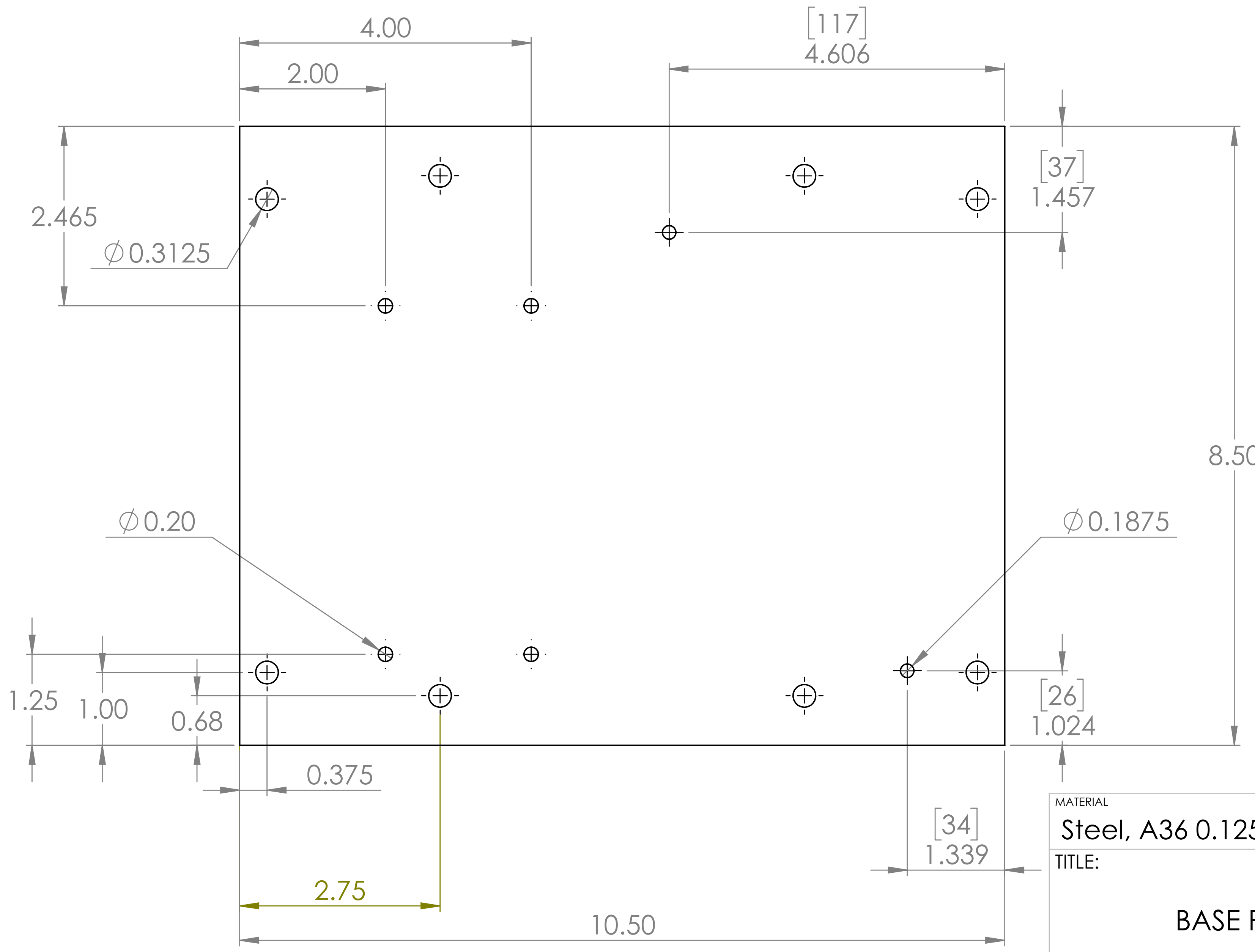
The drawing order is as follows:

- Control Module Assembly
- Base Plate
- Motor Mount Assembly
    - Motor Mount
- Wood Side Piece
- Wood Top Piece
- Front Piece
- Back Piece
- Pendulum Assembly
    - Pendulum
    - Pendulum Dowel
- myRio Mounting Plate

ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	Base Plate	Steel, A36 0.125"	1
2	Motor Driver		1
3	Motor Mounting Assembly		1
4	Power Supply	24V Power Supply	1
5	myRio		1
6	Wood Side Piece	AC Radiata Plywood	2
7	97467A725	18-8 Stainless Steel Heavy-Duty Rivet Nut	4
8	Wood Top Piece	AC Radiata Plywood	1
9	91274A160	Zinc-Aluminum-Coated Alloy Steel Socket Head Screw	4
10	Front Piece	Clear Acrylic 0.117"	1
11	Back Piece	Aluminum 16 gauge	1
12	Coupler		1
13	Pendulum Assembly		1
14	90272A163	Zinc-Plated Steel Pan Head Phillips Screw	6
15	90631A007	Low-Strength Steel Nylon-Insert Locknut	4
16	myRIO_Mounting_Plate V2		1
17	90675A007	Steel Locknut with External-Tooth Lock Washer	2



MATERIAL: Multiple
TITLE/DESC:
FILE NAME: Enclosure Assembly
DWG. #: REV #:
SCALE: 1:4 SIZE B SHEET 1 OF 1



MATERIAL
Steel, A36 0.125"

TITLE:

BASE PLATE

SIZE A

DIMENSIONS IN: inches [mm]

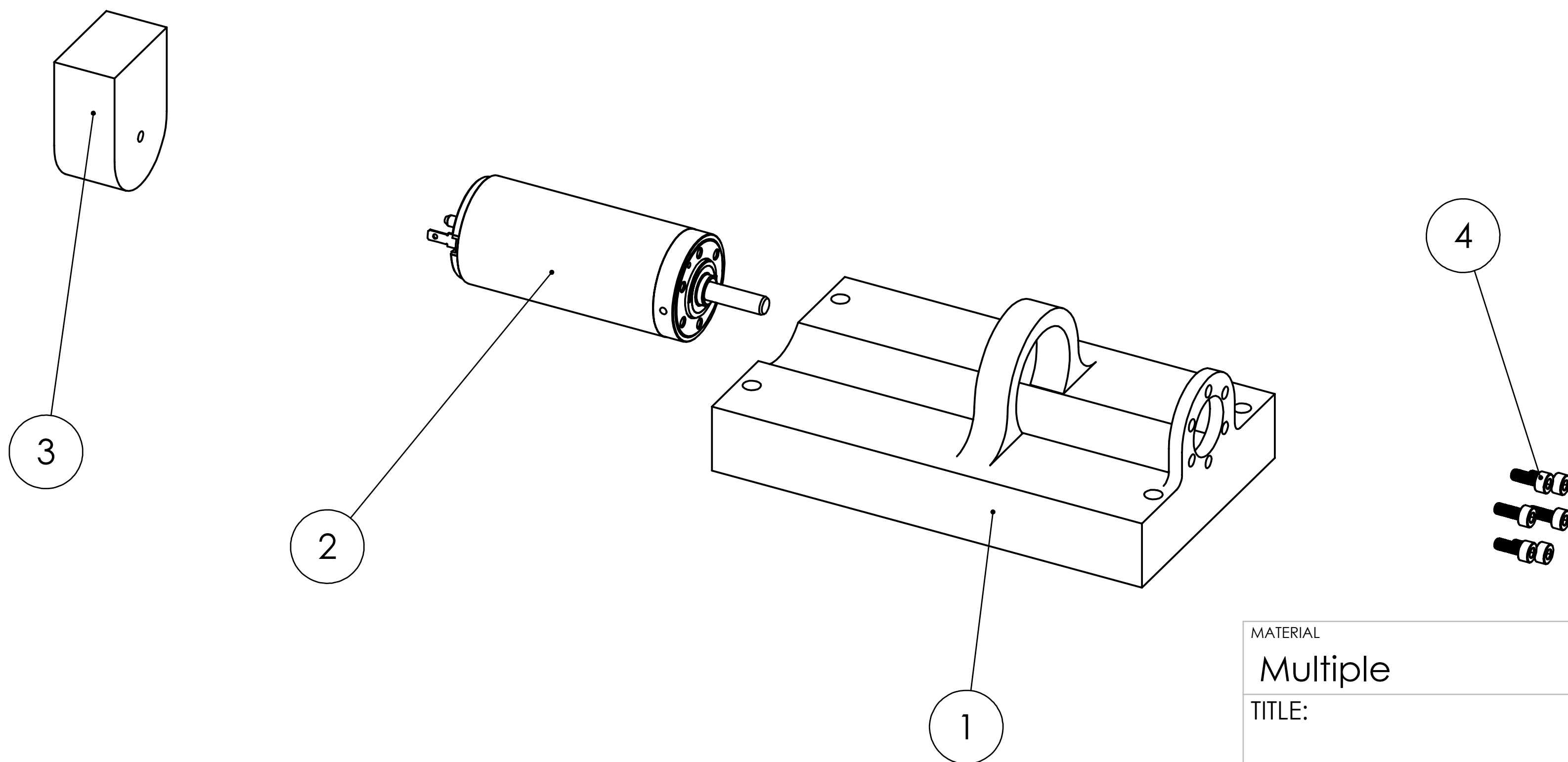
SCALE: 1:2

SHEET 1 OF 1

2 1

ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	Motor Mount		1
2	Maxon Motor	24V DC Motor	1
3	Encoder		1
4	91274A103	Zinc-Aluminum-Coated Alloy Steel Socket Head Screw	6

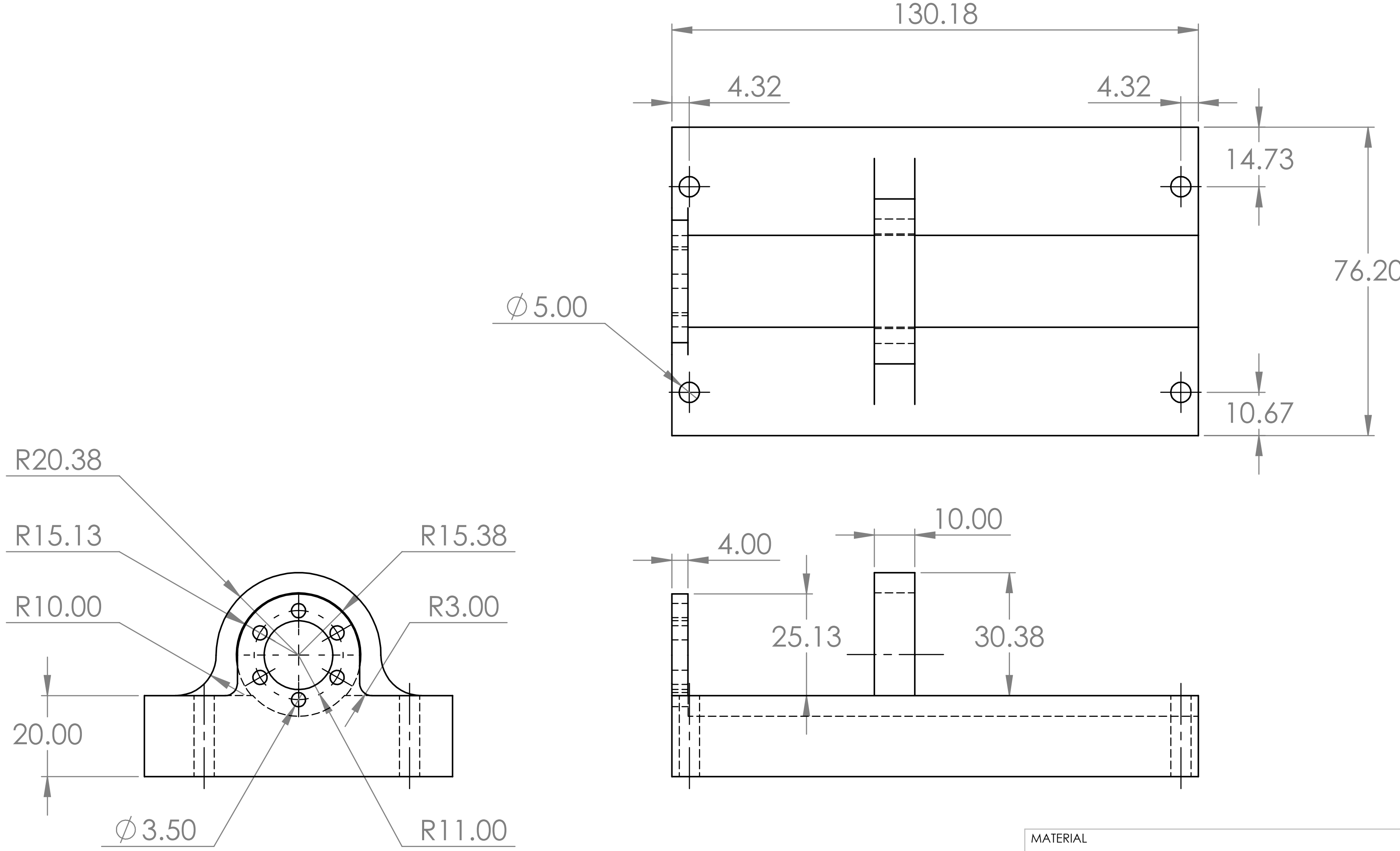
B B



A A

MATERIAL Multiple	
TITLE: MOTOR MOUNTING ASSEMBLY	
SIZE **A**	
SCALE: 1:2	SHEET 1 OF 1

2 1



MATERIAL

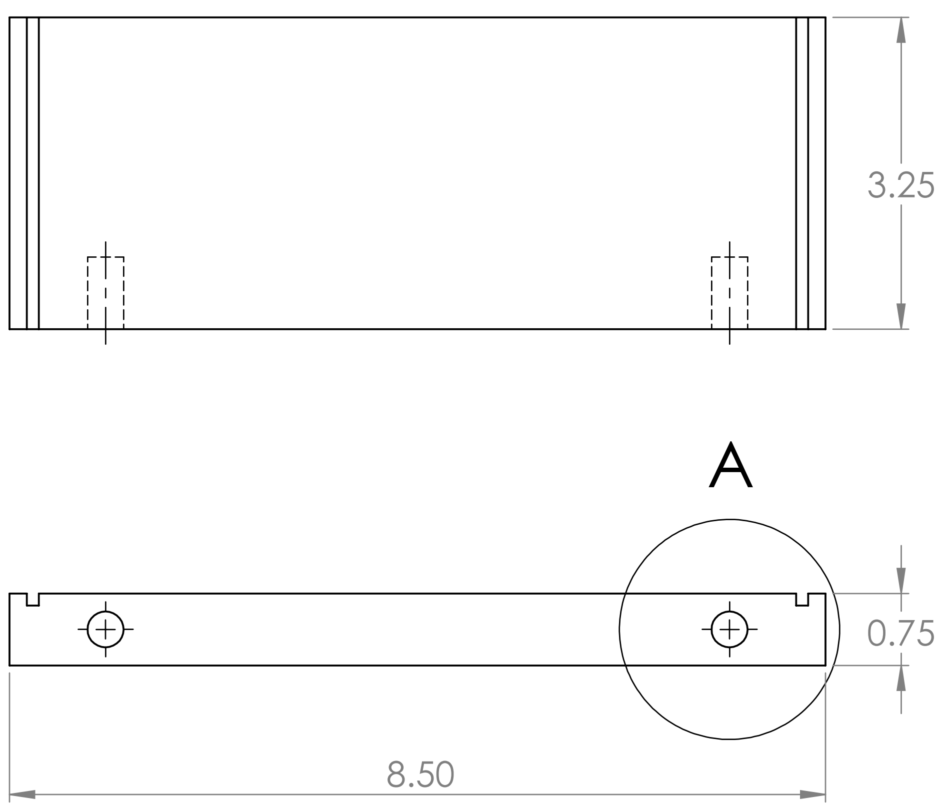
Polylactic Acid

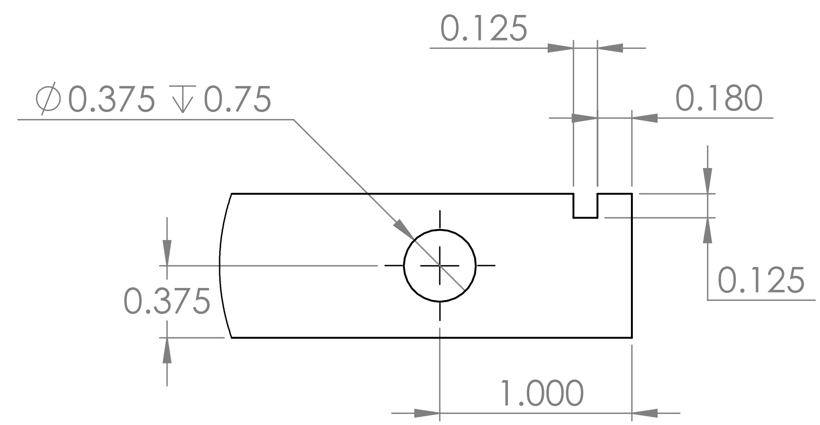
TITLE:

MOTOR MOUNT

SIZE A | DIMENSIONS IN: mm

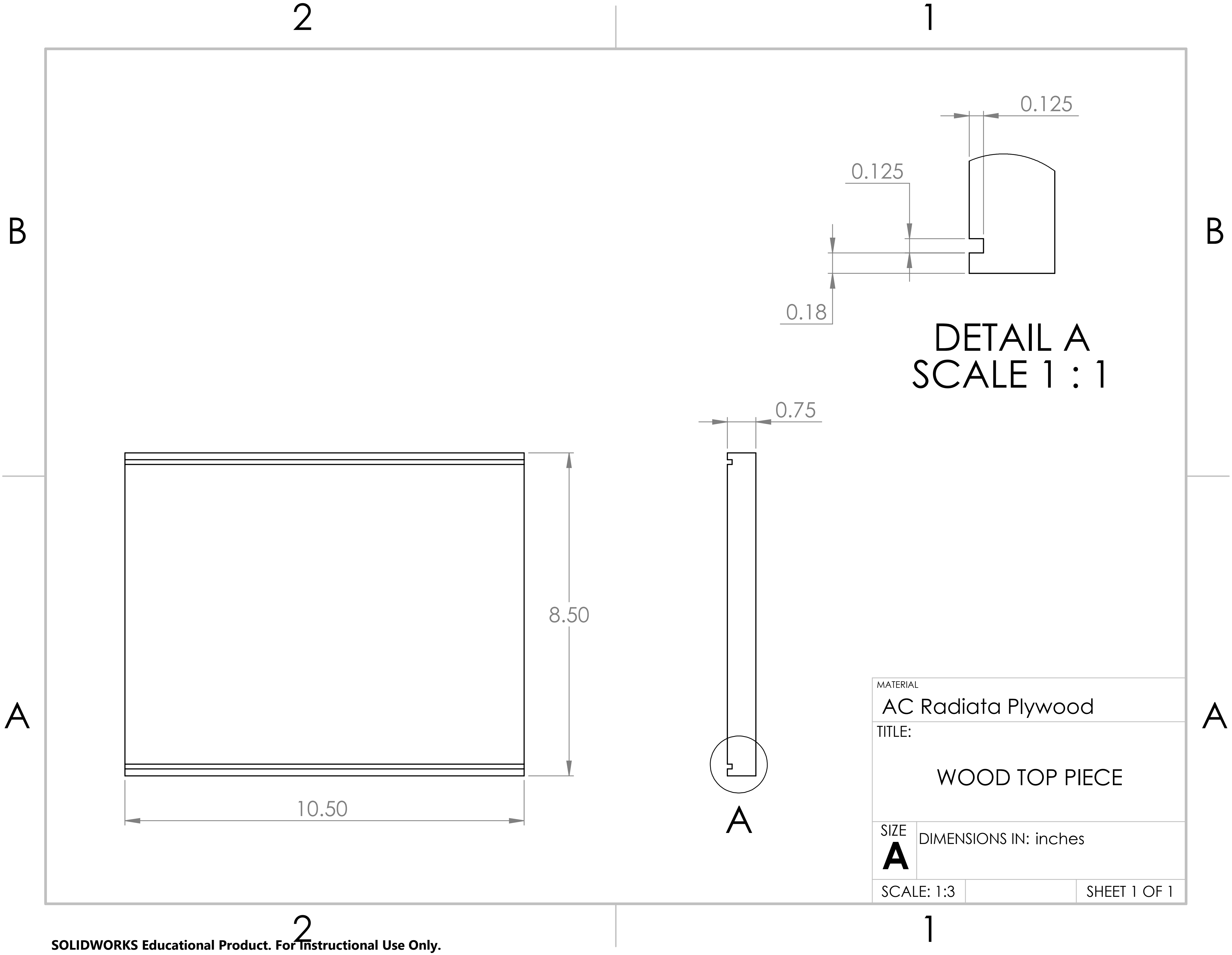
SCALE: 2:3 | SHEET 1 OF 1



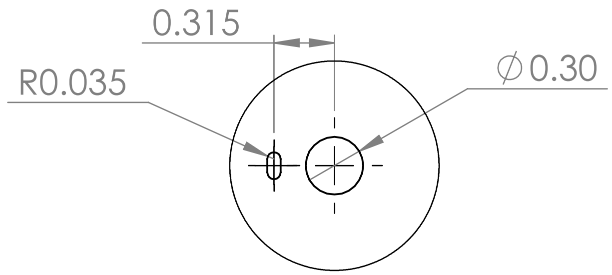


DETAIL A
SCALE 1 : 1

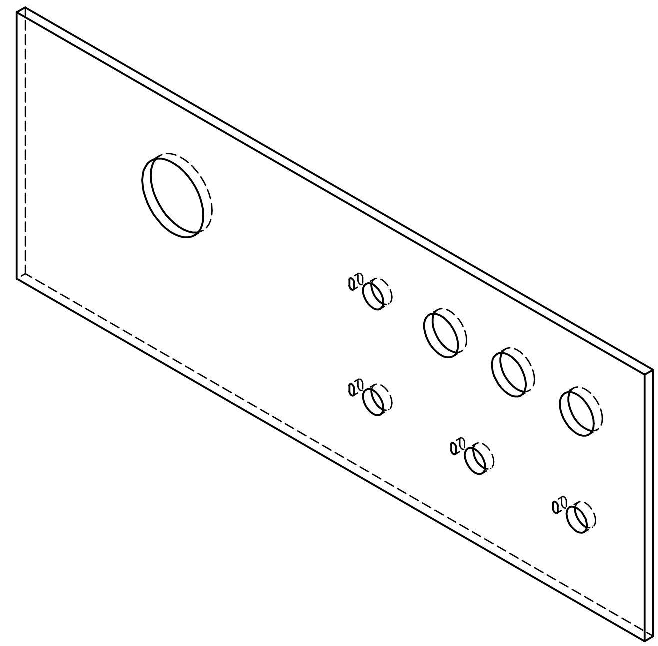
MATERIAL	AC Radiata Plywood
TITLE:	WOOD SIDE PIECE
SIZE A	DIMENSIONS IN: inches
SCALE: 1:2	SHEET 1 OF 1

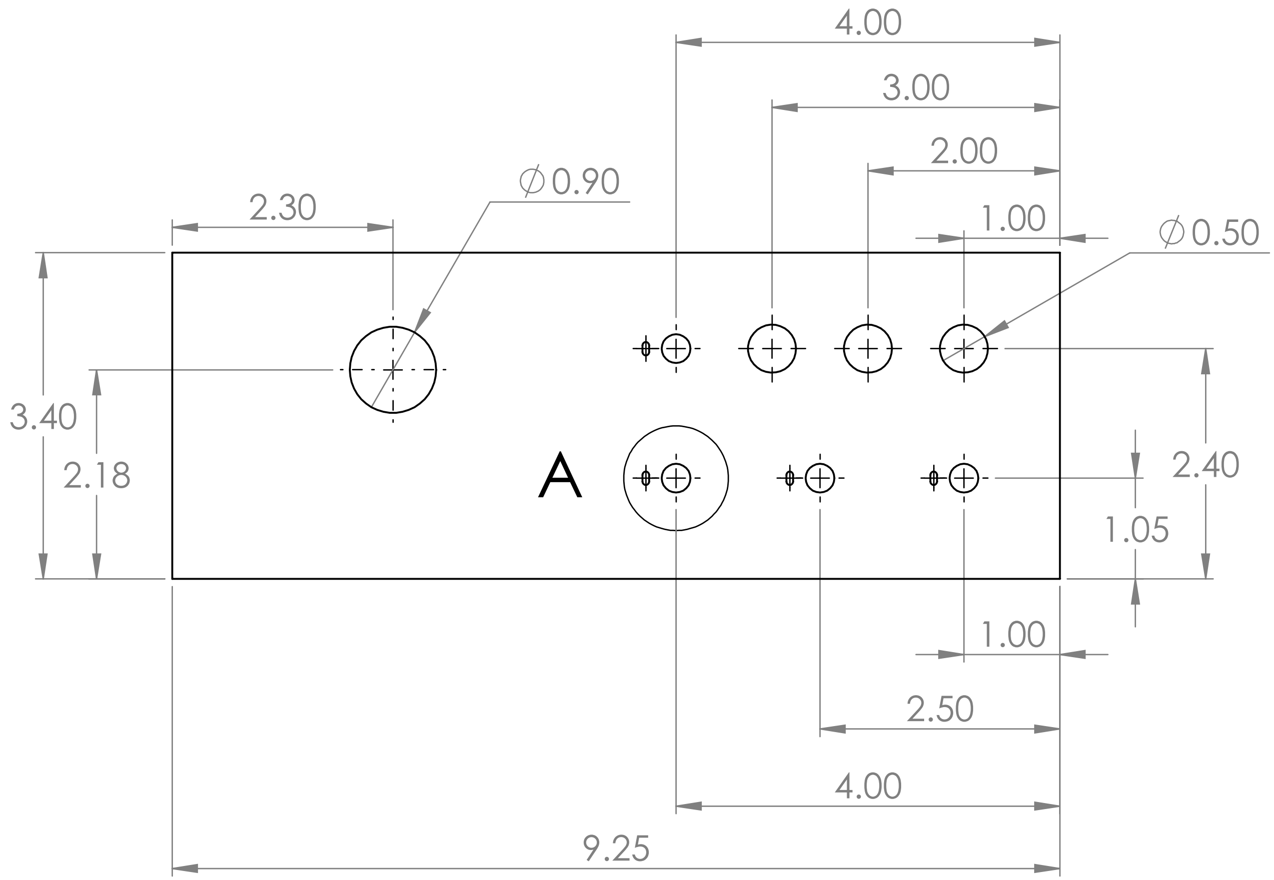


MATERIAL		
AC Radiata Plywood		
TITLE: WOOD TOP PIECE		
SIZE A	DIMENSIONS IN: inches	
SCALE: 1:3		SHEET 1 OF 1

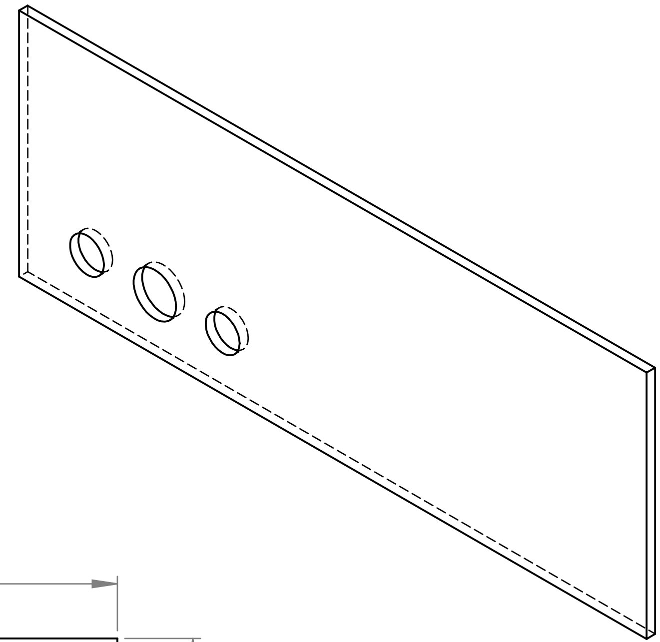


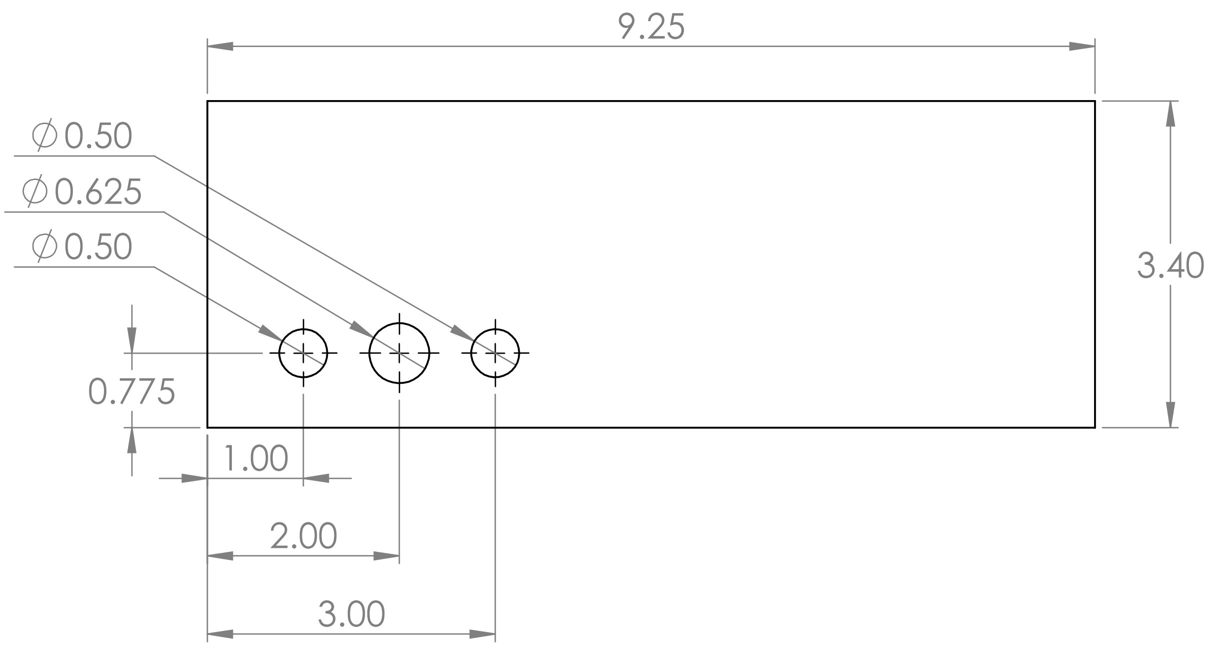
DETAIL A
SCALE 1 : 1





MATERIAL Clear Acrylic 0.117"	
TITLE: FRONT PIECE	
SIZE A	DIMENSIONS IN: inches
SCALE: 1:2	SHEET 1 OF 1





MATERIAL

Aluminum Sheet 16 gauge

TITLE:

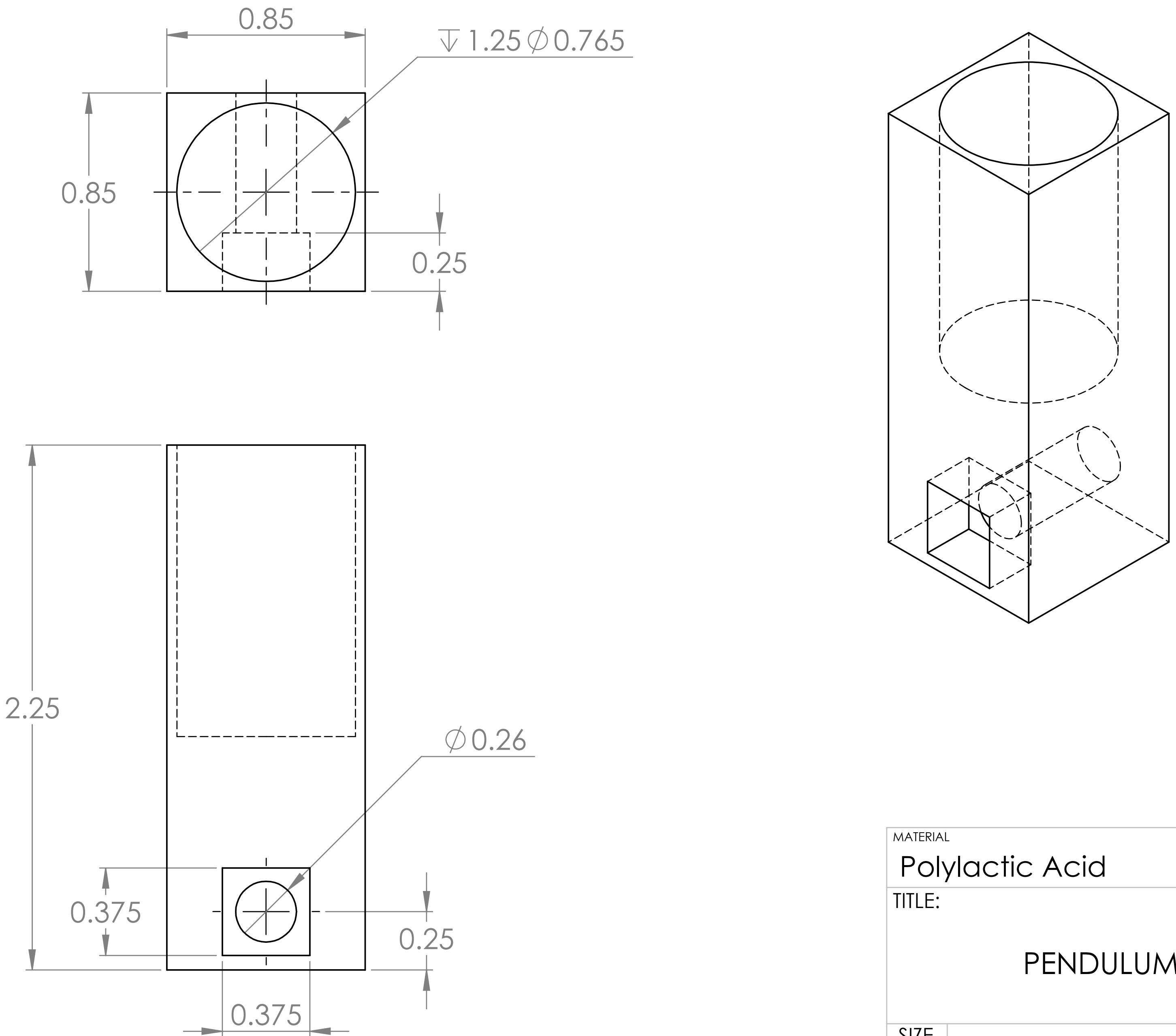
BACK PIECE

SIZE **A**

DIMENSIONS IN: inches

SCALE: 1:2

SHEET 1 OF 1



MATERIAL

Polylactic Acid

TITLE:

PENDULUM

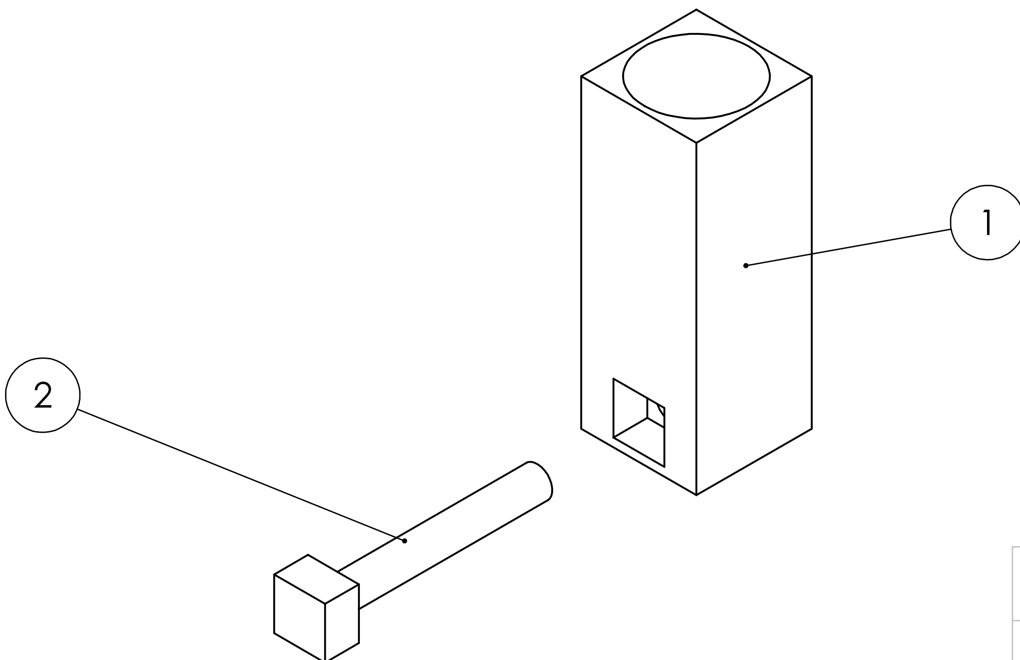
SIZE **A**

DIMENSIONS IN: inches

SCALE: 3:2

SHEET 1 OF 1

ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	Pendulum		1
2	Pendulum Dowel		1



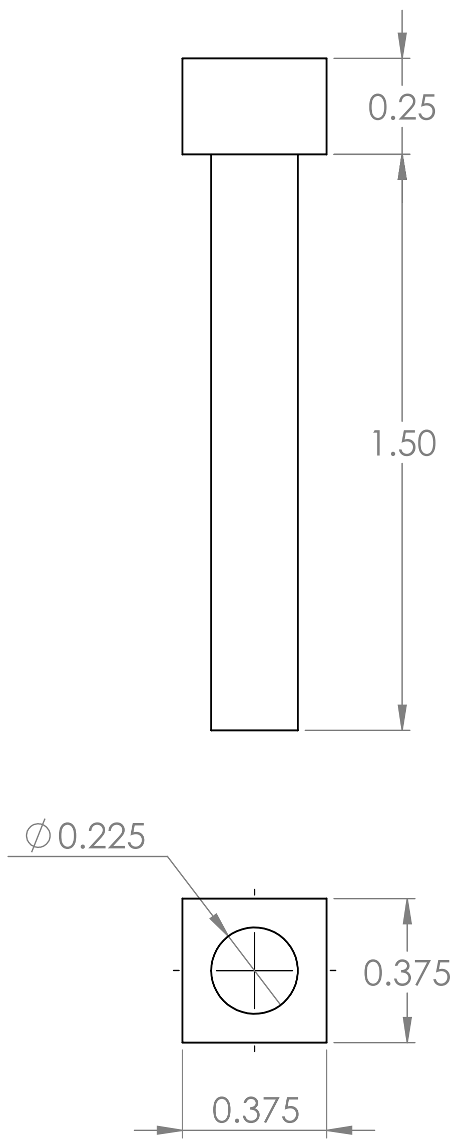
MATERIAL
Polylactic Acid

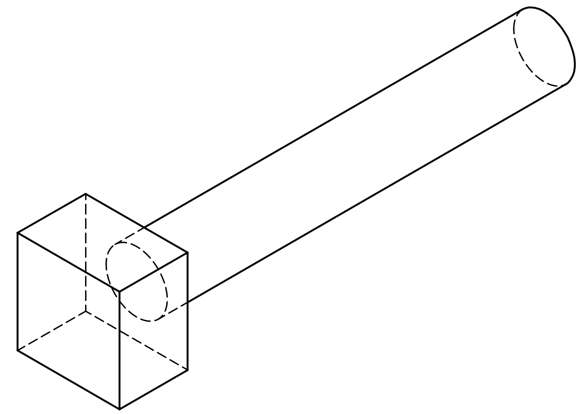
TITLE:
PENDULUM ASSEMBLY

SIZE
A

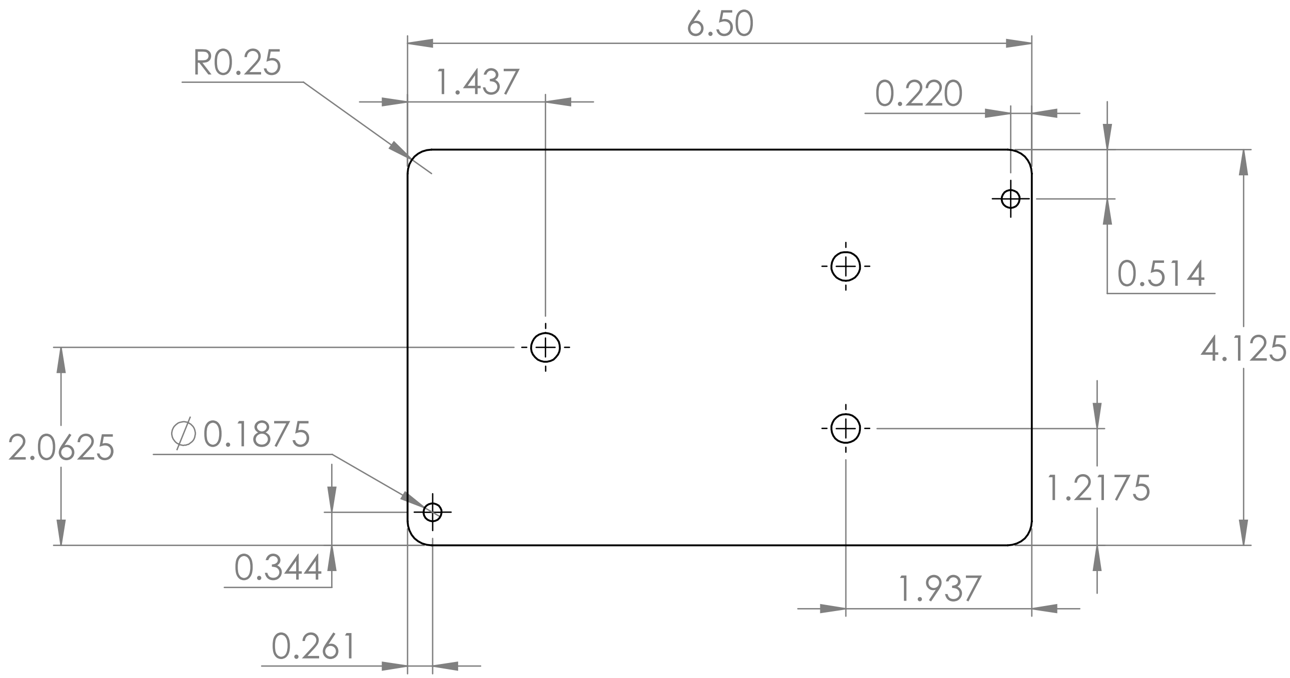
SCALE: 1:1

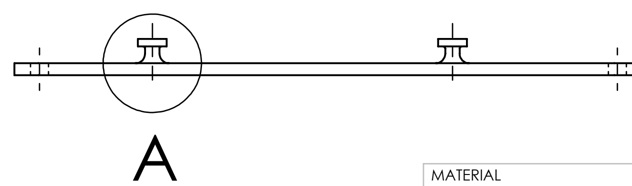
SHEET 1 OF 1

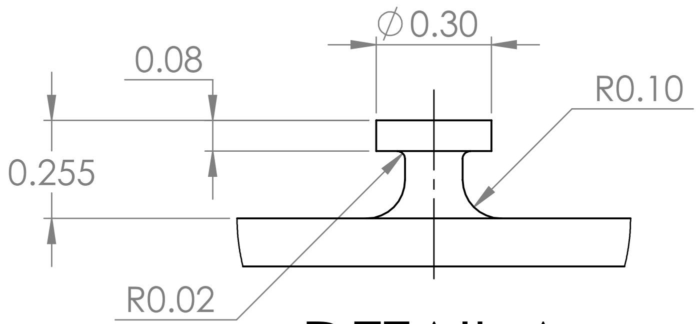




MATERIAL Polylactic Acid	
TITLE: PENDULUM DOWEL	
SIZE A	DIMENSIONS IN: inches
SCALE: 2:1	SHEET 1 OF 1







DETAIL A
SCALE 2 : 1

MATERIAL	Polylactic Acid
TITLE:	MYRIO_MOUNTING_PLATE V2
SIZE A	DIMENSIONS IN: inches
SCALE: 1:2	SHEET 1 OF 1