SCARA DRAWING ROBOT

By

Tianhao Chi

Chenghao Duan

Bingzhe Wei

Design Document for ECE 445, Senior Design, Spring 2018

TA: Dongwei Shi

February 20, 2018

Project No. 19

Contents

1. Introduction

1.1 Objective

Since the advent of computers and subsequently artificial intelligence, a key issue, and one that remains particularly elusive, has been creativity, the ability to generate output that is both novel and of significance as would be recognizable by humans. Although alternative programming paradigms that require less strict process specification and that can infer certain details automatically have been developed, these logic-based paradigms still require relative stringent and particular specifications as compared to natural language descriptions and are more difficult to use, and hence most computing today is still based on the imperative paradigm, resulting in algorithms can solve well-defined specific problems for which they have been designed to solve with high computational efficiency, but require manual algorithm discovery and implementation.

Whereas robotics is now commonly used in many industrial applications, there have been far fewer applications of robotics for artistic purpose, and we are yet to see a robot artist. As compared to other robotic arm designs, the Selective Compliance Assembly (SCARA) robotic arm design, originally designed for assembly line work, enables very high manipulator velocities with high positioning accuracy [1]. We thus seek to combine robotics, specifically, a SCARA robotic arm, together with deep neural networks, to create a robot capable of learning the art style of any image, applying the learned style, or a synthesized novel style, to any given input image, and finally output the transformed output image onto a physical medium, and hence demonstrate a significantly higher level of creative behavior by an artificial intelligence system.

1.2 Background

Early examples of robotic art systems include Maillardet's automaton [2], a mechanical plotting machine created back in 18th century based on a clockwork-type mechanism that traced a master copy held within the mechanism. More recently, there have been computed generated art programs, such as The Painting Fool [3] and AARON [4]. However, all of these previous systems were either based on replicating a master copy designed by a human, or were based on careful parameter selection and relatively rigid, manually designed algorithms, with the algorithms either based on combinations of rules and/or art primitives, or mathematical relationships such as fractals, chaotic systems, and Lissajous patterns [5, 6, 7], resulting in computer generated art often having a line-dominant, recursive structure.

Recently, there have been renewed interest in deep neural networks for image processing applications [8]. Unlike conventional programming, neural networks require comparatively little human input – only a set of primitives and their interconnections, with most of the detailed instruction specification required by an imperative programming paradigm instead automatically inferred via parameter optimization over a given set of input/output example pairs. Viewed as universal function approximators, neural networks automatically interpolate and extrapolate over the given input/output domains, thus enabling the creation of arbitrary and potentially novel output from a finite set of examples without necessitating human engineering [9]. The emergence of deep neural networks therefore enables the generation of art without first defining any particular algorithm, enabling the generation of art with arbitrary styles and increased complexity which would be otherwise difficult to describe with primitives or mathematical relationships. In particular, by taking advantage of the generalization properties of deep neural networks, it is possible to generate new artistic styles from examples of existing art without additional human intervention.

1.3 High-Level Requirements List

a. The robotic arm should be able to draw an image of size bounded by the size of a sheet of Letter-sized paper with arbitrary content onto a sheet of Letter-sized paper.
b. The PC-side processing program should be able to produce an artistically appealing output image for any user given input image.
c. The PC-side processing program should be able be transform the above output image into a set of commands for the robotic arm.

2. Design

2.1 Block Diagram

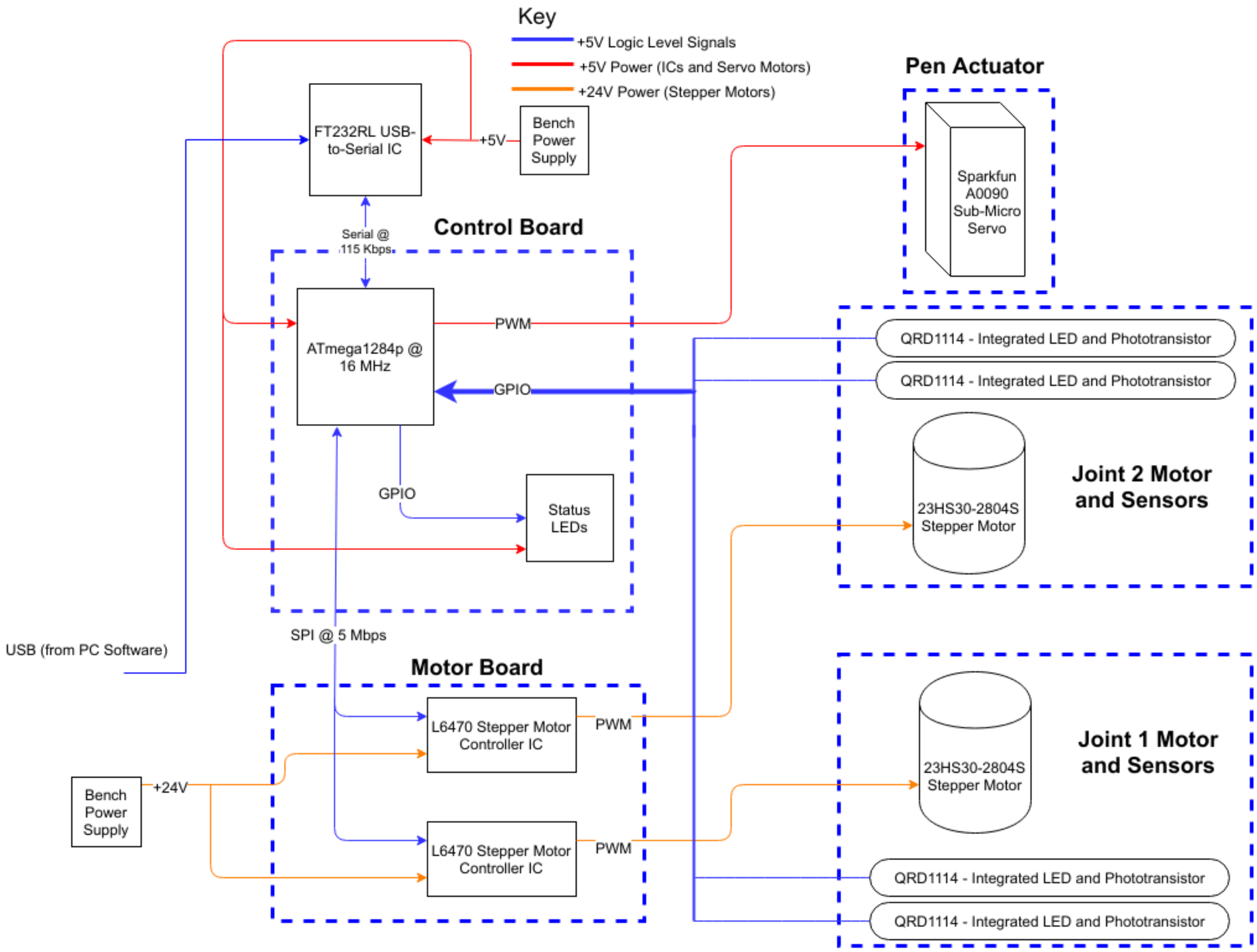


Figure 1: Block Diagram

Overall, the system consists of four components: software running on a PC performing all image processing steps, a control board with a microcontroller to coordinate all motor movements, a motor driver board to control the stepper motors required due to the high current and voltage needed to drive the stepper motors, and the mechanical components of the SCARA robot arm. Given an input image and a style image, the PC software first performs style transformation on the input image using a deep neural network, then converts the result into a series of arm angles by clustering colors and thresholding to get color regions, which are then converted into a series of lines via a vectorization algorithm, before finally transforming the lines into a list of arm angles via inverse kinematics. The result is sent to the control board via a USB serial port provided by a USB-to-Serial IC, and subsequently, the control board sends stepper motor positioning commands via SPI to the stepper motor controller ICs on the motor board, and pen actuator commands directly as PWM waveforms. The resulting sequence of motor motions then produces physical output on a piece of Letter-sized paper placed appropriately beforehand in the robotic arm’s workspace.

2.2 Physical Design

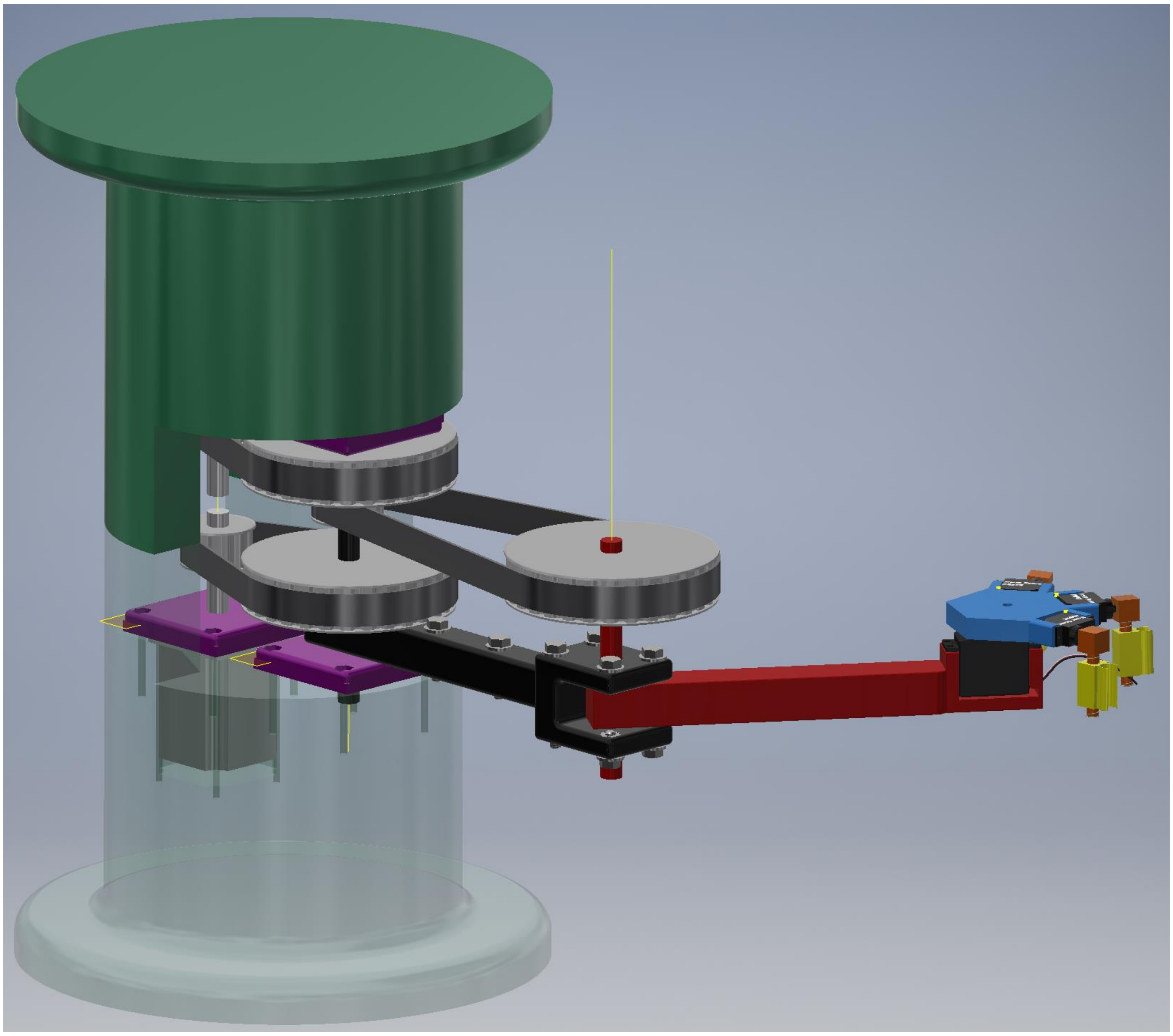


Figure 2: Proposed Design SCARA Robot Diagram

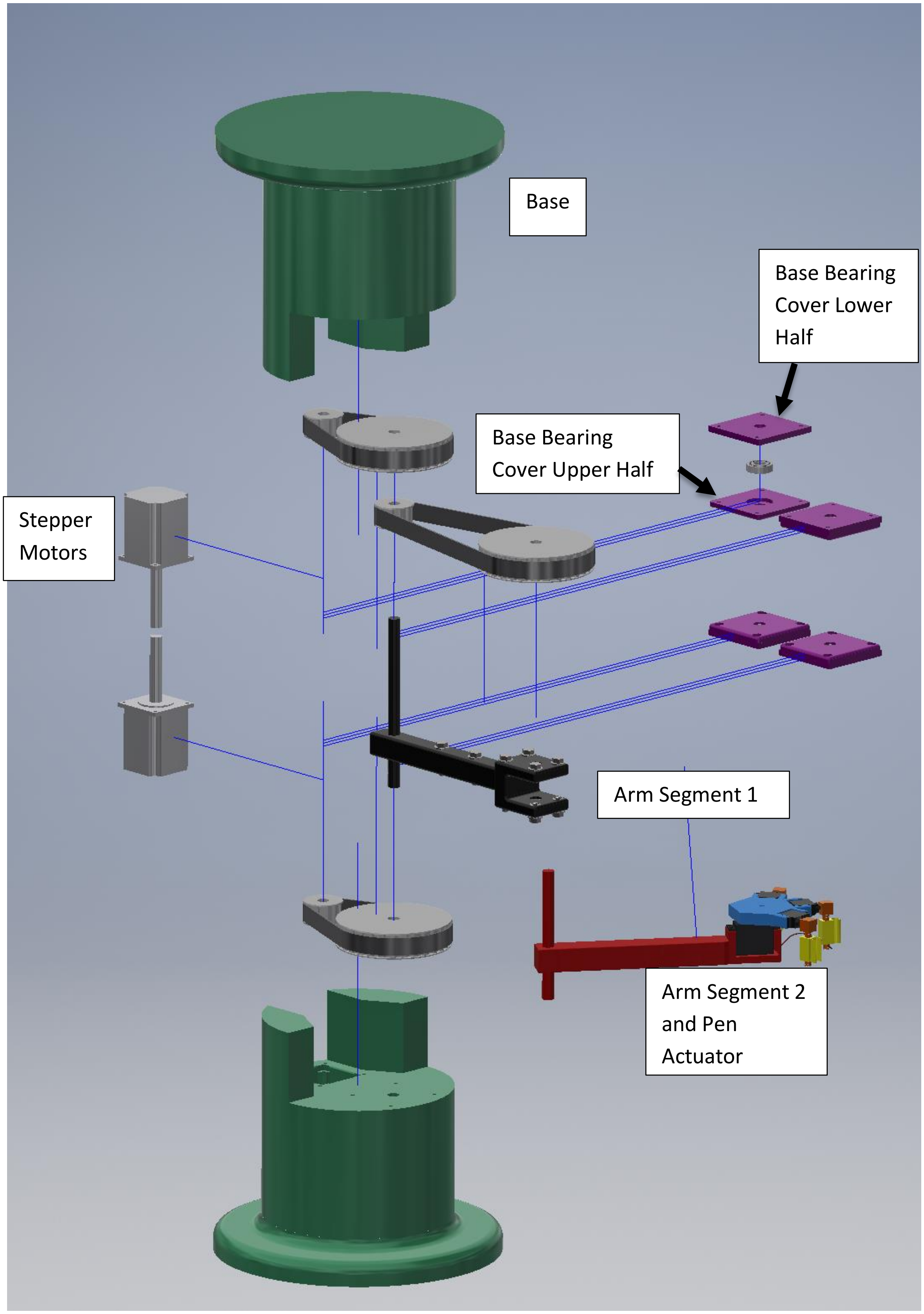


Figure 3: Proposed Design SCARA Robot Exploded View

The physical design of the proposed robot arm is shown in Figure 2 above, while the block diagram of the SCARA robotic arm is shown below in Figure 4. The SCARA arm type consists of a chain of two arm segments interconnected with a revolute joint, with a rotational platform controlling a pen located at one end and another revolute joint at the other end that joins the arm with a base. Each revolute joint is driven via stepper motors though a short timing belt and aluminum gears and supported via bearings to ensure accurate movement with low slippage, while rotational platform is driven by a standard size servo to allow for pen color changes, with three sub-micro servos each attached to a pen to enable accurate positioning with minimal weight and size. The pen is placed in and out of contact with the paper by rotating the sub-micro servo attached to that pen. As compared to other robotic arm designs, the SCARA robotic arm design, originally designed for assembly line work, enables very high manipulator velocities with high positioning accuracy [1], as all joints are positioned such that self-collisions are impossible, enabling the ability to move all joints independently and simultaneously. Meanwhile, the arm provides compliance only in the x-y plane [10], ensuring that the pen will always stay perpendicular to the work surface and thus further enhancing accuracy. As the workspace of a SCARA robotic arm is a distorted cardioid-like shape, we plan to have each arm segment be eight inches long, such that a piece of Letter-sized paper fits within the workspace of the robotic arm. We expect to construct the robotic arm via 3D printing and bolted connections. Fast and accurate positioning, with very low vibration, is required to produce a high-quality output with true edges instead of wavy lines in a reasonable amount of time, which necessitates not only high-powered motors but also smooth, low-slip, low-backlash joints and transmissions, and we have used synchronous belt drives to drive each arm segment. As can be seen from Figure 2, Arm Segment 1 is driven by a 1:3 ratio synchronous belt drive, while Arm Segment 2 is driven by two 1:3 ratio synchronous belt drive in series.

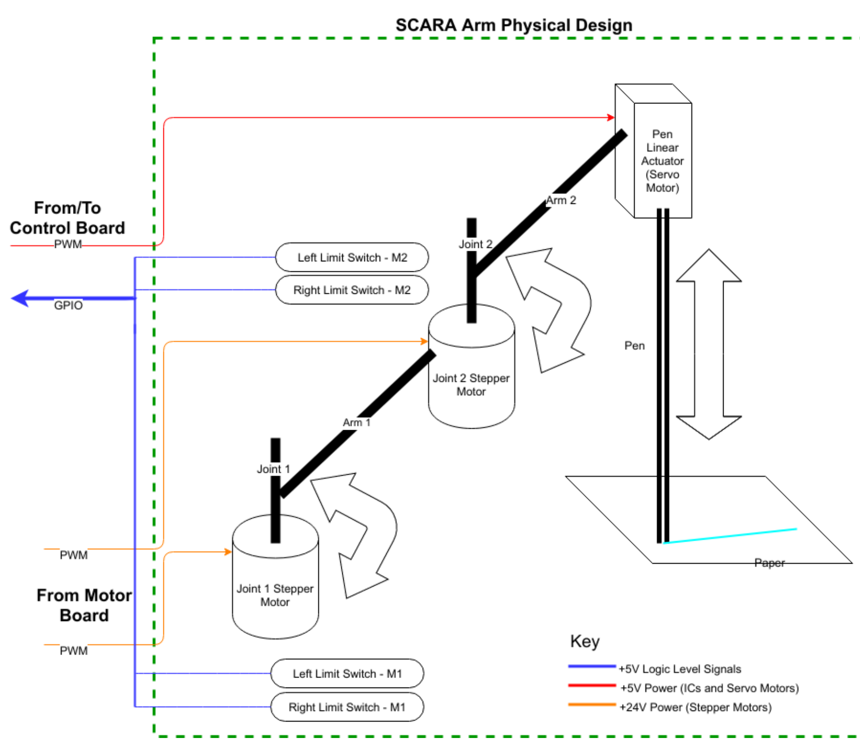


Figure 4: SCARA Robotic Arm Block Diagram

2.3 Functional Overview and Block Requirements

2.3.1 SCARA Robotic Arm (15 Points)

The SCARA arm will be responsible for translating the arm angles that the PC program outputs into physical motion resulting in a visible output on the paper. This requires both high positioning velocity as well as accuracy, which we enable via a stable mechanical design that bears the load with bearing in place of the motor shaft and the accuracy and high holding torque of stepper motors.

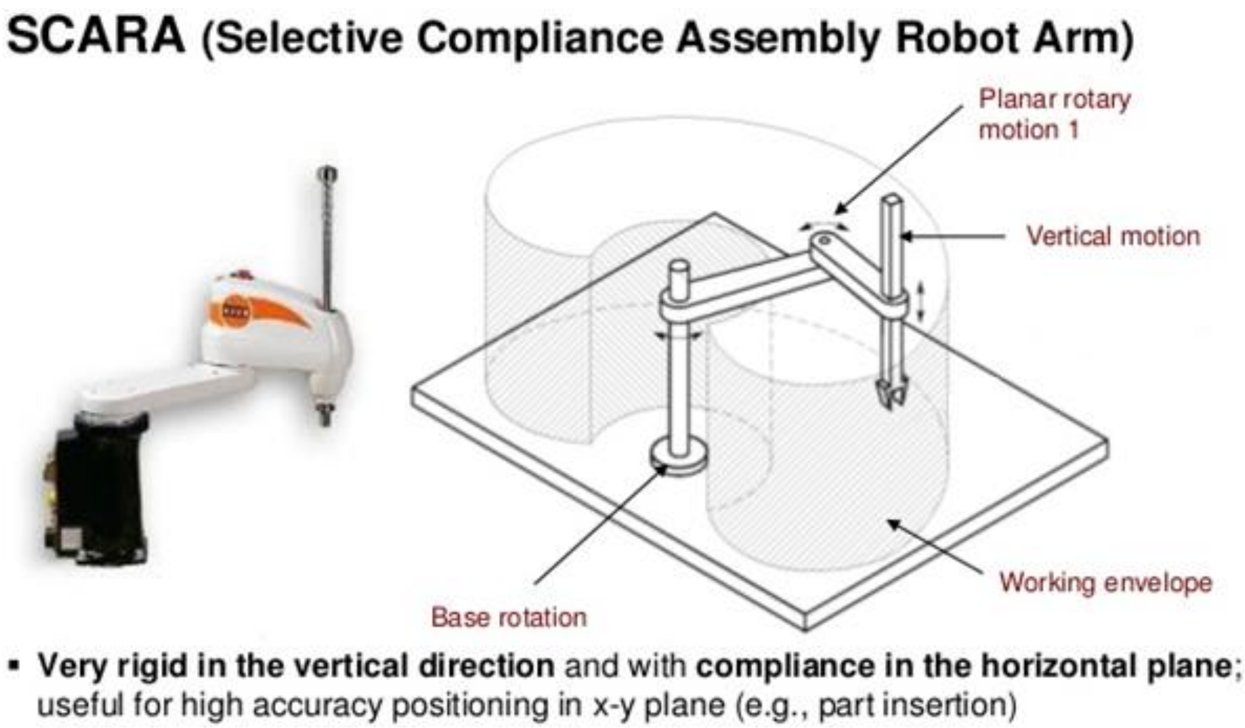


Figure 5: SCARA Robotic Arm Workspace

As the workspace of a SCARA robotic arm is a distorted cardioid-like shape, we plan to have each arm segment be eight inches long, such that a piece of Letter-sized paper fits within the workspace of the robotic arm. This is since we can approximate the main lobe of the cardioid by a semicircle, such that the following diagram holds:

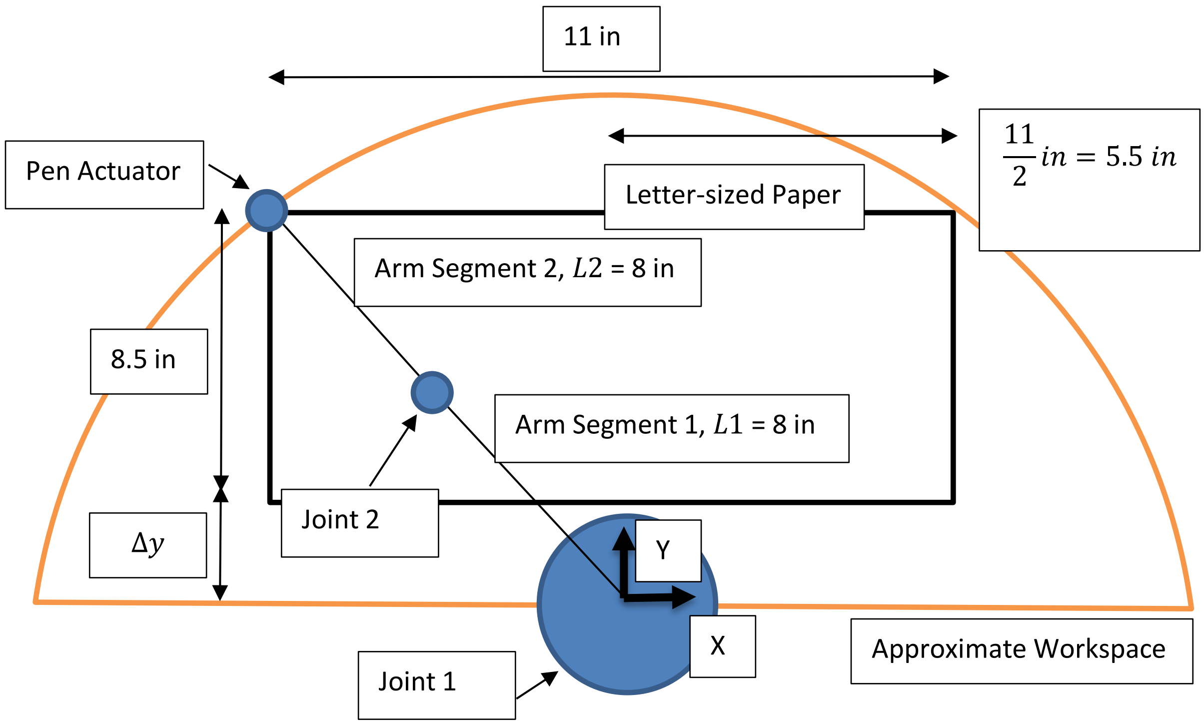


Figure 6: SCARA Robotic Arm Workspace Calculations

The robot's inverse kinematics are defined using the following as shown in Figure 7 below, where E is the elbow angle, the angle of Joint 2, S is the shoulder angle, the angle of Joint 1, $L1$ is the length of arm segment 1, and $L2$ is the length of arm segment 2. Assuming the base of the robot is located at the origin, given the desired (X, Y) coordinates, we can compute the needed Joint 1 angle and Joint 2 angle such the pen can be moved to the desired (X, Y) coordinates.

$$E = \cos^{-1}\left(\frac{X^2+Y^2-(L1)^2-(L2)^2}{2(L1)(L2)}\right) \tag{1}$$

$$S = atan2(Y, X) - \cos^{-1}\left(\frac{X^2+Y^2+(L1)^2-(L2)^2}{2(L1)\sqrt{X^2+Y^2}}\right) \tag{2}$$

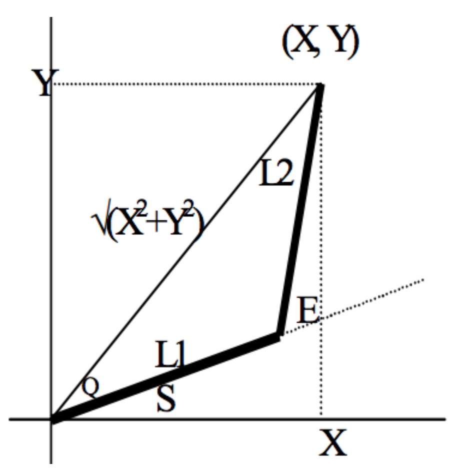


Figure 7: Inverse Kinematics for the SCARA Robotic Arm

We define $atan2(y,x)$ as

$$atan2(y,x) = \begin{cases} \tan^{-1}\left(\frac{y}{x}\right) & x > 0 \\ \tan^{-1}\left(\frac{y}{x}\right) + \pi & x < 0, y \geq 0 \\ \tan^{-1}\left(\frac{y}{x}\right) - \pi & x < 0, y < 0 \\ \frac{\pi}{2} & x = 0, y > 0 \\ -\frac{\pi}{2} & x = 0, y < 0 \\ undefined & x = 0, y = 0 \end{cases} \quad (3)$$

Suppose we place our piece of letter shaped paper as in Figure 6, where the base of robot, and by extension, Joint 1, is located at the origin of the coordinate system. From the above diagram, we have

$$L1 = L2 = 8\ in \quad (4)$$

By the Pythagorean theorem,

$$\Delta y = \sqrt{(8\ in + 8\ in)^2 - (5.5\ in)^2} - 8.5\ in = 6.5\ in \quad (5)$$

Hence, we can bound X and Y, since the paper defines a rectangular domain on the $X - Y$ plane. We have

$$-5.5\ in \leq X \leq 5.5\ in \quad (6)$$

$$6.5\ in = \Delta y \leq Y \leq 8.5\ in + \Delta y = 8.5\ in + 6.5\ \text{in} = 15\ \text{in} \quad (7)$$

Now, this implies that we have

$$0\ in^2 \leq X^2 \leq 30.25\ in^2 \quad (8)$$

$$42.25\ in^2 \leq Y^2 \leq 225\ \text{in}^2 \quad (9)$$

and from the subexpression to calculate Joint 2 angle

$$\frac{X^2+Y^2-(L1)^2-(L2)^2}{2(L1)(L2)} \quad (10)$$

we have, for possible boundary value of the subexpression

$$\frac{0\ in^2+42.25\ in^2-(8\ in)^2-(8\ in)^2}{2(8\ in)(8\ in)} = -0.670 \quad (11)$$

$$\frac{30.25\ in^2+42.25\ in^2-(8\ in)^2-(8\ in)^2}{2(8\ in)(8\ in)} = -0.434 \quad (12)$$

$$\frac{0\ in^2+225\ in^2-(8\ in)^2-(8\ in)^2}{2(8\ in)(8\ in)} = 0.758 \quad (13)$$

$$\frac{30.25\ in^2+225\ in^2-(8\ in)^2-(8\ in)^2}{2(8\ in)(8\ in)} = 0.994 \quad (14)$$

Hence the above subexpression, Equation 9, is bounded as follows, as $X^2 + Y^2$ gives a function that does not change in concavity and has no saddle points

$$-0.670 \leq \frac{X^2+Y^2-(L1)^2-(L2)^2}{2(L1)(L2)} \leq 0.994 \quad (15)$$

Since the domain of $\cos^{-1}(x)$ is $x \in [-1, 1]$, we see that for all points on the paper, there is a solution to the Joint 2 angle. Similarly, for the Joint 1 angle, we have the subexpression

$$\frac{X^2+Y^2+(L1)^2-(L2)^2}{2(L1)\sqrt{X^2+Y^2}} = \frac{X^2+Y^2+(8\ in)^2-(8\ in)^2}{2(8\ in)\sqrt{X^2+Y^2}} = \frac{1}{16\ in}\sqrt{X^2+Y^2} \quad (16)$$

we have, for possible boundary value of the subexpression

$$\frac{0\ in^2+42.25\ in^2+(8\ in)^2-(8\ in)^2}{2(8\ in)\sqrt{0\ in^2+42.25\ in^2}} = 0.406 \quad (17)$$

$$\frac{30.25\ in^2+42.25\ in^2+(8\ in)^2-(8\ in)^2}{2(8\ in)\sqrt{30.25\ in^2+42.25\ in^2}} = 0.532 \quad (18)$$

$$\frac{0\ in^2+225\ in^2+(8\ in)^2-(8\ in)^2}{2(8\ in)\sqrt{0\ in^2+225\ in^2}} = 0.938 \quad (19)$$

$$\frac{30.25\ in^2+225\ in^2+(8\ in)^2-(8\ in)^2}{2(8\ in)\sqrt{30.25\ in^2+225\ in^2}} = 0.999 \quad (20)$$

Hence the above subexpression, Equation 9, is bounded as follows, as $\sqrt{X^2 + Y^2}$ gives a function that does not change in concavity and has no saddle points

$$0.406 \leq \frac{X^2+Y^2-(L1)^2-(L2)^2}{2(L1)(L2)} \leq 0.999 \quad (21)$$

Since the domain of $\cos^{-1}(x)$ is $x \in [-1, 1]$ and the domain of $atan2(y, x)$ is all values of x and y except $x = 0, y = 0$, we see that for all points on the paper, there is a solution to the Joint 1 angle.

Hence, we see that with each arm segment being eight inches long, there is sufficient space to place centrally in a relatively easy to reach portion of the workspace a piece of Letter-sized paper, provided the piece of Letter-sized paper is placed as discussed above. Meanwhile, as the pen actuator only needs a very small amount of z-travel on the order of millimeters, we can approximate our pen actuator as a linear actuator via the small angle approximation for sine, and as we are using a Cartesian coordinate system, this implies that our robotic should be able to reach any desired value of z as long as the value of z is such that the small angle approximation for sine is valid, since both the axis and the joint are of the same nature, that is, linearly varying.

To determine the stepper motor torque needed, we researched some industrial robotic arms and decided to target a joint velocity of at least $100\ °/s$, reached in $2s$ or less from zero angular velocity. Assuming that we start from zero angular velocity and do not attempt to brake after reaching the target velocity, we have, where ω is the angular velocity about the rotational axis and α is the angular acceleration about the rotational axis

$$\omega = \alpha t \tag{22}$$

Since torque is

$$\tau = I\alpha \tag{23}$$

where I is the moment of inertia about the desired rotational axis, we have

$$\tau = \frac{I\omega}{t} \tag{24}$$

From Autodesk Inventor, the rotational inertia of the arm segment about the rotational axis is $I = 9.845\ lb \cdot in$. Thus, the required torque is

$$\tau = \frac{9.845\ lb{\cdot}in{\cdot}100\frac{deg}{s}\cdot\frac{2\pi}{360\ deg}}{2\ s} \cdot \frac{1\ N}{0.22481\ lb} \cdot \frac{0.0254\ m}{1\ in} = 0.971\ N \cdot m \tag{25}$$

which we round to approximately $1\ N \cdot m$ to allow for some part variation tolerance.

From the simulations below, we can see that the structure as designed has sufficient structural integrity for the anticipated loads and forces, evident from the very low displacement and high safety factor of the structure. To analyze vibration stability, we first convert our targeted joint velocity to frequency f via the following formula, where all other symbols are as defined below, and obtain

$$f = \frac{\omega}{2\pi} = \frac{100\frac{deg}{s}\cdot\frac{2\pi}{360\ deg}}{2\pi} = 0.278\ Hz \tag{26}$$

Hence, modal analysis shows that all resonance frequencies are at least $20Hz$ higher compared to the joint angular velocity we are designing for and thus vibrations will be minimal.

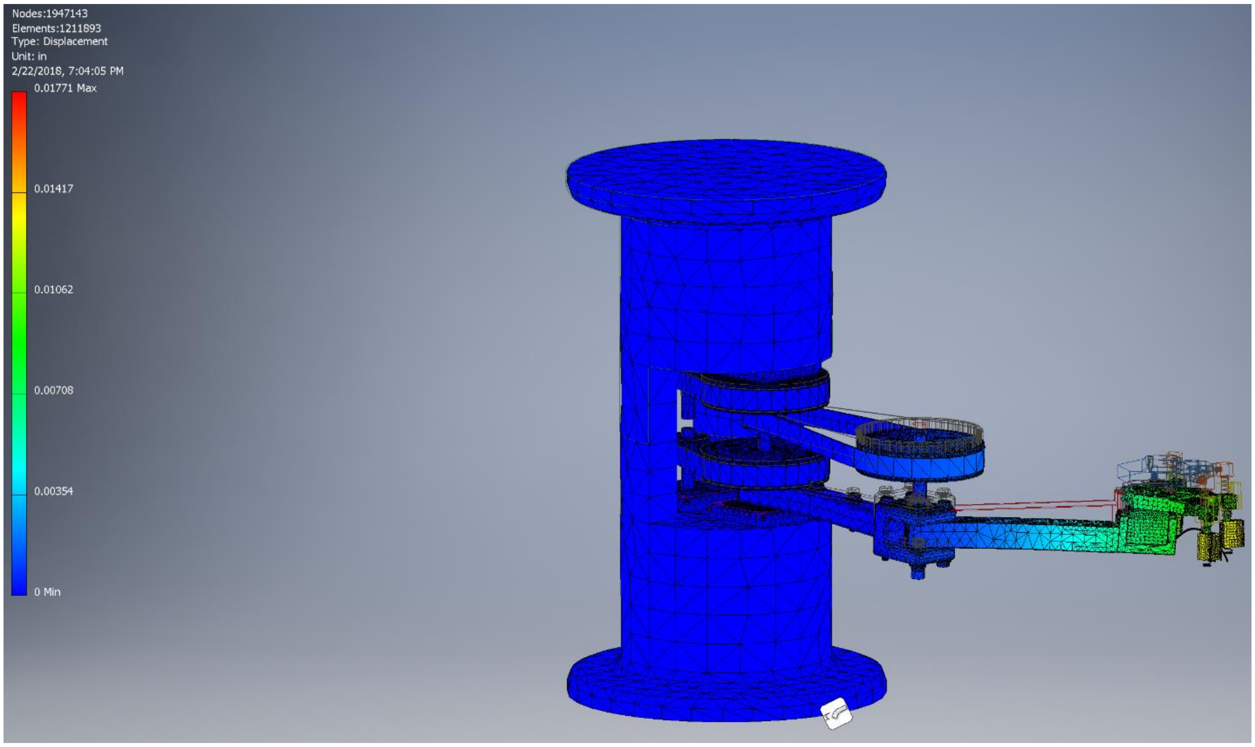


Figure 8: Robot Arm Static Analysis - Displacement

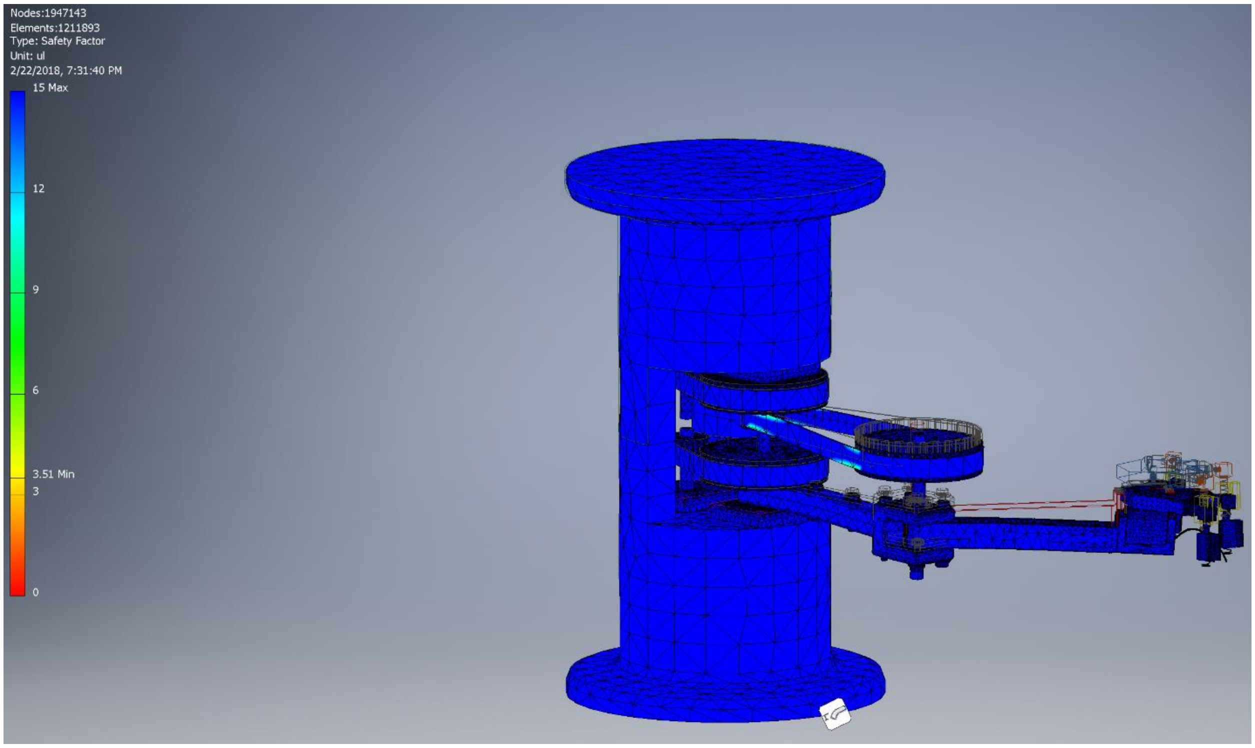


Figure 9: Robot Arm Static Analysis – Safety Factor

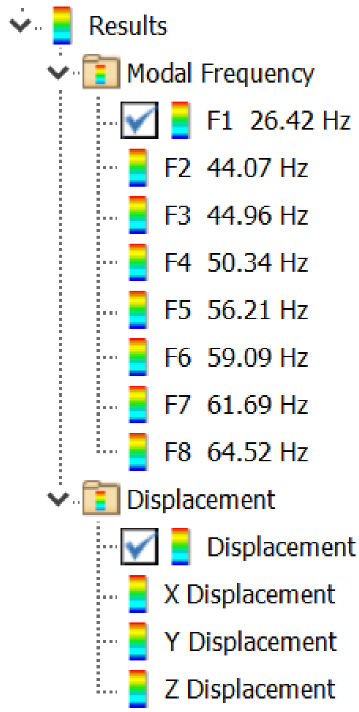


Figure 10: Robot Arm Modal Analysis – List of Modes

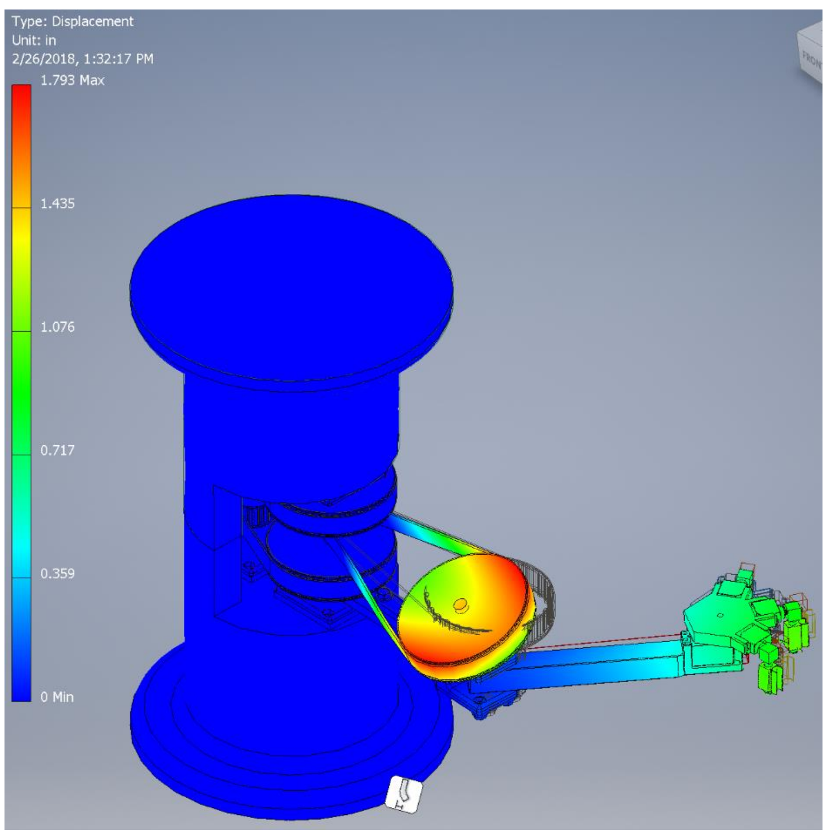


Figure 11: Robot Arm Modal Analysis – Mode 1

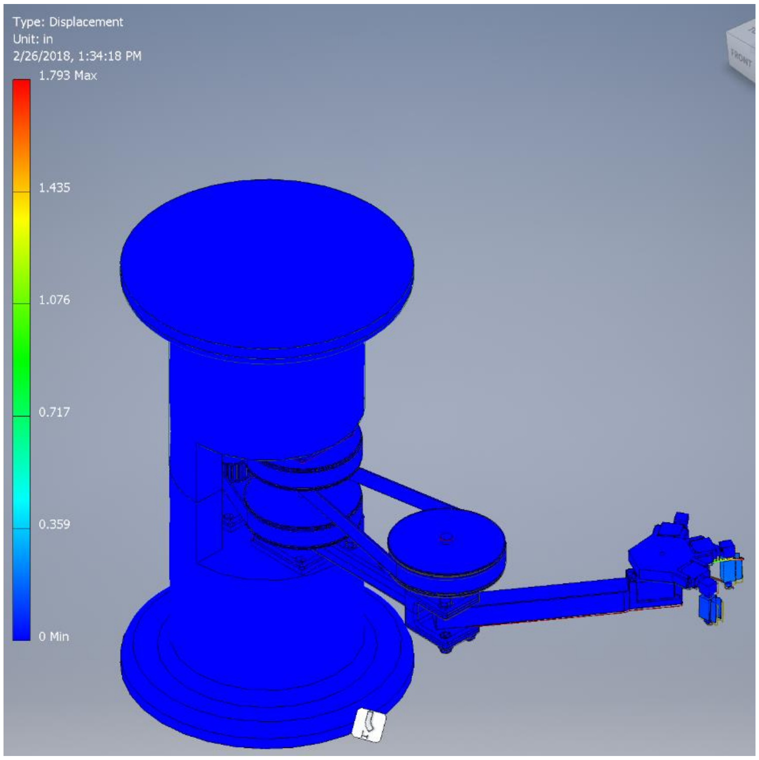


Figure 12: Robot Arm Modal Analysis – Mode 2

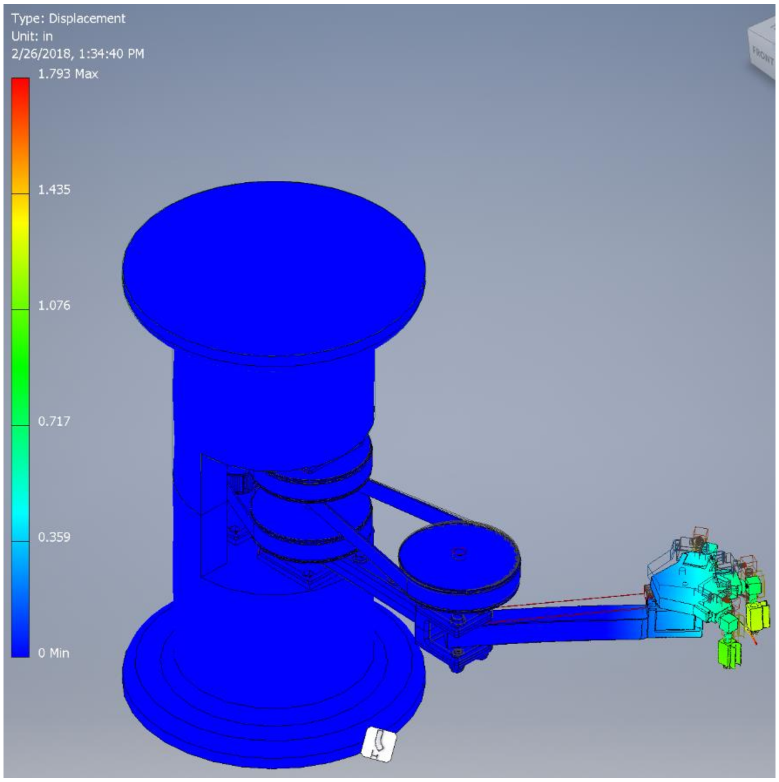


Figure 13: Robot Arm Modal Analysis – Mode 3

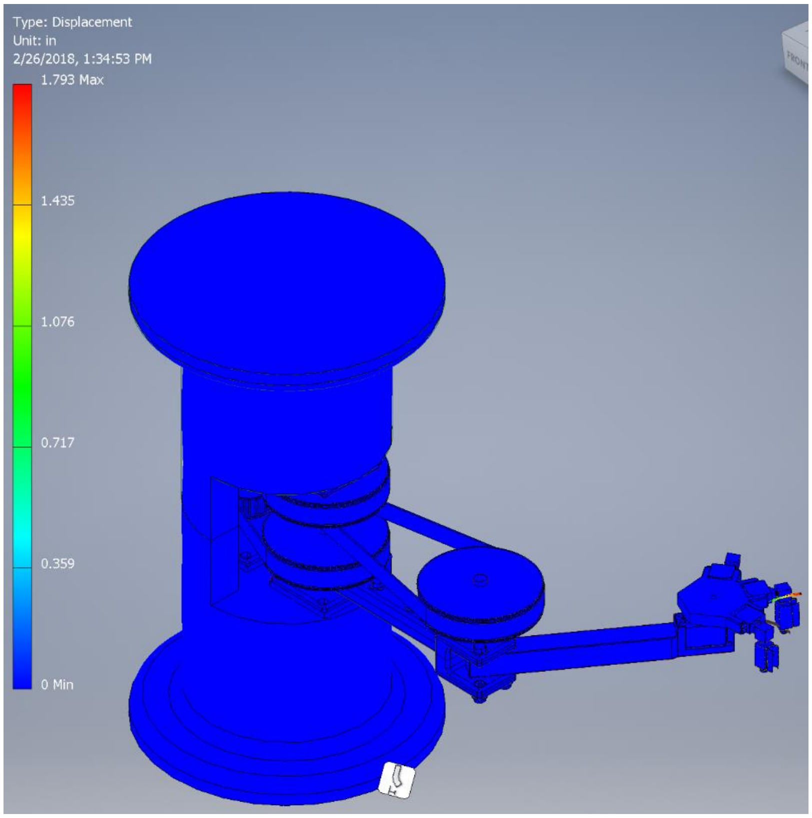


Figure 14: Robot Arm Modal Analysis – Mode 4

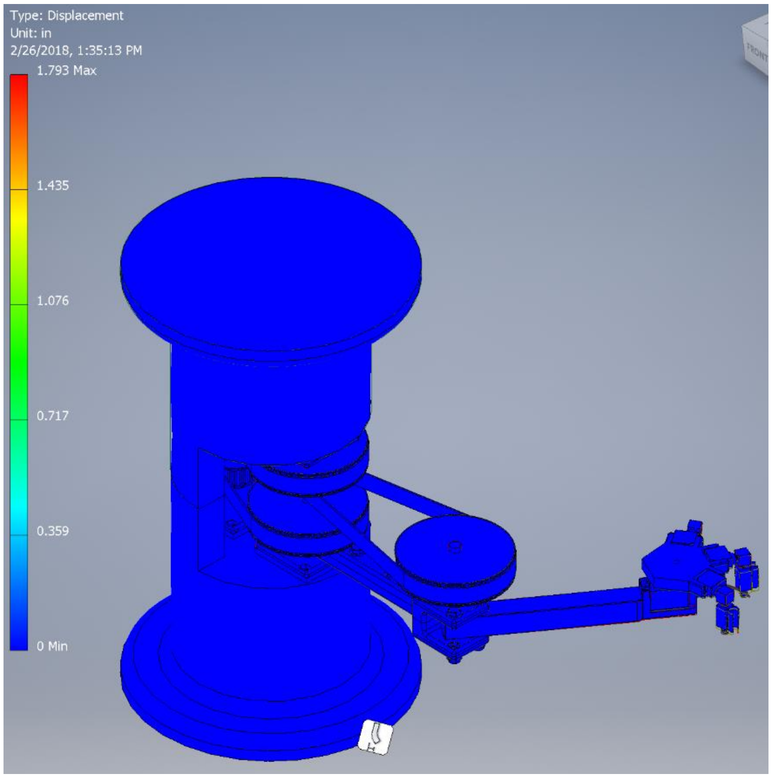


Figure 15: Robot Arm Modal Analysis – Mode 5

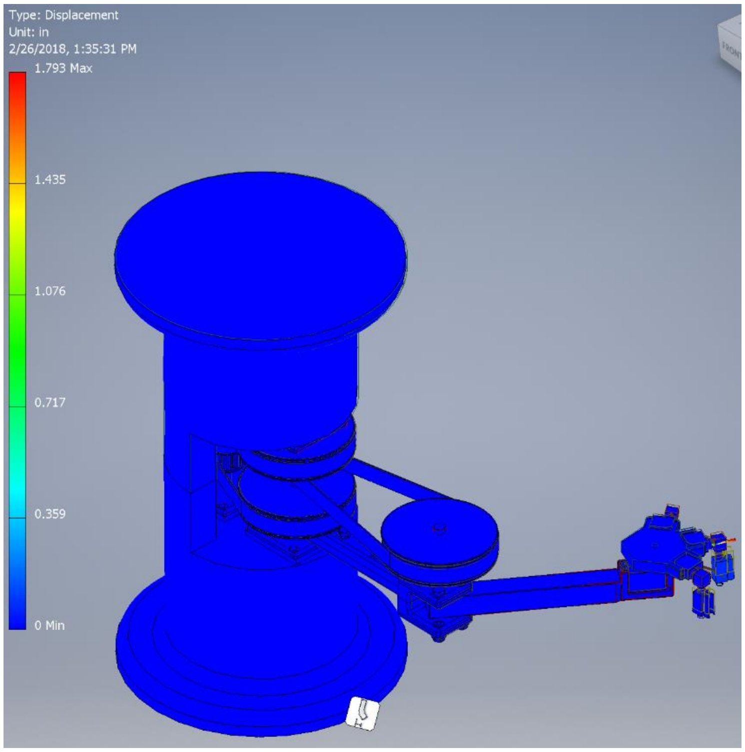


Figure 16: Robot Arm Modal Analysis – Mode 6

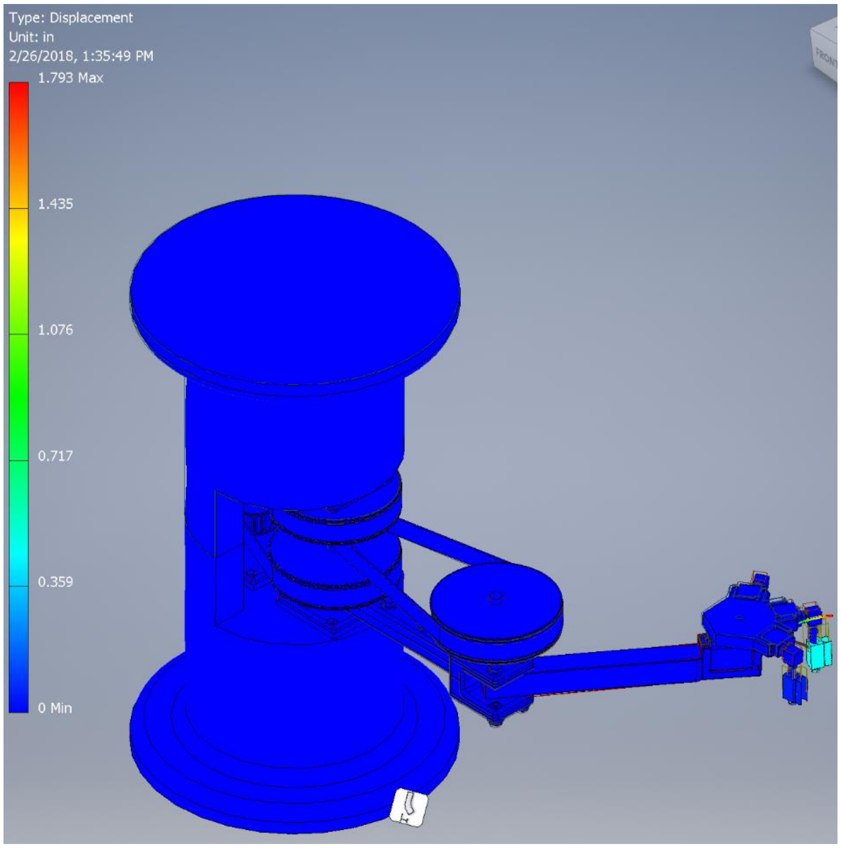


Figure 17: Robot Arm Modal Analysis – Mode 7

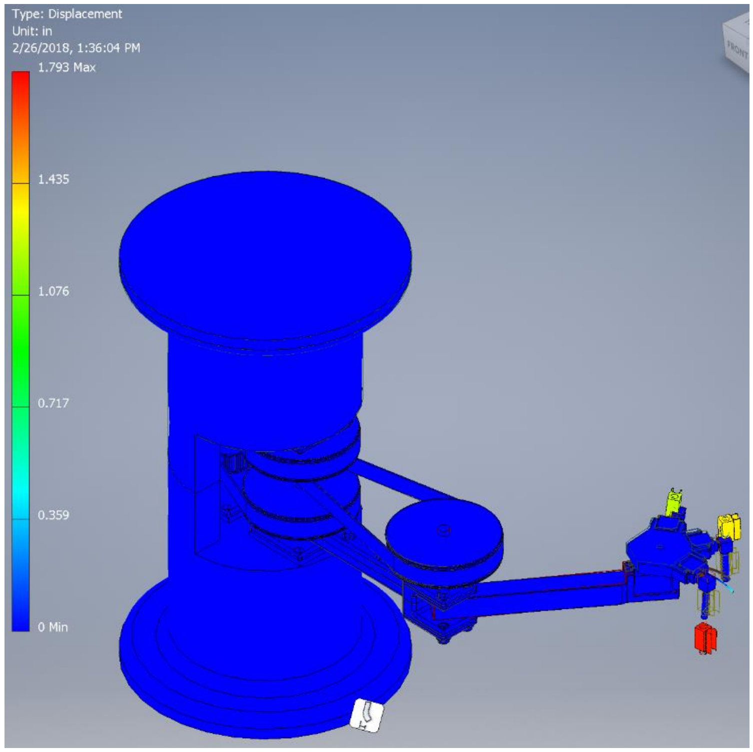


Figure 18: Robot Arm Modal Analysis – Mode 8

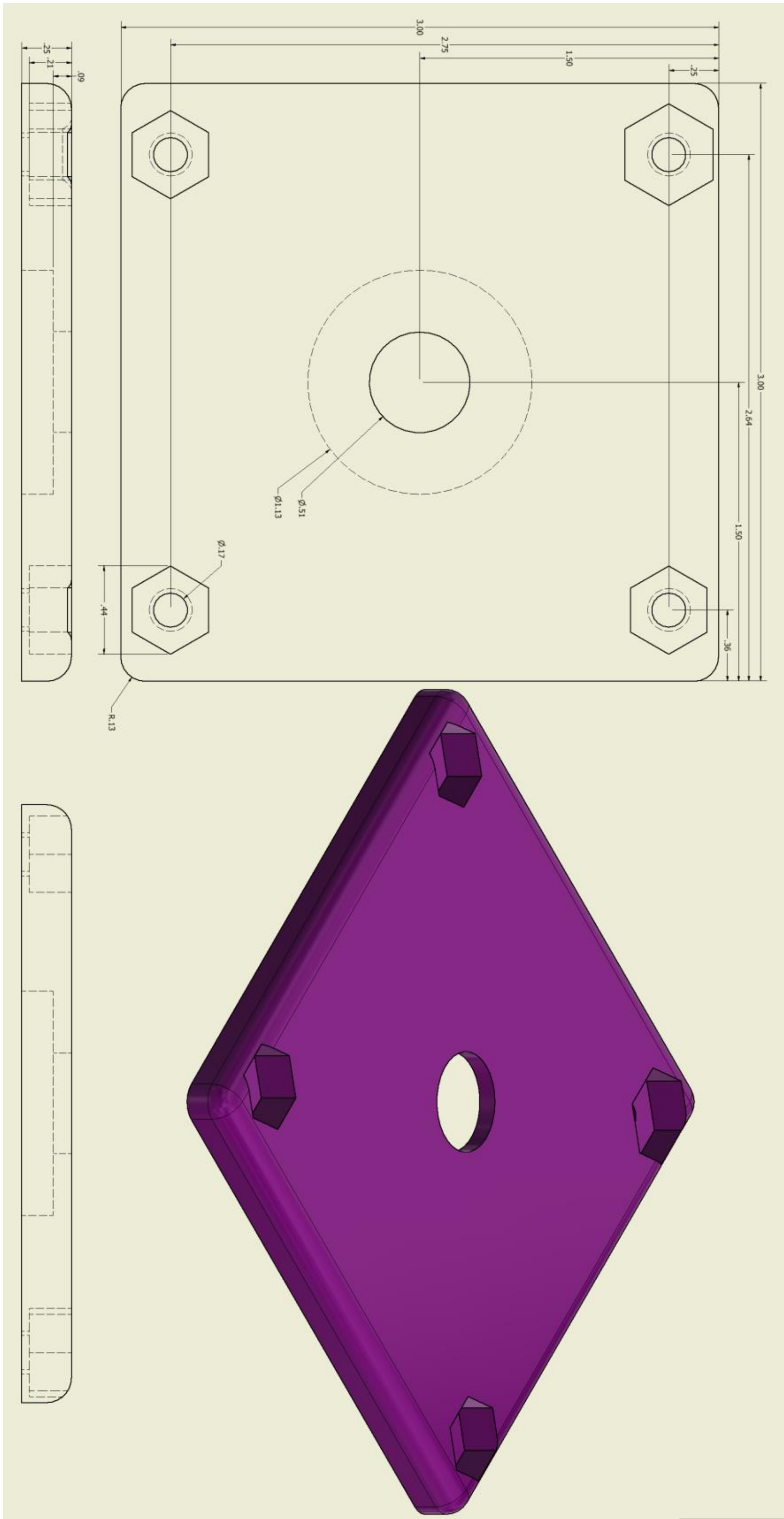


Figure 19: Base Bearing Cover Upper Half Mechanical Drawing (Units: in)

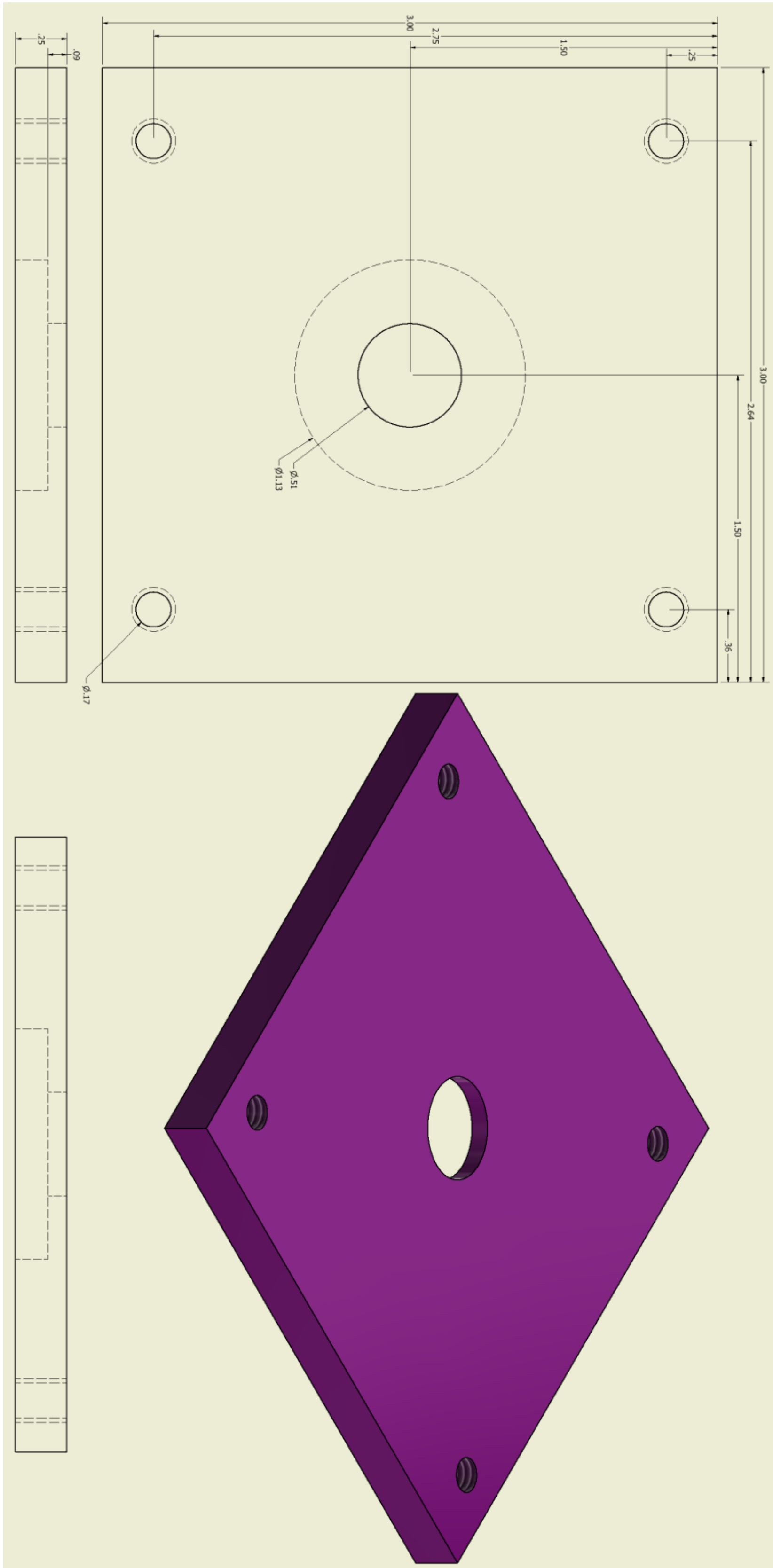


Figure 20: Base Bearing Cover Lower Half Mechanical Drawing (Units: in)

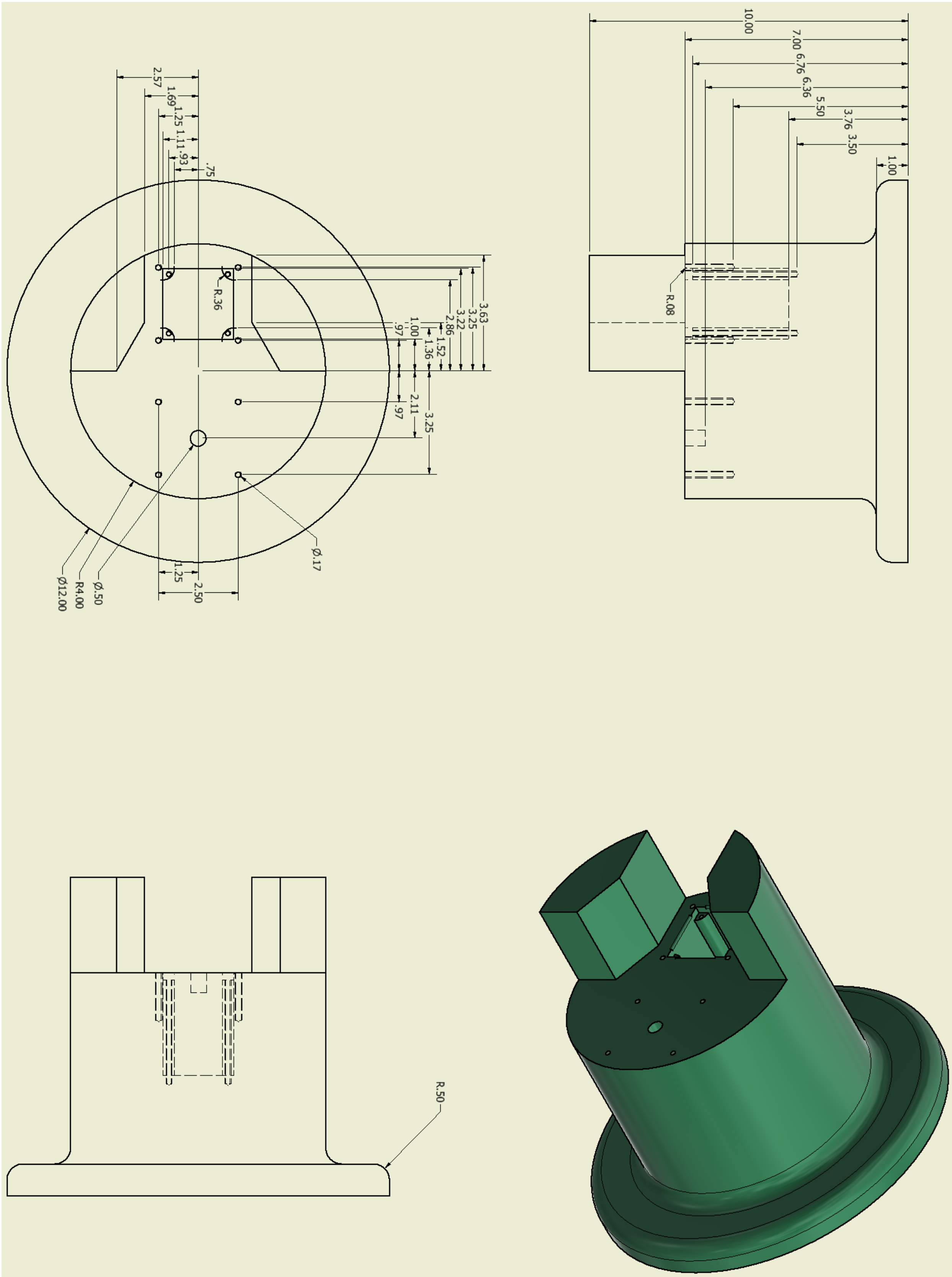
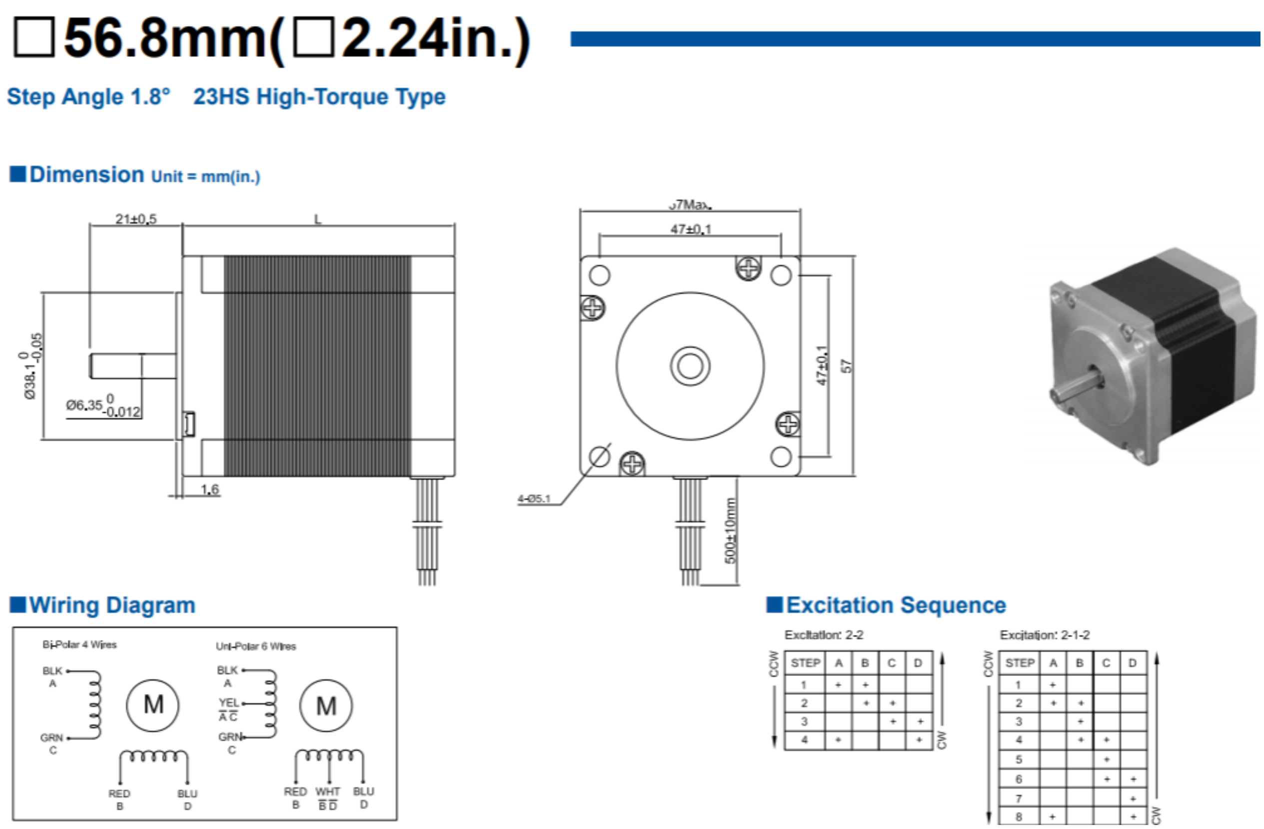


Figure 21: Base Mechanical Drawing (Units: in)

2.3.1.1 Joint Motor 1, Motor 2

These stepper motors move arm segment one and arm segment two respectively. They require high holding torque to ensure no slippage and fast movement. As our mechanical design has the motors mounted to the base with power transmission via timing belts and gears, which means the motors do not themselves need to be moved as part of either arm segment, neither mass nor heat dissipation i.e. current consumption, is a significant concern. Please note that the connections between each arm segment, joint, and linear actuator is detailed above in Section 2.2. To satisfy the torque requirement above, we have picked the 23HS22-2804S, which has the following specifications:



23HS22-2804S	126	178.4	2.5	2.8	0.9	2.5	300	Bi (4)	0.7	56	2.2

Figure 22: 23HS22-2804S Stepper Motor Specification

Table 1: Joint Motor Requirements and Verification Procedures

Requirement	Verification Procedure
1. Torque of at least 1 N-m.	1. Assemble and connect all electronic components and boards together as specified in the block diagram. 2. Mount a stepper motor using the base with additional holes drilled and bolts to a sturdy table. 3. One end of solid aluminum 6063 bar of dimensions 1 m x 0.1 m x 0.1m will be

	attached to the motor shaft such that the length axis of the metal bar is perpendicular to the motor shaft axis. 4. A mass of 1 N will be attached to the other end of the metal bar. 5. Using the PC and Control Board, command the Motor Board rotate the motor such that the mass moves upwards. 6. Gradually increase commanded current, starting from 1A, until the mass moves. 7. If the commanded current exceeds 3A, the test has failed.
2. Inductance of less than 10mH.	1. Connect the RED and BLU wire (See Figure 22 above), each to one terminal of an LCR meter. 2. Connect the BLK and GRN wire (See Figure 22 above), each to one terminal of an LCR meter. 3. Measure inductance with the LCR meter and verify.
3. Have standard NEMA-type mounting holes, as shown in Figure 23 below.	1. The relevant measurements' correctness will be confirmed by caliper measurement of the relevant distance between the indicated points, using a with a Husky 6 in Digital Fractional Caliper with a measurement accuracy of +/- 0.001 in.

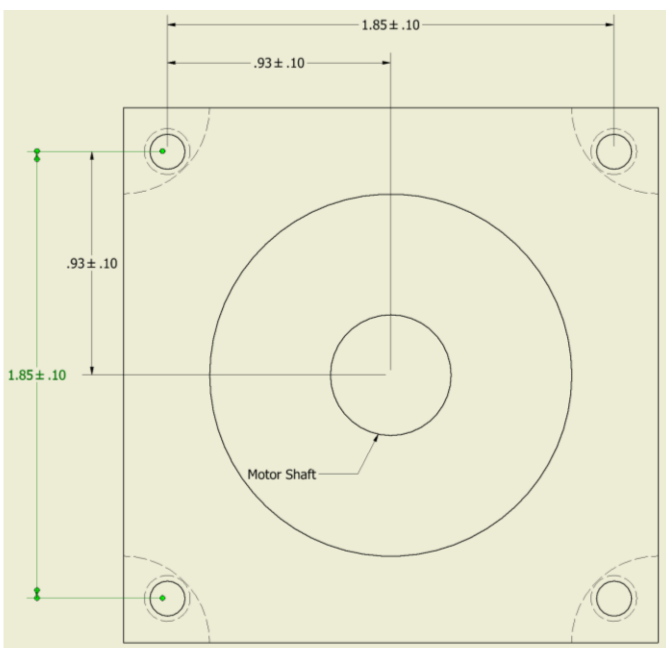


Figure 23: NEMA-type Mounting Hole Pattern (Units: in)

2.3.1.2 Pen Actuator (5 Points)

This actuator rotates the pen up and down onto the paper to enable drawing. While high speed is preferred to allow for fast drawing of certain styles e.g. points, very little travel is required. The actuator needs to prevent compliance in the x-y plane, but should be somewhat compliant in the z-axis to prevent breaking the pen or scratching the paper at high arm travel velocities. The linear actuator is controlled by a sub-micro servo responding to appropriate PWM waveforms generated by the microcontroller in the control board. Unlike the arm joint motors, PID control is integrated within the servo, and hence a control loop running on the microcontroller is not required for this linear actuator. Please note that the connections between each arm segment, joint, and linear actuator is detailed above in Section 2.2.

Table 2: Pen Actuator Requirements and Verification Procedures

Requirement	Verification Procedure
1. Actuation travel of 45 degrees +/- 5%.	1. Before manufacturing, the relevant measurement's correctness will be confirmed via applying the Measure tool to the relevant 3D model using Autodesk Inventor. 2. Assemble actuator. 3. After manufacturing, the relevant measurement's correctness will be confirmed by manually moving the relevant parts and caliper measurement of the relevant distance between the indicated points, using a with a Husky 6 in Digital Fractional Caliper with a measurement accuracy of +/- 0.001 in.
2. Actuation speed of 1s +/- 10%, as measured starting from the paper contacting position to the fully-lifted position, and vice versa.	1. Assemble all physical components and connect all electronic components and boards together as specified in the block diagram. 2. Install a camera such that the actuator is clearly visible. 3. Using the PC, command the actuator to move as required. 4. Use the camera recording to determine the delay.
3. Maximum load of 0.5 lb. +/- 5%.	1. Before manufacturing, the relevant measurement's correctness will be confirmed via applying static simulations to the relevant 3D model using Autodesk Inventor. 2. Assemble arm and add load as needed. 3. Confirm arm does not break or show signs of irreversible damage, such as visible stress patterns.

4. Suspension system enabling a z-axis compliance of 0.25 in +/- 5%.	1. Before manufacturing, the relevant measurement's correctness will be confirmed via applying the Measure tool to the relevant 3D model using Autodesk Inventor. 2. Assemble actuator. 3. After manufacturing, the relevant measurement's correctness will be confirmed by manually moving the relevant parts and caliper measurement of the relevant distance between the indicated points, using a with a Husky 6 in Digital Fractional Caliper with a measurement accuracy of +/- 0.001 in.

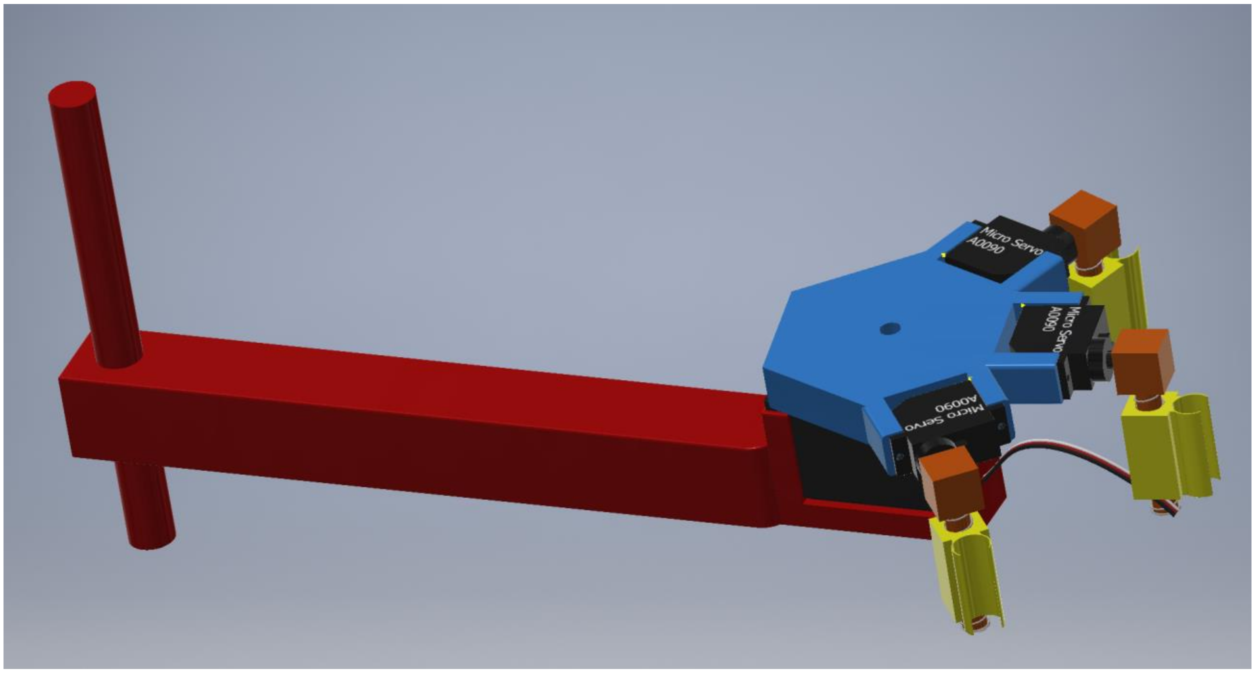


Figure 24: SCARA Robot Arm Segment 2 and Pen Actuator/Pen Holder 3D Model

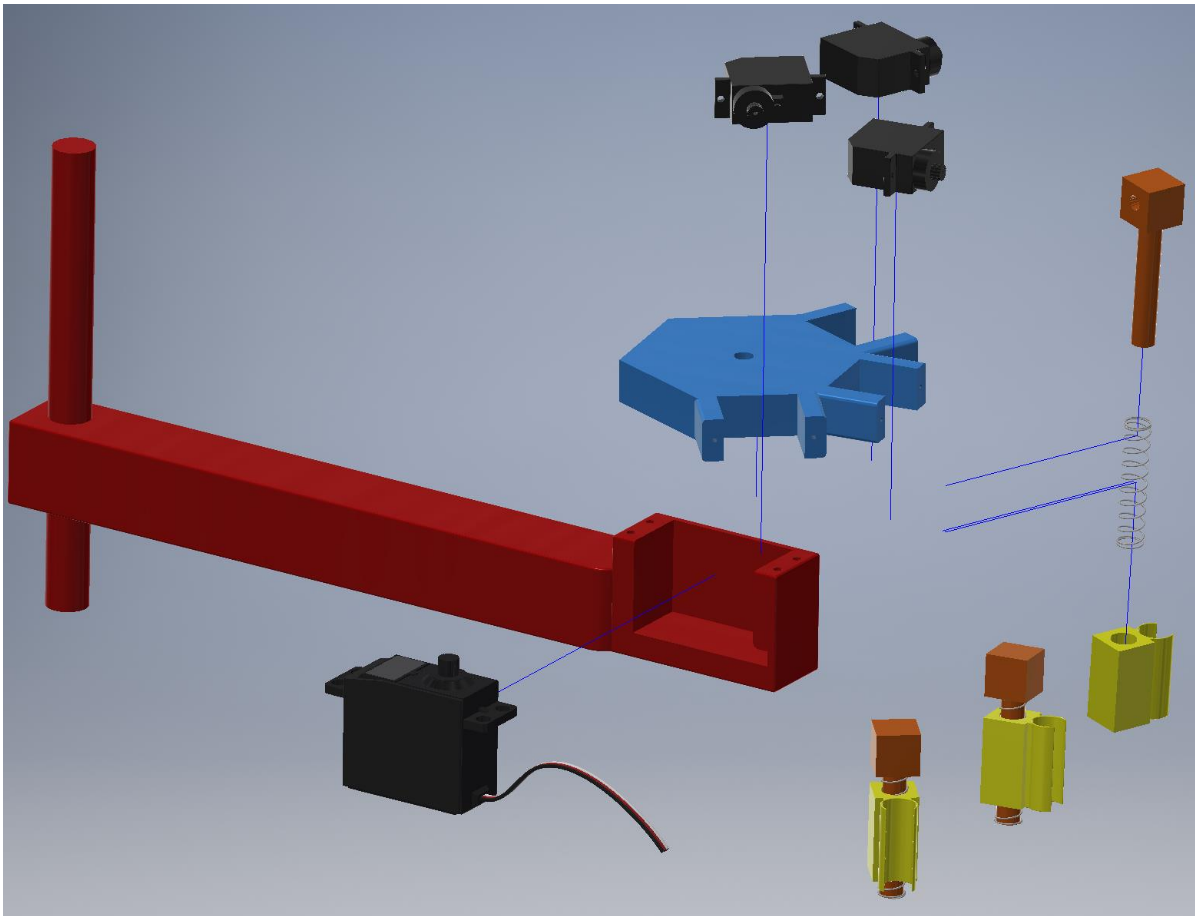


Figure 25: SCARA Robot Arm Segment 2 and Pen Actuator/Pen Holder Exploded View

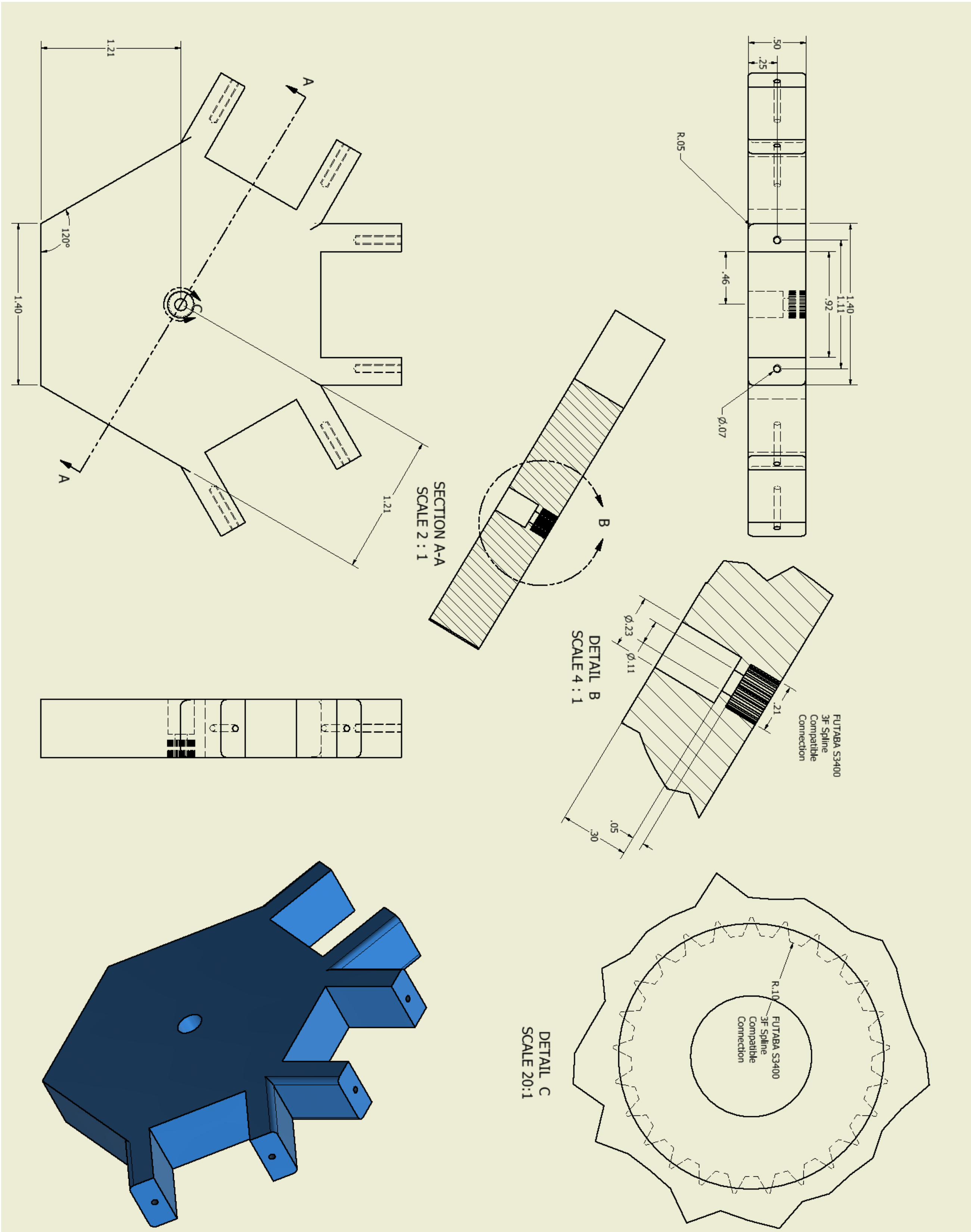


Figure 26: Pen Holder Rotational Platform Mechanical Drawing (Units: in)

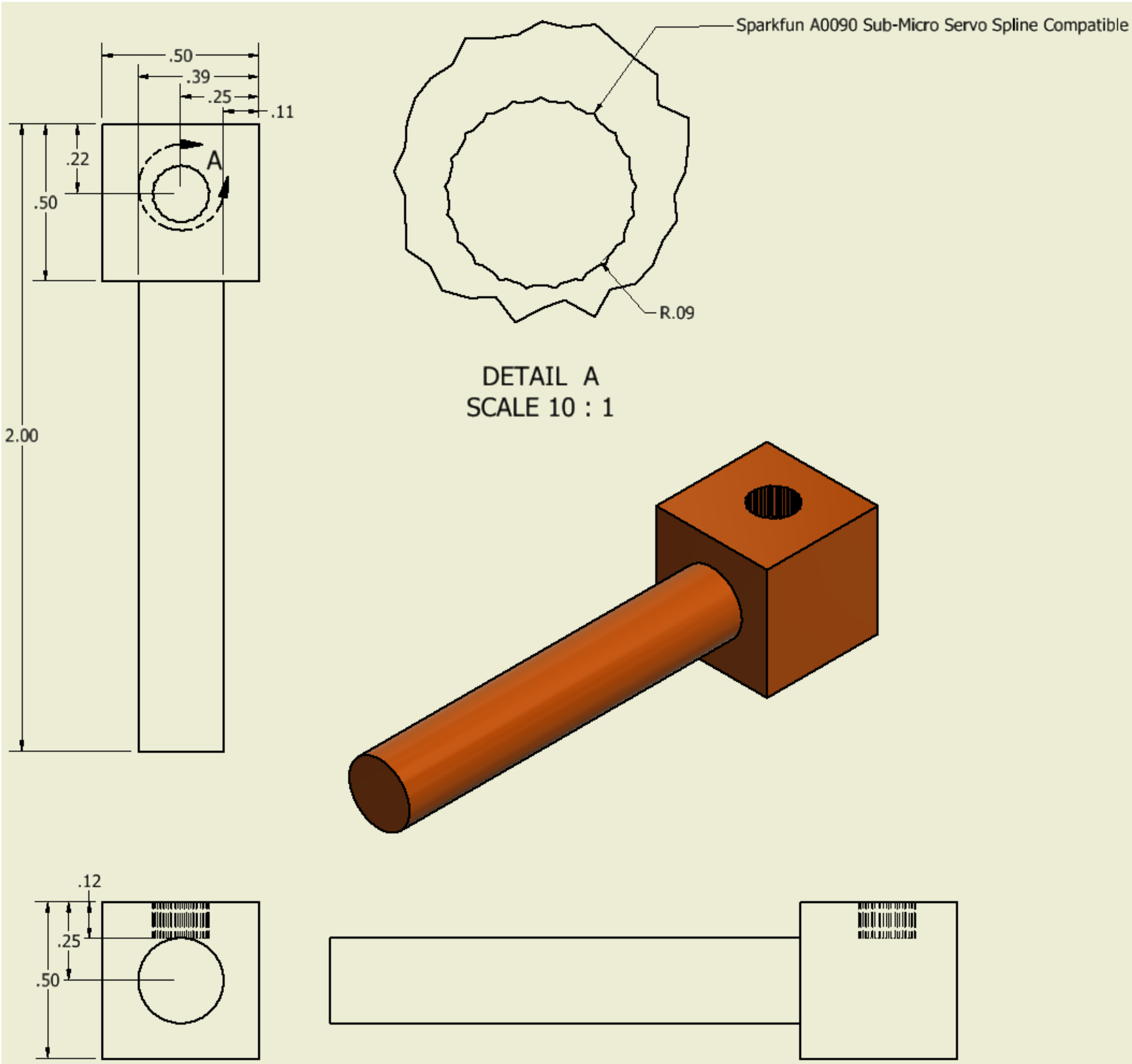


Figure 27: Pen Holder Sub-Micro Servo Attachment Mechanical Drawing (Units: in)

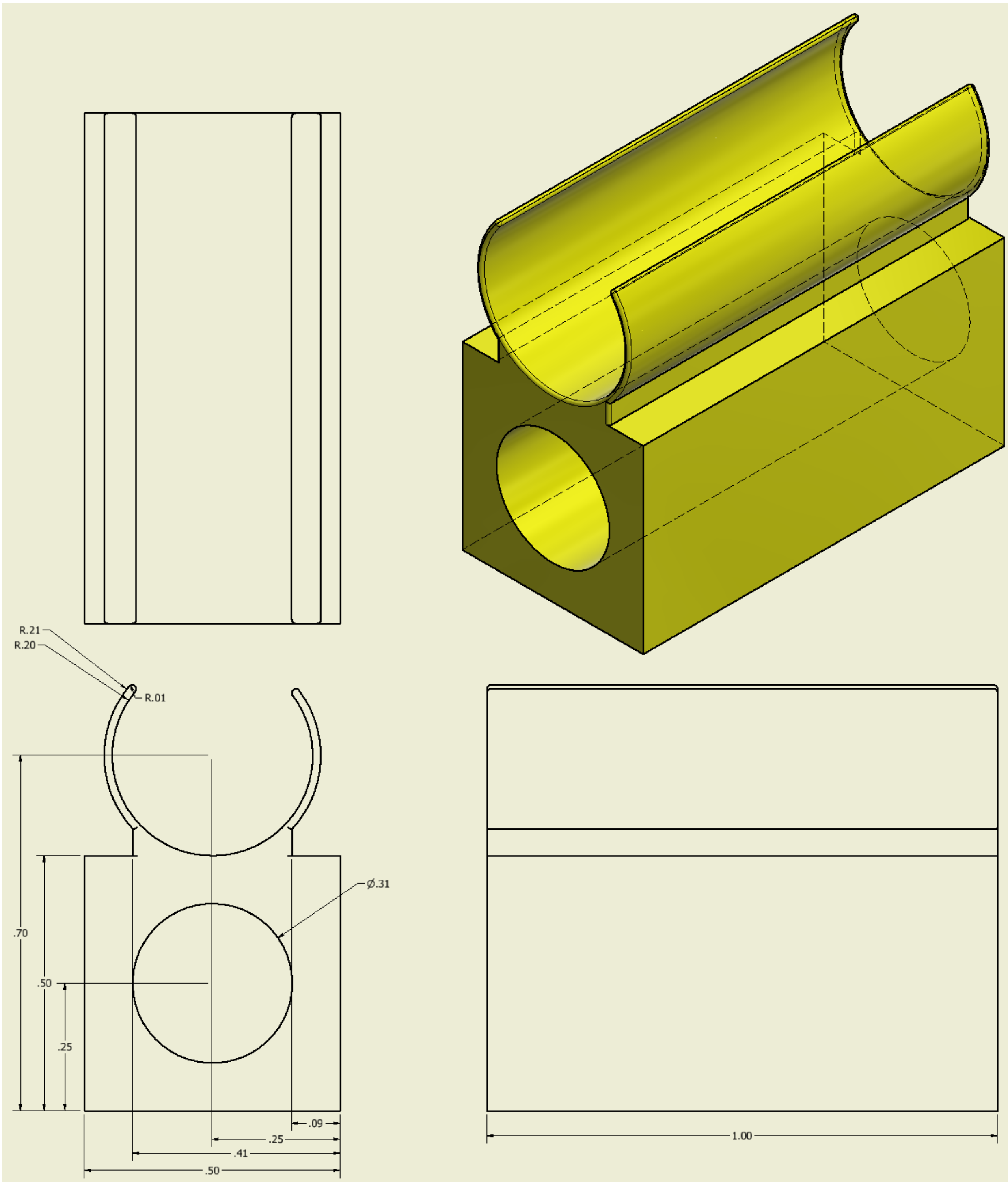


Figure 28: Pen Holder Clip Mechanical Drawing (Units: in)

2.3.1.3 Arm Segment/Joint (10 Points)

These arm segments form the actual robot arm, and necessitates a stable mechanical design that bears the load with bearing in place of the motor shaft to ensure positioning accuracy and speed. They must have very small deflection under full scale load to prevent vibrations and resonance that impair accuracy, and be of minimal mass to avoid loading down the joint motors. The connections between each arm segment, joint, and linear actuator is detailed above in Section 2.2. Please note that these components have been grouped together as our fabrication process is expected to arm segments and joints as integrated units, rather than as separate units.

Table 3: Arm Segment/Joint Requirements and Verification Procedures

Requirement	Verification Procedure
1. Each arm segment should have a length of 5 in +/- 10%, as measured between the rotational axis of Joint 1 and Joint 2 for Arm Segment 1, and as measured between the rotational axis of Joint 2 and the movement axis of the Pen Actuator for Arm Segment 2.	1. Before manufacturing, the relevant measurement's correctness will be confirmed via applying the Measure tool to the relevant 3D model using Autodesk Inventor. 2. After manufacturing, the relevant measurement's correctness will be confirmed by caliper measurement of the relevant distance between the indicated points, using a Husky 6 in Digital Fractional Caliper with a measurement accuracy of +/- 0.001 in.
2. Each arm segment and associated joint (including shafts, bearings, sensors, screws, etc.), excluding the motor/end effector associated with each arm segment should weight at most 3 pounds.	1. Relevant parts will be assembled and placed on a scale to determine if the arm segment assembly is overweight.
3. Sufficient structural integrity to ensure deflection with maximum load and under maximum motor acceleration is under 0.1 inches in in any direction.	1. Assemble all physical components and connect all electronic components and boards together as specified in the block diagram. 2. Install cameras such that every axis of the arm segment is visible, together with appropriate checkboard backgrounds to enable measurement. 3. Use a short PC program to send appropriate commands to the Control Board to actuate joint motors at maximum acceleration and measure flex sensor displacement. 4. Reconcile results and account for actual camera placement with model simulation.
4. Able to bear load of at least 2 pounds at end of Arm Segment 2.	1. Assemble all physical components and connect all electronic components and boards together as specified in the block diagram.

	2. Confirm arm does not break or show signs of irreversible damage, such as visible stress patterns.
5. Each revolute joint should enable 270 degrees +/- 20% of freedom about the joint axis.	1. Assemble all physical components and connect all electronic components and boards together as specified in the block diagram. 2. Use a short PC program to send appropriate commands to the Control Board to actuate joint motor counterclockwise until limit switches are triggered. 3. Measure maximum angular displacement via stepper motor IC controller motor angle feedback. 4. Use a short PC program to send appropriate commands to the Control Board to actuate joint motor clockwise until limit switches are triggered. 5. Measure maximum angular displacement via stepper motor IC controller motor angle feedback.

Figure 29: SCARA Robot Arm Segment 1 3D Model

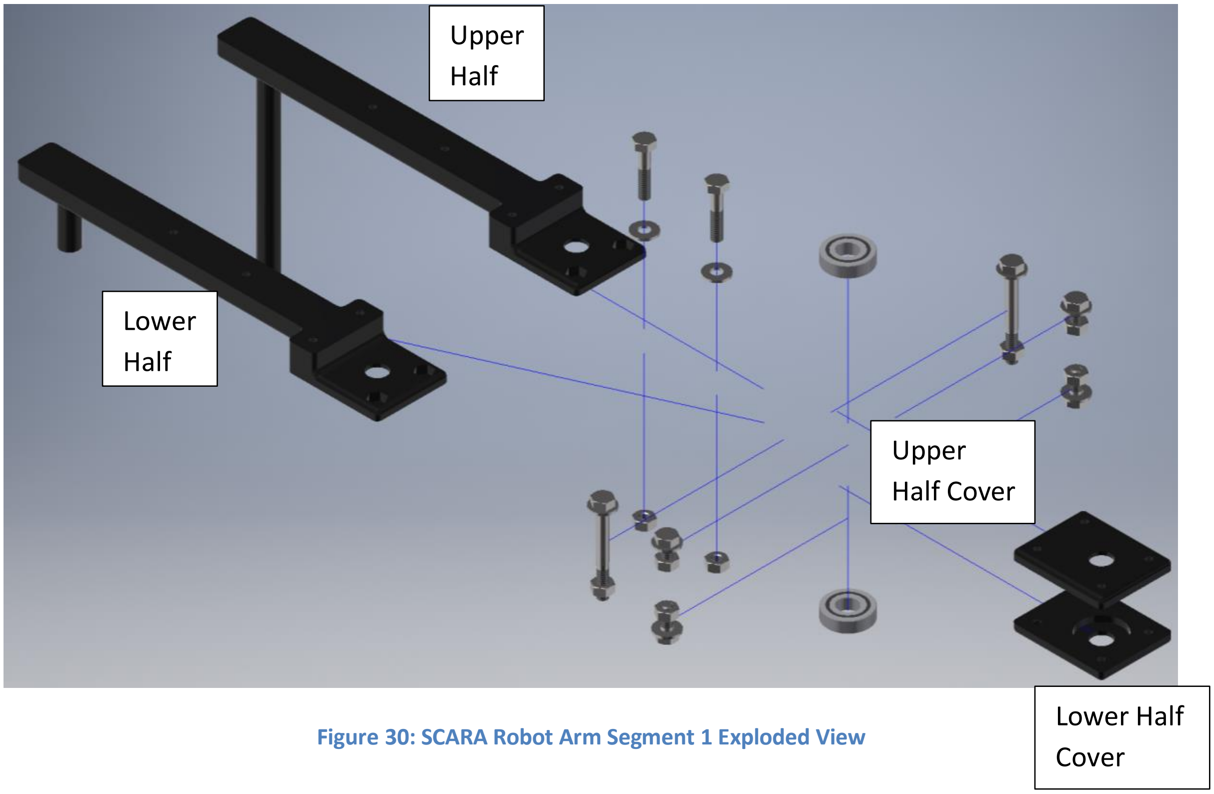


Figure 30: SCARA Robot Arm Segment 1 Exploded View

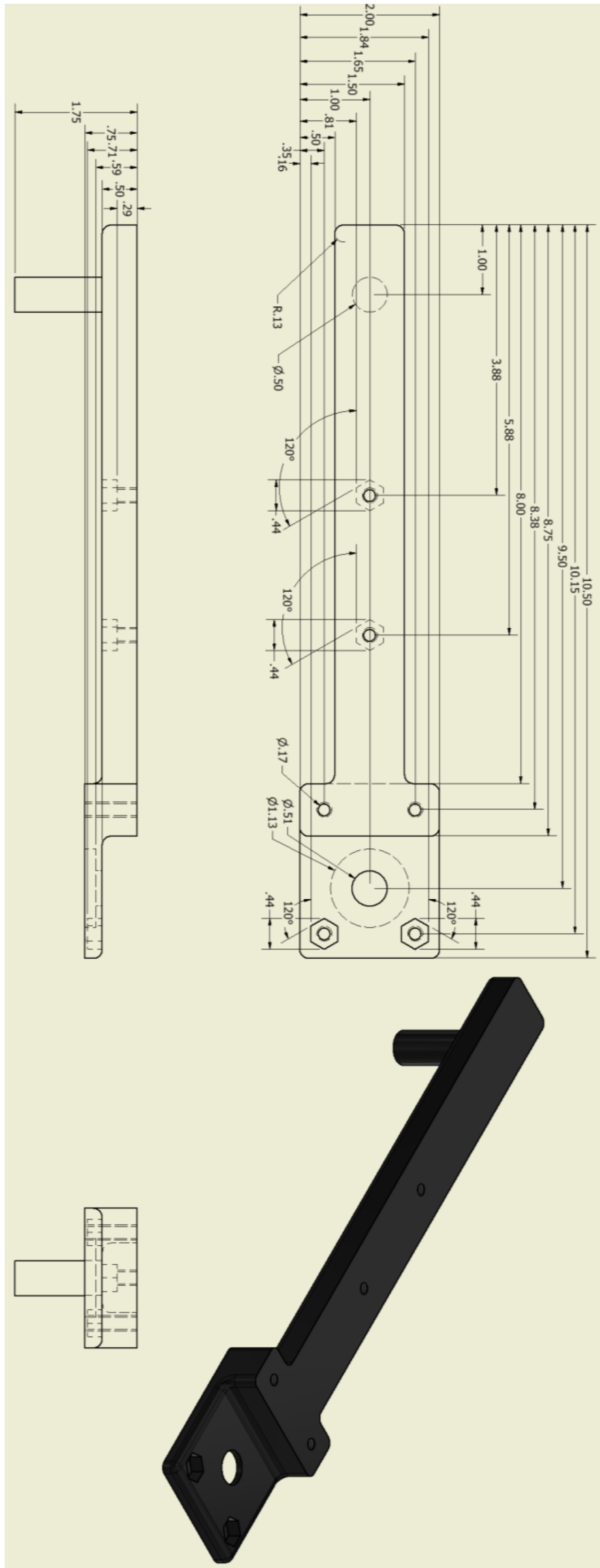


Figure 31: Arm Segment 1 Lower Half Mechanical Drawing (Units: in)

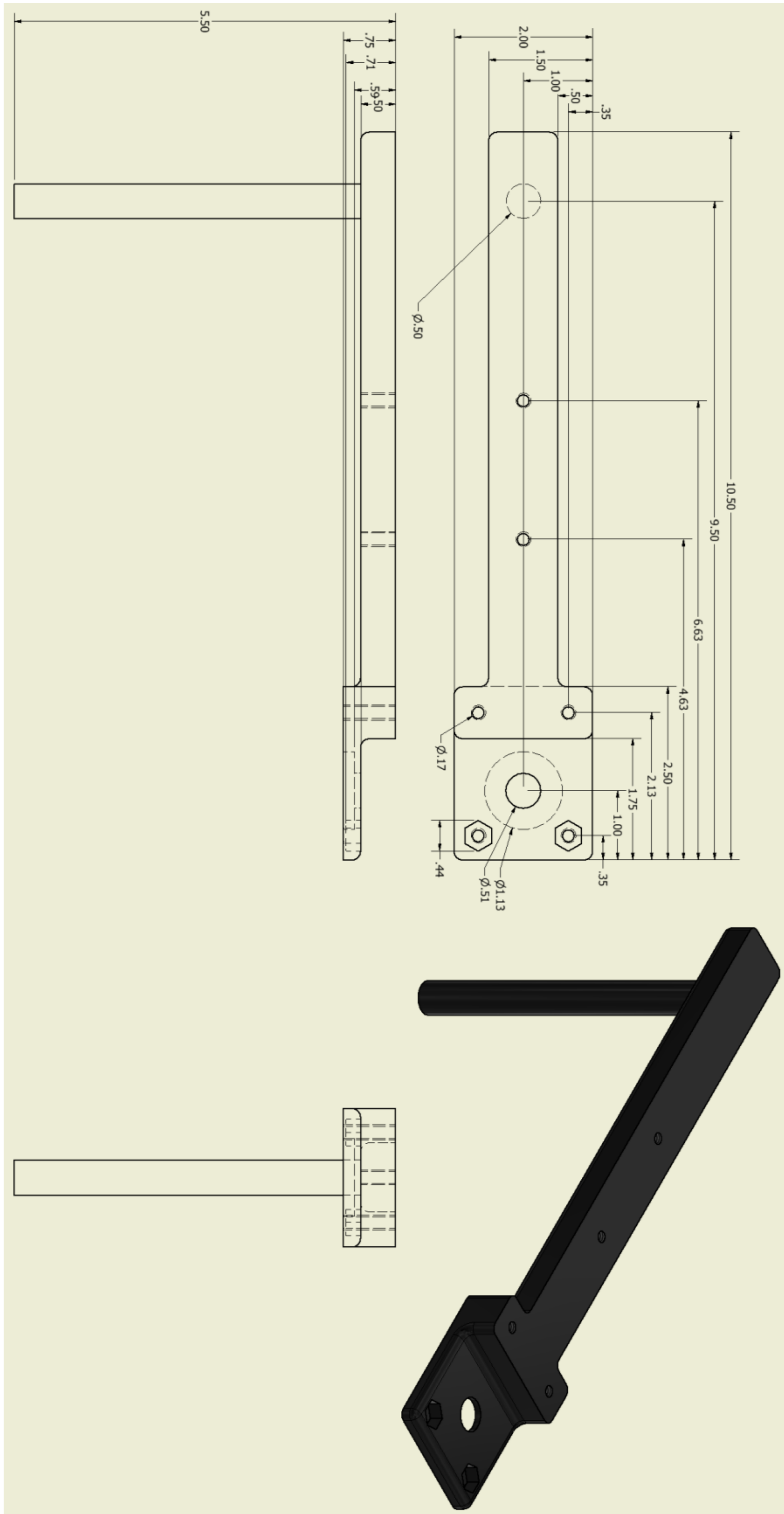


Figure 32: Arm Segment 1 Upper Half Mechanical Drawing (Units: in)

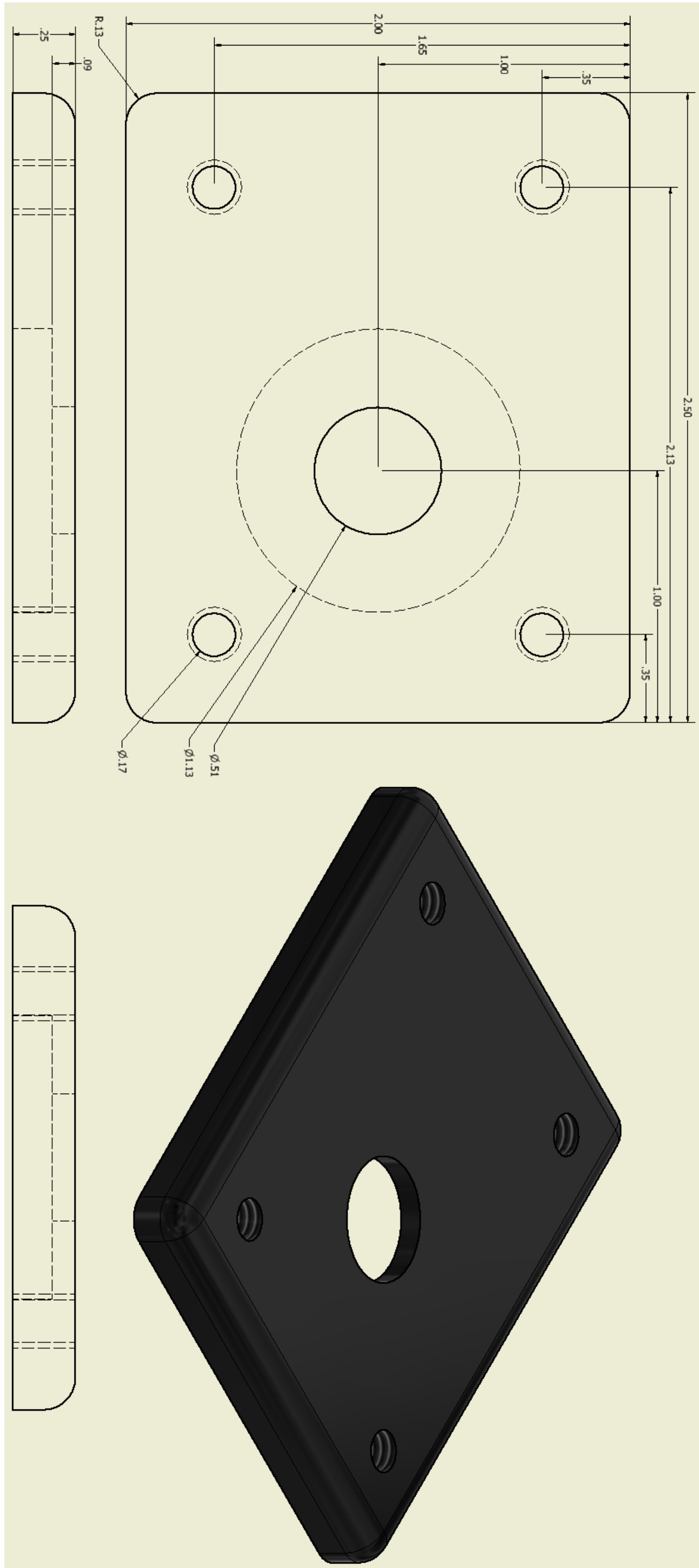


Figure 33: Arm Segment 1 Upper Half Bearing Cover Mechanical Drawing (Units: in)

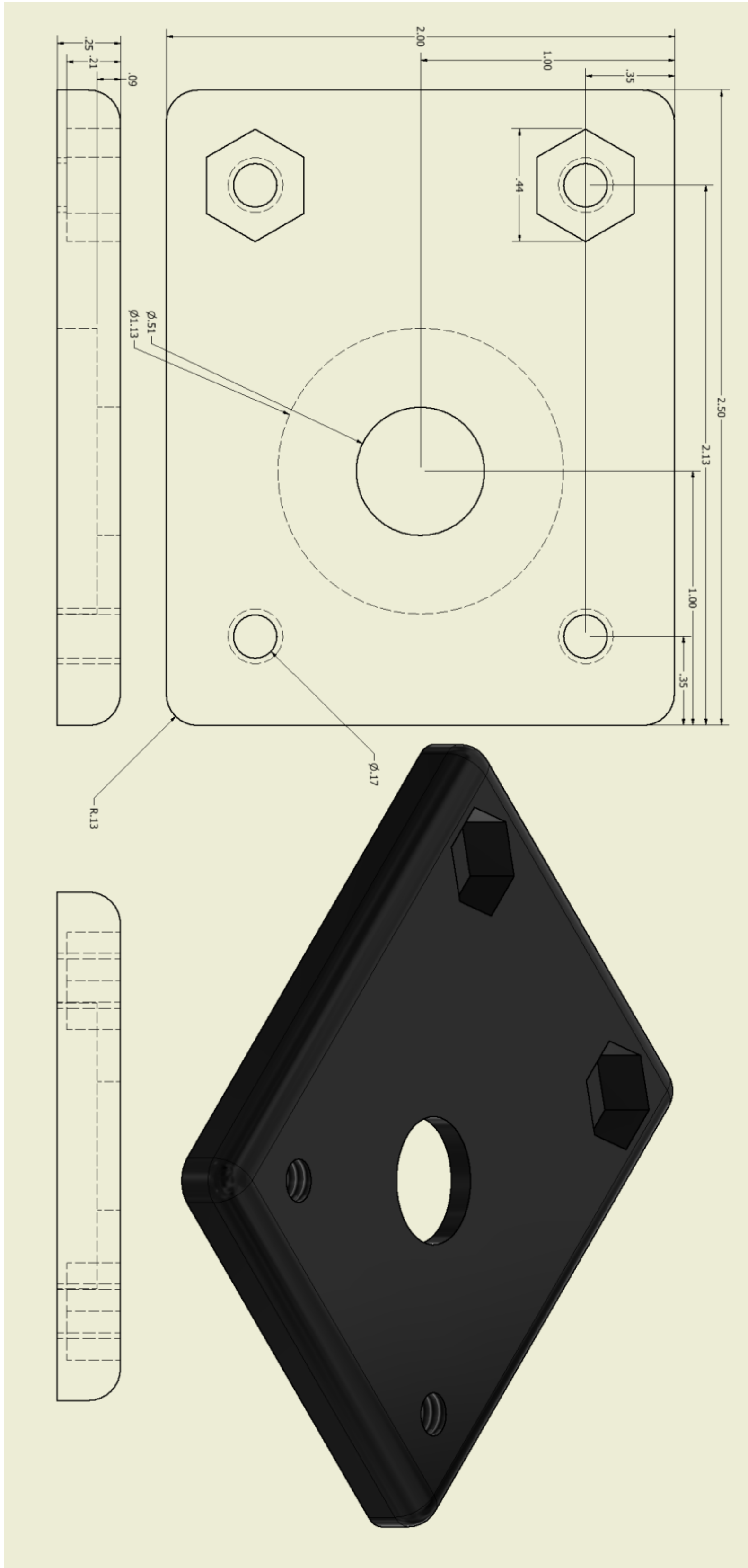


Figure 34: Arm Segment 1 Lower Half Bearing Cover Mechanical Drawing (Units: in)

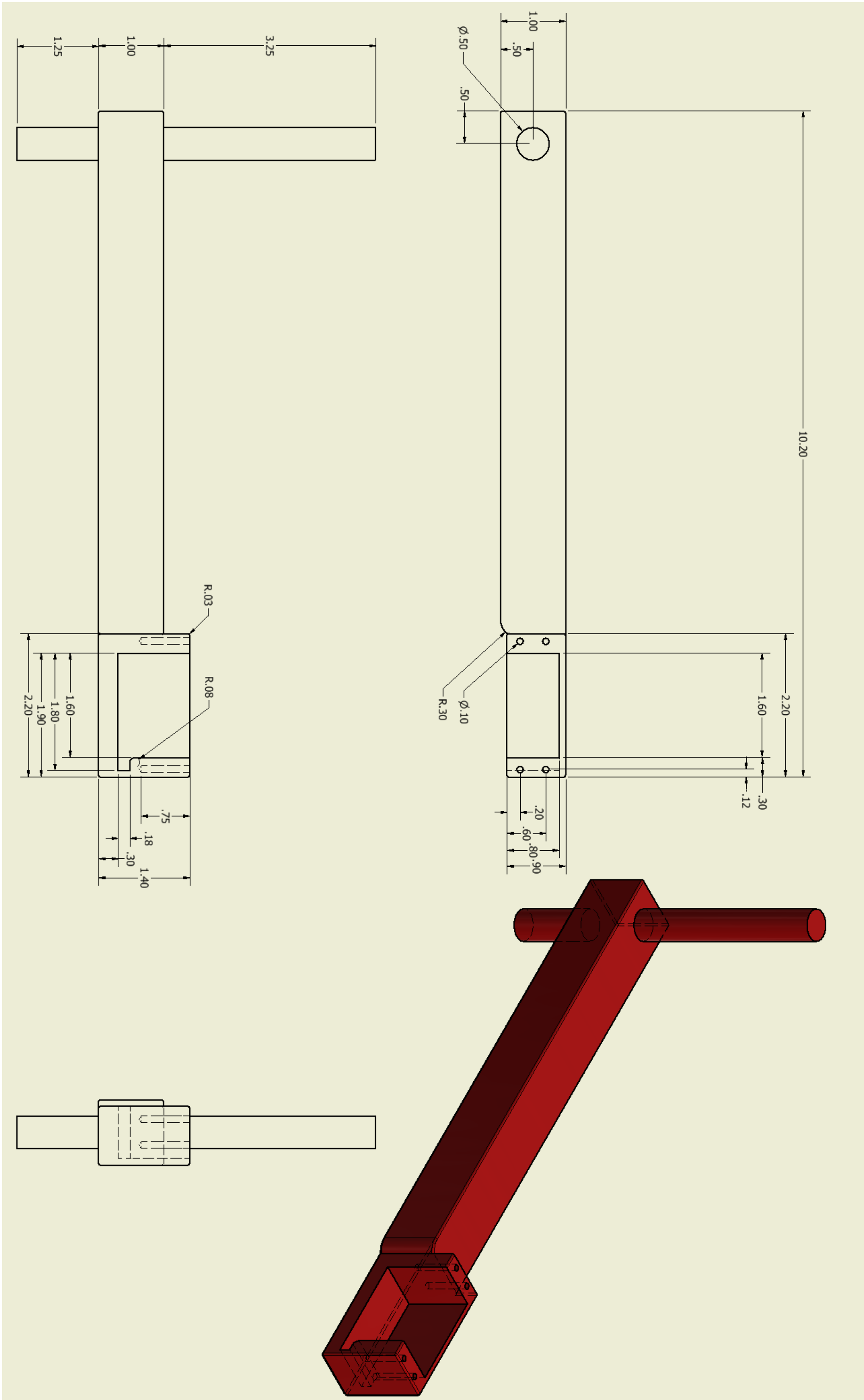


Figure 35: Arm Segment 2 Mechanical Drawing (Units: in)

2.3.1.4 Limit Switches

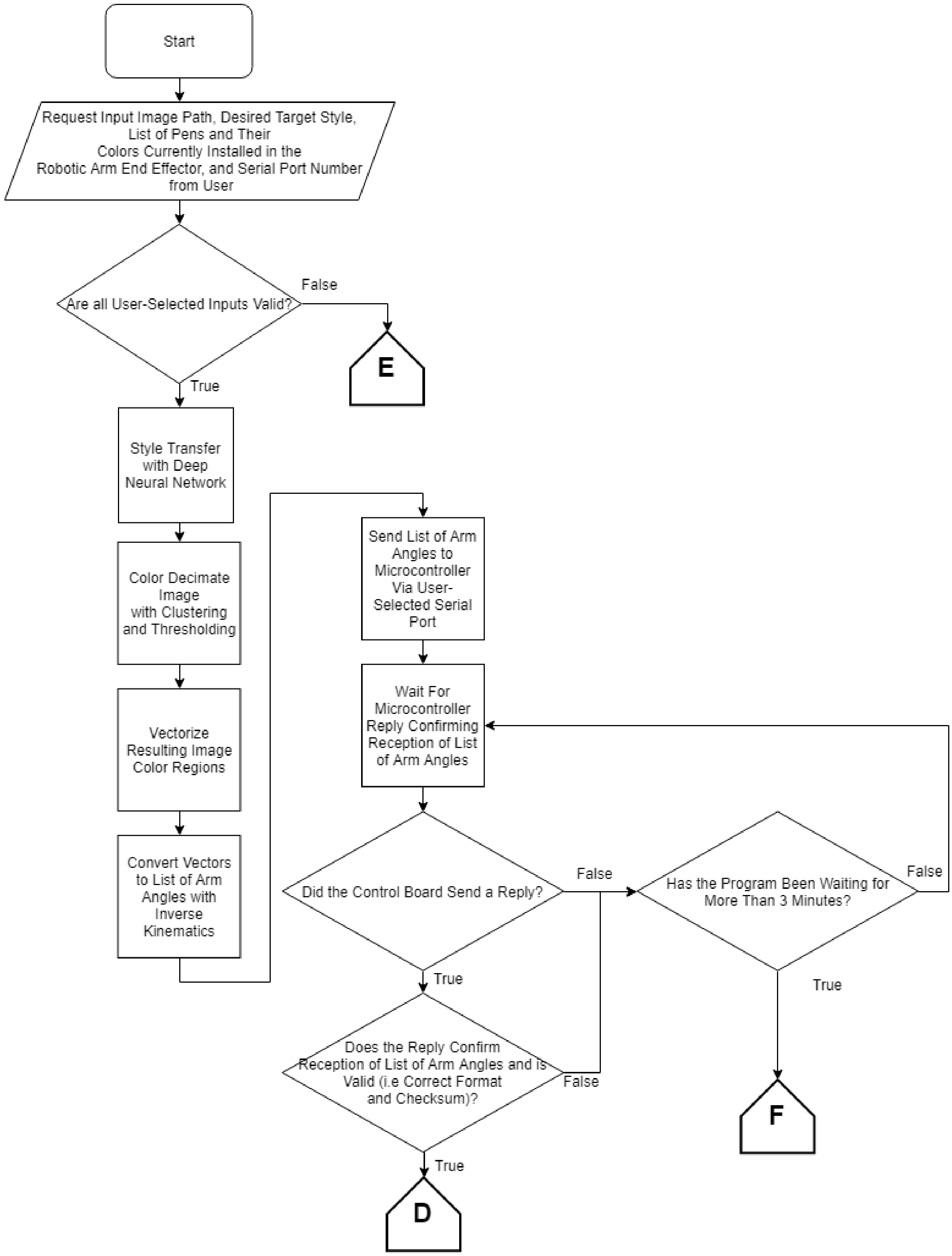
These limit switches are used to determine if an arm segment has reached the end of its allowable travel range and thus prevent the robotic arm from colliding with the base or itself, and are also used at startup to calibrate the zero position of the robotic arm, as by counting the number of steps sent to the stepper motor controllers allows relative position tracking, but not absolute position tracking. For simplicity, these switches will be connected directly to the microcontroller's GPIO pins and utilize built-in pull-up resistors.

Table 4: Limit Switch Requirements and Verification Procedures

Requirement	Verification Procedure
1. Latency of at most 10 ms +/- 1%.	1. Assemble all physical components and connect all electronic components and boards together as specified in the block diagram. 2. Upload a modified version of the microcontroller software that echoes back any limit switch triggering. 3. Install a camera such that the limit switch and arm's contact area is clearly visible. 4. Using the PC, command the arm such that the limit switch will be triggered. 5. Note the time recorded on the PC, and compare with the camera recording to determine the delay.
2. Able to withstand 10 mA or more of current at 5V +/- 10% for at least 100 hours.	1. Assemble all physical components and connect all electronic components and boards together as specified in the block diagram. 2. Leave system on for 100 hours. 3. Redo the verification procedures for Requirement 1 to confirm no degradation in functionality.

2.3.2 Software (10 Points)

Users are expected to have an image and a desired style as input. A simple GUI will show both the original image and the output image after style transfer with a deep neural network, and any fault conditions and the current progress on the current operation. Given an input image and a style, the PC software first performs style transformation on the input image using a deep neural network for the selected style, then converts the result into a series of arm angles by clustering colors and thresholding to get color regions, which are then converted into a series of lines via a vectorization algorithm, with fill patterns optimized for the polar nature of the SCARA robotic arm, before finally transforming the lines into a list of arm angles via inverse kinematics. Subsequently, the list of arm angles is transferred to the microcontroller via the USB-to-Serial converter IC. Meanwhile, the microcontroller software simply waits for the list of arm angles, and upon receiving the list of arm angles, actuates the motors and servos as necessary, halting if any abnormal conditions including motor overtemperature, motor overcurrent, and motor stall are detected.


Start
Request Input Image Path, Desired Target Style, List of Pens and Their Colors Currently Installed in the Robotic Arm End Effector, and Serial Port Number from User
Are all User-Selected Inputs Valid?
False
E
True
Style Transfer with Deep Neural Network
Color Decimate Image with Clustering and Thresholding
Vectorize Resulting Image Color Regions
Convert Vectors to List of Arm Angles with Inverse Kinematics
Send List of Arm Angles to Microcontroller Via User-Selected Serial Port
Wait For Microcontroller Reply Confirming Reception of List of Arm Angles
Did the Control Board Send a Reply?
False
Has the Program Been Waiting for More Than 3 Minutes?
False
True
True
F
Does the Reply Confirm Reception of List of Arm Angles and is Valid (i.e Correct Format and Checksum)?
False
True
D

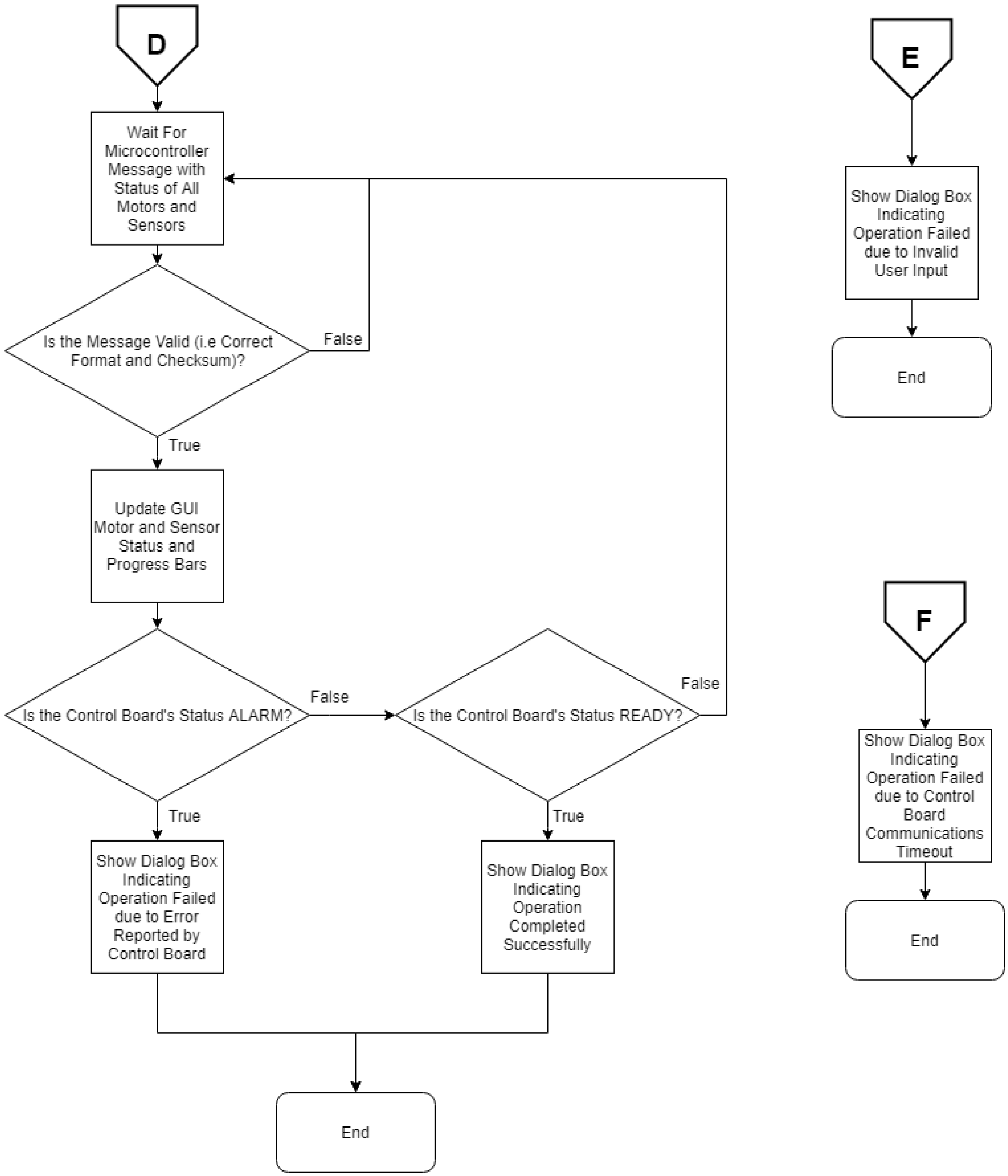


Figure 36: PC Software Logic Block Diagram

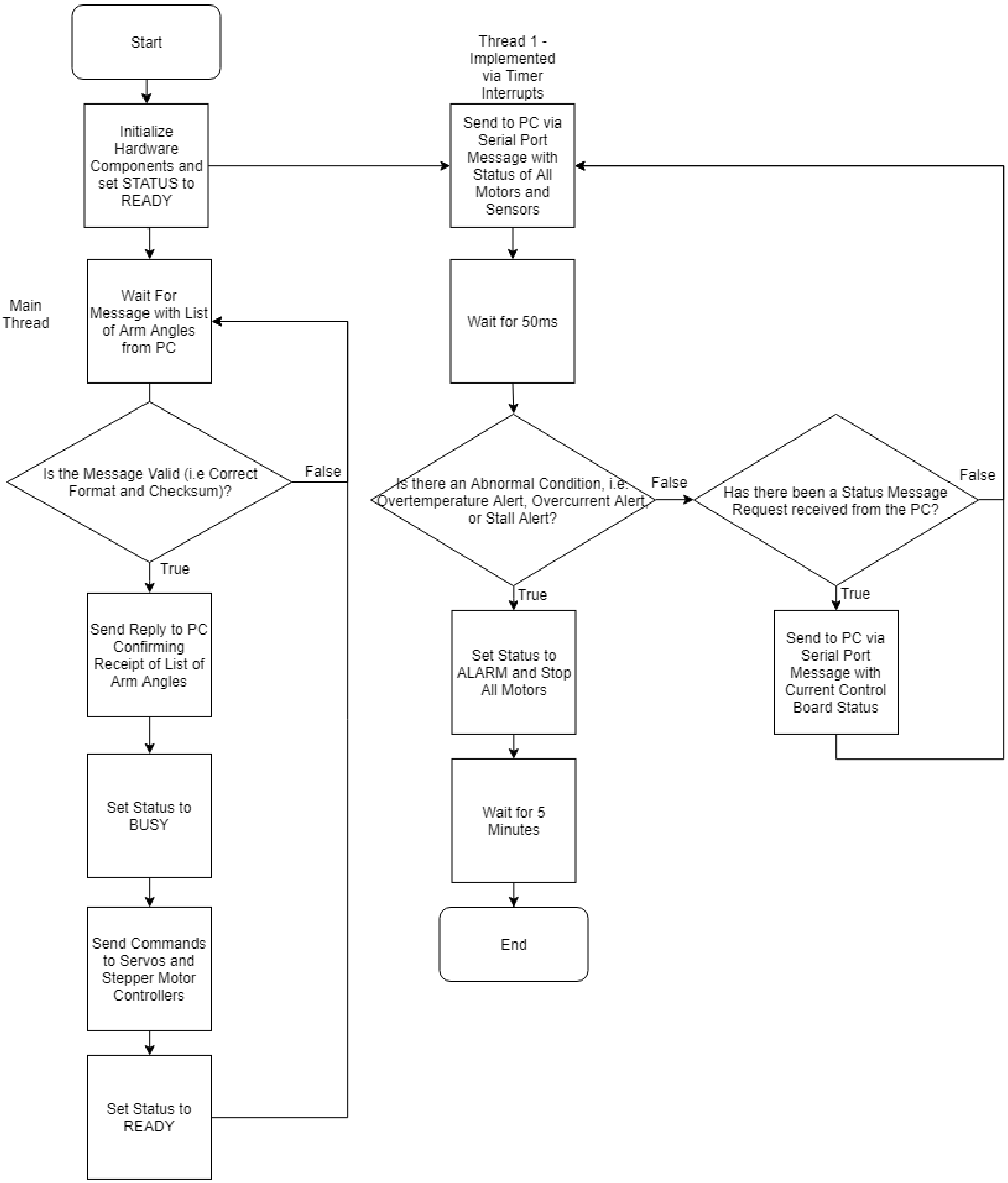


Figure 37: Microcontroller Software Logic Block Diagram

2.3.3 Control Board (10 Points)

The control board acts as the central coordinator of the system, receiving lists of arm angles from the PC via a USB-to-Serial interface IC, and sends appropriate command sequences to the joint motors and linear actuators while maintaining accurate position control throughout. The control board monitors for abnormal conditions, including Motor Board overtemperature, motor overcurrent, motor stalling, and arm segments reaching end of travel, as determined by the limit switches and other sensors built into the stepper motor controller IC, and stops the system as necessary, and reports any such conditions to the PC software and via LEDs such that the user can take corrective action.

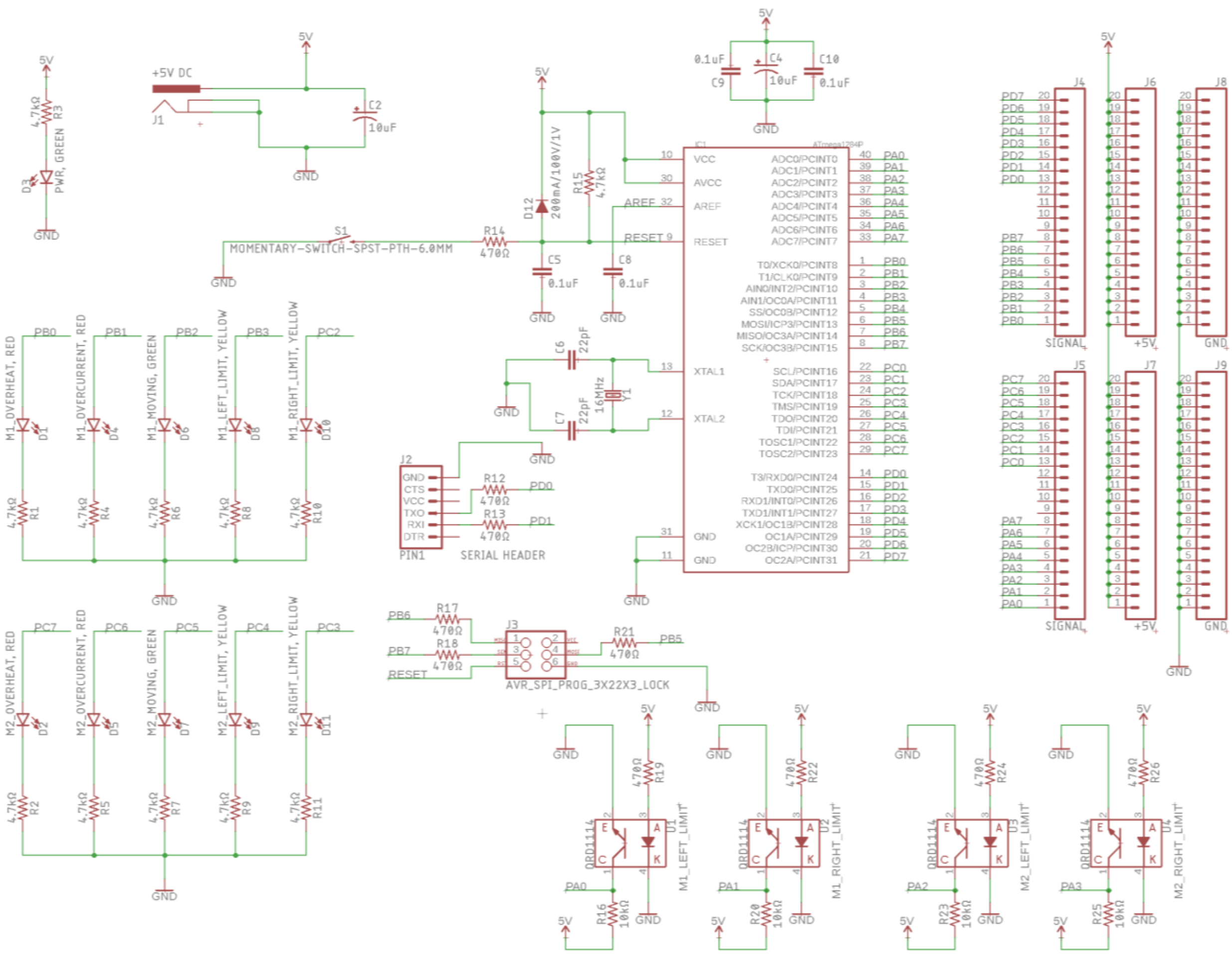


Figure 38: Control Board Circuit Schematic

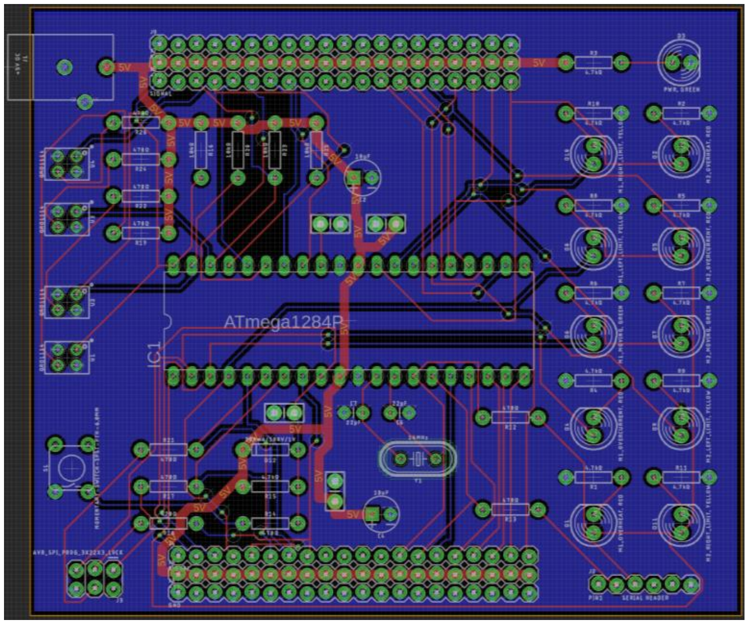


Figure 39: Control Board PCB Design

2.3.3.1 Microcontroller

The microcontroller communicates with the PC software through a USB-to-Serial IC that converts USB messages to and from logic level serial port signals to fetch lists of arm angles from the PC program. The microcontroller then parses the lists of motor angles, and commands the motor drivers to generate the corresponding motor driving waveforms via messages send over the SPI bus. The microcontroller also checks and stops the motor if any limit switches, connected to the GPIO pins of the microcontroller, are triggered, or if the motor driver reports any fault conditions as listed above, and relays this information to the PC program as well as by lighting/extinguishing appropriate output LEDs connected to GPIO pins.

The microcontroller also generates PWM signals for commanding for the servo on the pen linear actuator via a separate GPIO pin, which will then lift the pen from contact with the paper or drop the pen into contact with the paper accordingly. We have picked the ATmega1284p to fit our design requirements and as we are most familiar with the Atmel ATmega architecture.

Table 5: Microcontroller Requirements and Verification Procedures

Requirement	Verification Procedure
1. Built-in hardware timer for servo PWM with resolution of 0.1 ms or better.	1. Check microcontroller datasheet and confirm requirements. 2. Assemble and connect all electronic components and boards together as specified in the block diagram. 3. Upload a modified version of the microcontroller software that outputs a 50% duty-cycle, 1 ms period square wave using Timer1 on pin PD4 of the microcontroller.

	4. Verify output is as expected, i.e. error in square wave period is less than 0.1 ms, with an oscilloscope measuring the aforementioned pin.
2. 5V TTL logic voltage level.	1. Check microcontroller datasheet and confirm requirements. 2. Assemble and connect all electronic components and boards together as specified in the block diagram. 3. Upload a modified version of the microcontroller software that outputs a 50% duty-cycle, 1 ms period square wave using Timer1 on pin PD4 of the microcontroller. 4. Verify output on PD4 have low levels between 0V to 0.8V and high levels between 2V and 5V with an oscilloscope.
3. Greater than or equal 30 GPIO pins for reading sensors and controlling status LEDs.	1. Check microcontroller datasheet and confirm requirements. 2. Assemble and connect all electronic components and boards together as specified in the block diagram. 3. Upload a modified version of the microcontroller software that allows us to read and write to/from I/O ports from a PC serial connection. 4. Using a short PC program compatible with the aforementioned program, write to I/O ports containing LEDs all ones to turn on all LEDs. 5. Verify LEDs light as is as expected. 6. Using a short PC program compatible with the aforementioned program, write to I/O ports containing LEDs all zeros to turn off all LEDs. 7. Verify LEDs turn off as is as expected. 8. Set all switches to OFF position using tape. 9. Using a short PC program compatible with the aforementioned program, read from I/O ports containing switches. 10. Verify switches read as is as expected. 11. Set all switches to ON position using tape. 12. Using a short PC program compatible with the aforementioned program, read from I/O ports containing switches. 13. Verify switches read as is as expected.

4. At least one SPI port for communications with stepper motor drivers.	1. Check microcontroller datasheet and confirm requirements. 2. Assemble and connect all electronic components and boards together as specified in the block diagram. 3. Upload a modified version of the microcontroller software that echoes back all characters sent to it via SPI and Serial. 4. Using a short PC program, send random data to the SPI port. 5. Verify output is as expected.
5. At least one serial port for communication with PC.	1. Check microcontroller datasheet and confirm requirements. 2. Assemble and connect all electronic components and boards together as specified in the block diagram. 3. Upload a modified version of the microcontroller software that echoes back all characters sent to it via SPI and Serial. 4. Using a short PC program, send random data to the Serial port. 5. Verify output is as expected.
6. At least 32K program memory.	1. Check microcontroller datasheet and confirm requirements. 2. Assemble and connect all electronic components and boards together as specified in the block diagram. 3. Upload a modified version of the microcontroller software that allows us to read and write to/from arbitrary Flash locations from a PC serial connection. 4. Using a short PC program compatible with the aforementioned program, write random data to the entire Flash address space, i.e. every Flash memory bit, and read back the written data from Flash to the PC 5. Verify output is as expected.
7. At least 16K RAM for buffering PC arm angle lists.	1. Check microcontroller datasheet and confirm requirements. 2. Assemble and connect all electronic components and boards together as specified in the block diagram. 3. Upload a modified version of the microcontroller software that allows us

	to read and write to/from arbitrary RAM locations from a PC serial connection. 4. Using a short PC program compatible with the aforementioned program, write random data to the entire RAM address space, i.e. every RAM bit, and read back the written data from RAM to the PC. 5. Verify output is as expected.

2.3.3.2 Status LEDs

These LEDs are responsible for notifying the user if any limit switches are triggered, or if the motor driver reports any fault conditions, specifically if the Motor Board is overheating, a motor overcurrent warning is triggered, or if a motor has stalled and missed a step potentially resulting in incorrect positioning. The LEDs are connected to the microcontroller via the microcontroller's GPIO pins.

Table 6: Status LEDs Requirements and Verification Procedures

Requirement	Verification Procedure
1. 100 mcd brightness under a current draw of 0.5 mA +/- 10%.	1. Make sure the bench power supply mentioned below is plugged in correctly and grounded, and is powered off. 2. Connect one end of a 4.7kΩ resistor to the positive terminal of the bench power supply, and connect the other end of the resistor to the anode of the LED, using a breadboard and breadboard wires as necessary. 3. Connect the cathode of the LED to the bench power supply. 4. Set the power supply to +5V with a current limit of 0.5 mA, and turn on the output. 5. Verify brightness is as expected using a Digital Light Meter LT300 from Extech Instruments.

2.3.3.3 USB-to-Serial Converter

The USB-to-Serial converter IC is responsible for converting between USB signals and TTL logic level serial port signals such that the microcontroller can communicate with the PC software to both alert the user to fault conditions and to also get the list of arm angles such that drawing can occur. It has connections to both the PC's USB port as well as to the microcontroller's serial port.

Table 7: USB-to-Serial Converter Requirements and Verification Procedures

Requirement	Verification Procedure
1. Enables PC to communicate with serial port on microcontroller with speed of 115.2 Kbps or greater.	1. Check USB-to-Serial Converter IC datasheet. 2. Assemble and connect all electronic components and boards together as specified in the block diagram. 3. Upload a modified version of the microcontroller software that echoes back all characters sent to it via the serial port. 4. Using a short PC program, send random data to the Serial port at 115.2 Kbps. 5. Verify received output on computer is as expected.
2. Serial port output is 5V TTL compliant.	1. Check USB-to-Serial Converter IC datasheet. 2. Assemble and connect all electronic components and boards together as specified in the block diagram. 3. Using a short PC program, send random data to the Serial port at 115.2 Kbps. 4. Verify received output on serial port pins have low levels between 0V to 0.8V and high levels between 2V and 5V with an oscilloscope.

2.3.3.4 Bench Power Supply (+5V)

This power supply provides power for all logic ICs, in particular, the microcontroller and the USB-to-Serial converter IC, and also provides power for the status LEDs and the servo used to drive the linear actuator. As each servo consumes 250 mA and 500 mA peak when stalled, we estimate a 3A total board consumption. As the system is immobile, a static wall-based power supply is acceptable, and necessary due to the high system power required.

Table 8: Bench Power Supply (+5V) Converter Requirements and Verification Procedures

Requirement	Verification Procedure
1. 5V +/- 10%, 3A +/- 5% output.	1. Make sure the bench power supply, multimeter, and electronic load mentioned below are plugged in correctly and grounded, and are powered off. 2. Turn on and set a Keysight 34461A multimeter into DCV mode, then using banana cables, attach the multimeter such that the positive terminals of the multimeter and bench power supply are connected, and similarly for the negative terminals.

	3. Using banana cables, attach the Agilent 6060B electronic load such that the positive terminals of the electronic load and bench power supply are connected, and similarly for the negative terminals. 4. Turn on and set the bench power supply to output +5V with a current limit of 3A. 5. Turn on and set the Agilent 6060B electronic load in the correct mode, to emulate a +5V 3A DC load. 6. Check that the measured voltage and current meets requirements, using the multimeter and an ammeter integrated in the electronic load respectively.
2. Ripple of 100 mV or lower under steady state, no motor movement conditions.	1. Make sure the bench power supply, oscilloscope, and robotic arm electronics mentioned below are plugged in correctly and grounded, and are powered off. 2. Turn on the DSO7104B oscilloscope, then using a probe, attach the oscilloscope such that the signal terminal of the oscilloscope and the positive terminal of the +24V bench power supply are connected, and similarly for the ground terminals of both devices. 3. Using banana cables, attach the robotic arm's Control Boards such that the positive terminals of the Motor Board's motor power supply terminal and the +5V bench power supply are connected, and similarly for the negative terminals of the Motor Board's motor power supply terminal and the +5V bench power supply. 4. Using banana cables, attach the robotic arm's Motor Boards such that the positive terminals of the Motor Board's motor power supply terminal and the +24V bench power supply are connected, and similarly for the negative terminals of the Motor Board's motor power supply terminal and the +24V bench power supply. 5. Turn on both bench power supplies and command the Control Board via the PC to not attempt to move any motors.

	6. Check that the measured ripple meets specifications, using measurement capabilities built into the oscilloscope.
3. Input of 120V AC.	1. Check the specifications of the bench power supply for proper certifications.

2.3.4 Motor Board (15 Points)

The Motor Board simply houses the stepper motor controller ICs and any needed heatsinks. Due to the higher drive voltages used, to ensure isolation, and to allow for mounting closer to each joint motor, the Motor Board is a separate, self-contained unit. Each stepper motor controller IC monitors for abnormal conditions, including overtemperature, motor overcurrent, and motor stalling, and relays this information back to the microcontroller via the SPI bus, and simultaneously outputs the necessary current-limited drive waveform to position the stepper motors at the correct angles as commanded by messages sent from the microcontroller over the SPI bus. Based on our requirements below, we have chosen the L6470 Stepper Motor Controller IC.

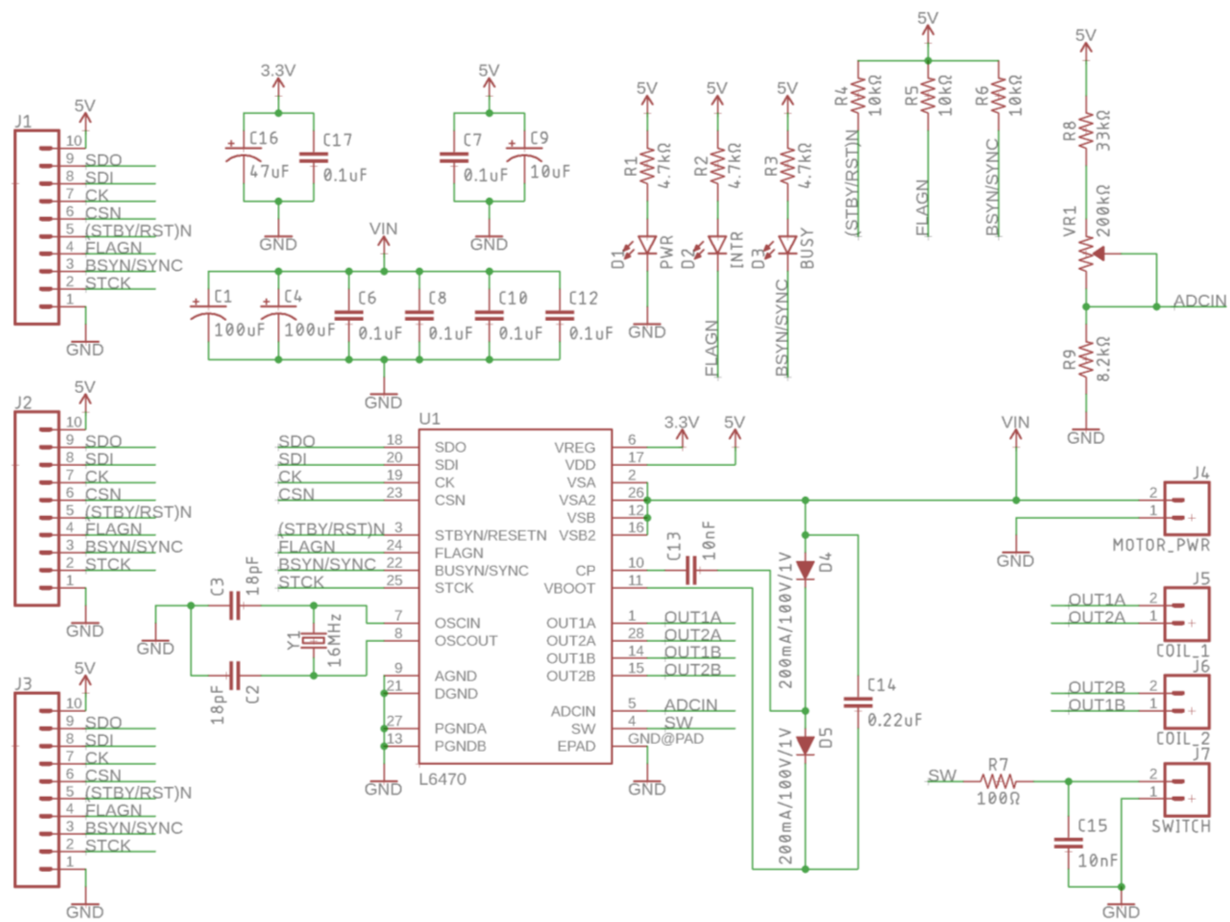


Figure 40: Motor Board Circuit Schematic

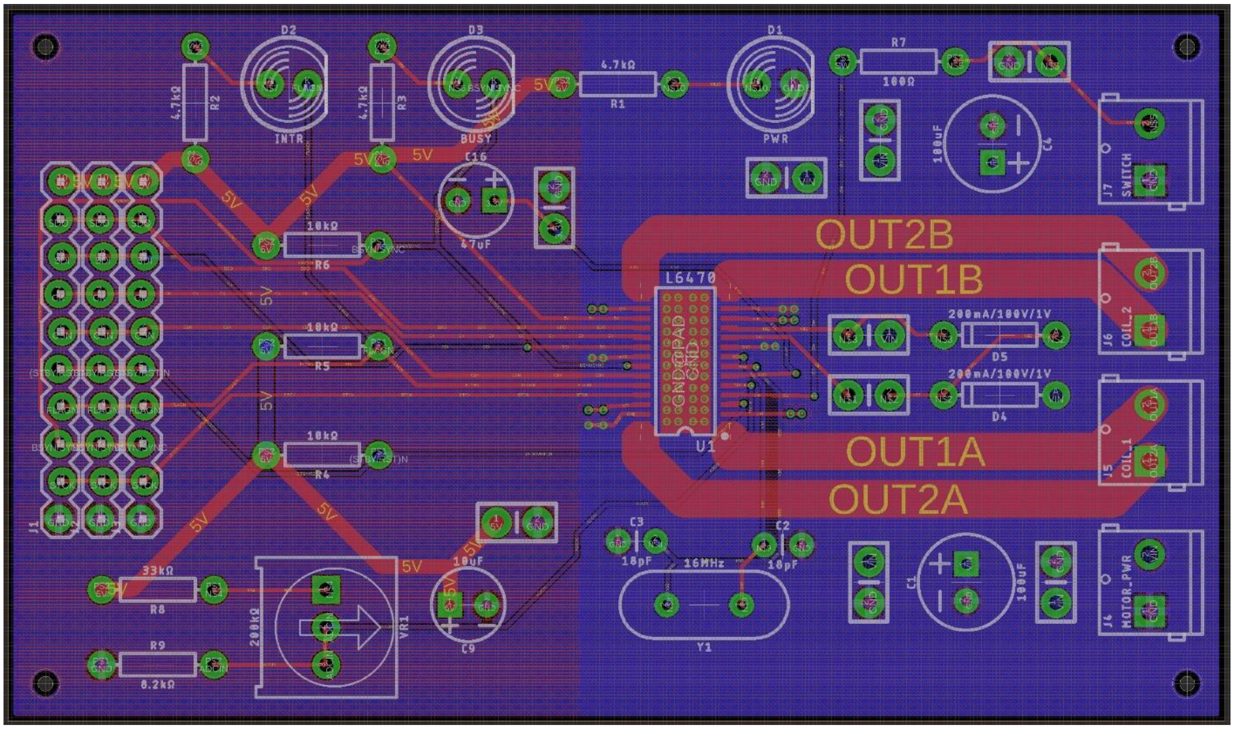


Figure 41: Motor Board PCB Design

2.3.4.1 Stepper Motor Driver IC

The motor driver ICs receive commands from the microcontroller and monitors for abnormal conditions, including overtemperature, motor overcurrent, and motor stalling, and relays this information back to the microcontroller, and generate a constant current PWM waveform to drive the stepper motors. Due to the inherent back-EMF of the stepper design, stepper motors require high voltage for high rotational speeds. We use a constant current PWM waveform to prevent the motor coils from overheating, and also to allow for microstepping, increasing stepper motor positioning accuracy and movement smoothness. The stepper motor controller ICs communicate with the microcontroller over the SPI bus, while power is supplied by a separate +24V bench power supply.

Table 9: Joint Motor Requirements and Verification Procedures

Requirement	Verification Procedure
1. 3A R.M.S motor drive current or greater at 24 V operating voltage or greater.	1. Assemble and connect all electronic components and boards together as specified in the block diagram, except for the stepper motors. 2. Make sure the bench power supply, electronic load, and robotic arm electronics mentioned below are plugged in correctly and grounded, and are powered off. 3. Using banana cables, attach the Agilent 6060B electronic load such that the positive terminals of the electronic load

	and bench power supply are connected, and similarly for the negative terminals. 4. Using banana cables, attach the robotic arm’s Control Boards such that the positive terminals of the Motor Board’s motor power supply terminal and the +5V bench power supply are connected, and similarly for the negative terminals of the Motor Board’s motor power supply terminal and the +5V bench power supply. 5. Using banana cables, attach the robotic arm’s Motor Boards such that the positive terminals of the Motor Board’s motor power supply terminal and the +24V bench power supply are connected, and similarly for the negative terminals of the Motor Board’s motor power supply terminal and the +24V bench power supply. 6. Turn on both bench power supplies. 7. Turn on and set the Agilent 6060B electronic load in the correct mode, to emulate a +24V 3A RMS, 5A peak load. 8. Command the Control Board via the PC to move the stepper motors with a 3A R.M.S. drive current. 9. Maintain this load for 1 hour. 10. Check that the measured current meets specifications, using measurement capabilities built into the electronic load and bench power supplies, and check the Motor Board and stepper motor controller for any sign of thermal damage via visual inspection and a Etekcity 630 Lasergrip non-contact thermometer with measurement accuracy of +- 0.1 deg. C. All board and component temperatures should remain below 100 deg. C.
2. 5A peak motor drive current or greater at 24 V operating voltage or greater.	1. Assemble and connect all electronic components and boards together as specified in the block diagram, except for the stepper motors. 2. Make sure the bench power supply, electronic load, and robotic arm electronics mentioned below are plugged in correctly and grounded, and are powered off.

	3. Using banana cables, attach the Agilent 6060B electronic load such that the positive terminals of the electronic load and bench power supply are connected, and similarly for the negative terminals. 4. Using banana cables, attach the robotic arm's Control Boards such that the positive terminals of the Motor Board's motor power supply terminal and the +5V bench power supply are connected, and similarly for the negative terminals of the Motor Board's motor power supply terminal and the +5V bench power supply. 5. Using banana cables, attach the robotic arm's Motor Boards such that the positive terminals of the Motor Board's motor power supply terminal and the +24V bench power supply are connected, and similarly for the negative terminals of the Motor Board's motor power supply terminal and the +24V bench power supply. 6. Turn on both bench power supplies. 7. Turn on and set the Agilent 6060B electronic load in the correct mode, to emulate a +24V 3A RMS, 5A peak load. 8. Command the Control Board via the PC to move the stepper motors with a 3A R.M.S. drive current. 9. Maintain this load for 1 hour. 10. Check that the measured current meets specifications, using measurement capabilities built into the electronic load and bench power supplies, and check the Motor Board and stepper motor controller for any sign of thermal damage via visual inspection and a Etekcity 630 Lasergrip non-contact thermometer with measurement accuracy of +- 0.1 deg. C. All board and component temperatures should remain below 100 deg. C.
3. Overcurrent protection.	1. Assemble and connect all electronic components and boards together as specified in the block diagram, except for the stepper motors. 2. Make sure the bench power supply, electronic load, and robotic arm

	electronics mentioned below are plugged in correctly and grounded, and are powered off. 3. Using banana cables, attach the Agilent 6060B electronic load such that the positive terminals of the electronic load and bench power supply are connected, and similarly for the negative terminals. 4. Using banana cables, attach the robotic arm's Control Boards such that the positive terminals of the Motor Board's motor power supply terminal and the +5V bench power supply are connected, and similarly for the negative terminals of the Motor Board's motor power supply terminal and the +5V bench power supply. 5. Using banana cables, attach the robotic arm's Motor Boards such that the positive terminals of the Motor Board's motor power supply terminal and the +24V bench power supply are connected, and similarly for the negative terminals of the Motor Board's motor power supply terminal and the +24V bench power supply. 6. Turn on both bench power supplies. 7. Turn on and set the Agilent 6060B electronic load in the correct mode, to emulate a +24V 7A RMS, 7A peak load. 8. Command the Control Board via the PC to move the stepper motors with a 7A R.M.S. drive current. 9. Wait until overcurrent shutdown occurs. 10. During wait, check that the measured current meets specifications, using measurement capabilities built into the electronic load and bench power supplies, and check the Motor Board and stepper motor controller for any sign of thermal damage via visual inspection and a Etekcity 630 Lasergrip non-contact thermometer with measurement accuracy of +- 0.1 deg. C. All board and component temperatures should remain below 100 deg. C.
4. Overtemperature protection.	1. Assemble and connect all electronic components and boards together as

	specified in the block diagram, except for the stepper motors. 2. Make sure the bench power supply, electronic load, and robotic arm electronics mentioned below are plugged in correctly and grounded, and are powered off. 3. Using banana cables, attach the Agilent 6060B electronic load such that the positive terminals of the electronic load and bench power supply are connected, and similarly for the negative terminals. 4. Using banana cables, attach the robotic arm's Control Boards such that the positive terminals of the Motor Board's motor power supply terminal and the +5V bench power supply are connected, and similarly for the negative terminals of the Motor Board's motor power supply terminal and the +5V bench power supply. 5. Using banana cables, attach the robotic arm's Motor Boards such that the positive terminals of the Motor Board's motor power supply terminal and the +24V bench power supply are connected, and similarly for the negative terminals of the Motor Board's motor power supply terminal and the +24V bench power supply. 6. Turn on both bench power supplies. 7. Turn on and set the Agilent 6060B electronic load in the correct mode, to emulate a +24V 5A RMS, 7A peak load. 8. Command the Control Board via the PC to move the stepper motors with a 5A R.M.S. drive current. 9. Wait until overheat shutdown occurs. 10. During wait, check that the measured current meets specifications, using measurement capabilities built into the electronic load and bench power supplies, and check the Motor Board and stepper motor controller for any sign of thermal damage via visual inspection and a Etekcity 630 Lasergrip non-contact thermometer with measurement accuracy of +- 0.1 deg. C. All board and

	component temperatures should remain below 100 deg. C.
5. Motor stall detection.	1. Assemble all physical components and connect all electronic components and boards together as specified in the block diagram. 2. Make sure the bench power supply and robotic arm electronics mentioned below are plugged in correctly and grounded, and are powered off. 3. Using banana cables, attach the robotic arm's Control Boards such that the positive terminals of the Motor Board's motor power supply terminal and the +5V bench power supply are connected, and similarly for the negative terminals of the Motor Board's motor power supply terminal and the +5V bench power supply. 4. Using banana cables, attach the robotic arm's Motor Boards such that the positive terminals of the Motor Board's motor power supply terminal and the +24V bench power supply are connected, and similarly for the negative terminals of the Motor Board's motor power supply terminal and the +24V bench power supply. 5. Turn on both bench power supplies. 6. Command the Control Board via the PC to move the stepper motors with a 3A R.M.S. drive current. 7. Block the movement and wait for shutdown with a piece of metal placed in the path of the arm. 8. Wait until stall shutdown occurs. 9. Check that the measured current meets specifications, using measurement capabilities built into the electronic load and bench power supplies, and check the Motor Board and stepper motor controller for any sign of thermal damage via visual inspection and a Etekcity 630 Lasergrip non-contact thermometer with measurement accuracy of +- 0.1 deg. C. All board and component temperatures should remain below 100 deg. C.

6. Constant-current motor drive with capability for 1/32 microstep divisions or better.	1. Check the specifications of the stepper motor controller IC for relevant functionality. 2. Assemble all physical components and connect all electronic components and boards together as specified in the block diagram. 3. Make sure the bench power supply and robotic arm electronics mentioned below are plugged in correctly and grounded, and are powered off. 4. Using banana cables, attach the robotic arm's Control Boards such that the positive terminals of the Motor Board's motor power supply terminal and the +5V bench power supply are connected, and similarly for the negative terminals of the Motor Board's motor power supply terminal and the +5V bench power supply. 5. Using banana cables, attach the robotic arm's Motor Boards such that the positive terminals of the Motor Board's motor power supply terminal and the +24V bench power supply are connected, and similarly for the negative terminals of the Motor Board's motor power supply terminal and the +24V bench power supply. 6. Turn on both bench power supplies. 7. Command the Control Board via the PC to move the stepper motors with a 3A R.M.S. drive current using the correct microstepping mode register settings. 8. Check that the measured current meets specifications for a microstepping waveform, i.e. is sinusoidal with a Total Harmonic Distortion of less than 10% with the correct R.M.S. current level, using an oscilloscope, and check the Motor Board and stepper motor controller for any sign of thermal damage via visual inspection and a Etekcity 630 Lasergrip non-contact thermometer with measurement accuracy of +- 0.1 deg. C. All board and component temperatures should remain below 100 deg. C.

2.3.4.2 Bench Power Supply (+24V)

This power supply provides power for the stepper motor driver ICs and the stepper motors themselves. A separate supply is required for both isolation and to reduce noise coupled into the logic level power supply. As we have two stepper motors in the system, each with a R.M.S. current consumption of 3A, to allow some margin for current peaks when the motors are stalled, we estimate a total current consumption of 10A. As the system is immobile, a static wall-based power supply is acceptable, and necessary due to the high system power required.

Table 10: Bench Power Supply (+24V) Requirements and Verification Procedures

Requirement	Verification Procedure
1. 24V +/- 10%, 10A +/- 5% output.	1. Make sure the bench power supply, multimeter, and electronic load mentioned below are plugged in correctly and grounded, and are powered off. 2. Turn on and set a Keysight 34461A multimeter into DCV mode, then using banana cables, attach the multimeter such that the positive terminals of the multimeter and bench power supply are connected, and similarly for the negative terminals. 3. Using banana cables, attach the Agilent 6060B electronic load such that the positive terminals of the electronic load and bench power supply are connected, and similarly for the negative terminals. 4. Turn on and set the bench power supply to output +24V with a current limit of 10A. 5. Turn on and set the Agilent 6060B electronic load in the correct mode, to emulate a +24V 10A DC load. 6. Check that the measured voltage and current meets requirements, using the multimeter and an ammeter integrated in the electronic load respectively.
2. Ripple of 100 mV or lower under steady state, no motor movement conditions.	1. Make sure the bench power supply, oscilloscope, and robotic arm electronics mentioned below are plugged in correctly and grounded, and are powered off. 2. Turn on the DSO7104B oscilloscope, then using a probe, attach the oscilloscope such that the signal terminal of the oscilloscope and the positive terminal of the +24V bench power supply are

	connected, and similarly for the ground terminals of both devices. 3. Using banana cables, attach the robotic arm's Control Boards such that the positive terminals of the Motor Board's motor power supply terminal and the +5V bench power supply are connected, and similarly for the negative terminals of the Motor Board's motor power supply terminal and the +5V bench power supply. 4. Using banana cables, attach the robotic arm's Motor Boards such that the positive terminals of the Motor Board's motor power supply terminal and the +24V bench power supply are connected, and similarly for the negative terminals of the Motor Board's motor power supply terminal and the +24V bench power supply. 5. Turn on both bench power supplies and command the Control Board via the PC to not attempt to move any motors. 6. Check that the measured ripple meets specifications, using measurement capabilities built into the oscilloscope.
3. Input of 120V AC.	1. Check the specifications of the bench power supply for proper certifications.

2.4 Tolerance Analysis

The most significant risk to the successful completion of this project is the mechanical design of the SCARA robotic arm, in particular, we require accurate positioning to produce a high-quality output with true edges instead of wavy lines. Using the same setup as in Section 2.3.1, the inverse kinematic equations for the SCARA robotic arm are as follows:

$$E = \cos^{-1}\left(\frac{X^2+Y^2-(L1)^2-(L2)^2}{2(L1)(L2)}\right) \tag{27}$$

$$S = atan2(Y,X) - \cos^{-1}\left(\frac{X^2+Y^2+(L1)^2-(L2)^2}{2(L1)\sqrt{X^2+Y^2}}\right) \tag{28}$$

To attain a resolution of 200 px x 200 px, each pixel is spaced $\frac{8.5}{200}\ in = 0.0425\ in$ apart in the y-axis and $\frac{11}{200}\ in = 0.055\ in$ apart in the x-axis, hence, the maximum permissible error in X, ΔX, is $\Delta X = \frac{\frac{11}{200} in}{2} = 0.0275\ in$ and the maximum permissible error in Y, ΔY, is $\Delta Y = \frac{\frac{8.5}{200} in}{2} = 0.0213\ in$. From [11], using differentials, we have

$$\Delta E = \frac{dE}{dX}\Delta X \tag{29}$$

where ΔE is the permissible error in Joint 2 angle for the given permissible error in X, ΔX.

$$\Delta E = \frac{dE}{dY}\Delta Y \tag{30}$$

where ΔE is the permissible error in Joint 2 angle for the given permissible error in Y, ΔY.

$$\Delta S = \frac{dS}{dX}\Delta X \tag{31}$$

where ΔS is the permissible error in Joint 1 angle for the given permissible error in X, ΔX.

$$\Delta S = \frac{dE}{dY}\Delta Y \tag{32}$$

where ΔS is the permissible error in Joint 1 angle for the given permissible error in Y, ΔY. We have

$$\frac{dE}{dX} = -\frac{\mathrm{X}}{(L1)(L2)\sqrt{1-\frac{\left(X^2+Y^2-(L1)^2-(L2)^2\right)^2}{4(L1)^2(L2)^2}}} \tag{33}$$

$$\frac{dE}{dY} = -\frac{\mathrm{Y}}{(L1)(L2)\sqrt{1-\frac{\left(X^2+Y^2-(L1)^2-(L2)^2\right)^2}{4(L1)^2(L2)^2}}} \tag{34}$$

$$\frac{dS}{dX} = -\frac{Y}{X^2+Y^2} + \frac{\frac{X}{L1\sqrt{X^2+Y^2}} - \frac{X(X^2+Y^2+(L1)^2-(L2)^2)}{2(L1)(X^2+Y^2)^{\frac{3}{2}}}}{\sqrt{1-\frac{\left(X^2+Y^2+(L1)^2-(L2)^2\right)^2}{4(L1)^2(X^2+Y^2)}}} \tag{35}$$

$$\frac{dS}{dX} = -\frac{X}{X^2+Y^2} + \frac{\frac{Y}{L1\sqrt{X^2+Y^2}} - \frac{Y(X^2+Y^2+(L1)^2-(L2)^2)}{2(L1)(X^2+Y^2)^{\frac{3}{2}}}}{\sqrt{1-\frac{\left(X^2+Y^2+(L1)^2-(L2)^2\right)^2}{4(L1)^2(X^2+Y^2)}}} \tag{36}$$

therefore

$$\Delta E = -\frac{\mathrm{X}}{(L1)(L2)\sqrt{1-\frac{\left(X^2+Y^2-(L1)^2-(L2)^2\right)^2}{4(L1)^2(L2)^2}}}\Delta X \tag{37}$$

$$\Delta E = -\frac{\mathrm{Y}}{(L1)(L2)\sqrt{1-\frac{\left(X^2+Y^2-(L1)^2-(L2)^2\right)^2}{4(L1)^2(L2)^2}}}\Delta Y \tag{38}$$

$$\Delta S = \left(-\frac{Y}{X^2+Y^2} + \frac{\frac{X}{L1\sqrt{X^2+Y^2}} - \frac{X(X^2+Y^2+(L1)^2-(L2)^2)}{2(L1)(X^2+Y^2)^{\frac{3}{2}}}}{\sqrt{1-\frac{(X^2+Y^2+(L1)^2-(L2)^2)^2}{4(L1)^2(X^2+Y^2)}}} \right) \Delta X \quad (39)$$

$$\Delta S = \left(-\frac{X}{X^2+Y^2} + \frac{\frac{Y}{L1\sqrt{X^2+Y^2}} - \frac{Y(X^2+Y^2+(L1)^2-(L2)^2)}{2(L1)(X^2+Y^2)^{\frac{3}{2}}}}{\sqrt{1-\frac{(X^2+Y^2+(L1)^2-(L2)^2)^2}{4(L1)^2(X^2+Y^2)}}} \right) \Delta Y \quad (40)$$

Using the bounds on X and Y, i.e. Equations 6-9, we get

Boundary Condition	Maximum Allowable ΔS (Joint 1 Error) for Given ΔX	Maximum Allowable ΔS (Joint 1 Error) for Given ΔY	Maximum Allowable ΔE (Joint 2 Error) for Given ΔX	Maximum Allowable ΔE (Joint 2 Error) for Given ΔY
$X = -5.5\ in, Y = 15\ in$	-0.0125 rad	0.0236 rad	0.0219 rad	-0.0462 rad
$X = 5.5\ in, Y = 6.5\ in$	-0.00117 rad	-0.000408 rad	-0.00262 rad	-0.00240 rad
$X = 5.5\ in, Y = 15\ in$	0.00931 rad	0.0226 rad	-0.0219 rad	-0.0462 rad
$X = -5.5\ in, Y = 6.5\ in$	$-0.0376\ rad$	$0.00282\ rad$	$0.00262\ rad$	$-0.00240\ rad$

From the above, the tolerance required is bounded by the smallest $|\Delta S|$ or $|\Delta E|$ found previously, i.e. $maximum\ angular\ error \leq 0.00424$ rad, as E and S do not change in concavity and has no saddle points, and hence neither do the derivatives of E and S change in concavity and have saddle points. The 23HS22-2804S stepper motors used have 200 steps per revolution and the L6470 is capable for 1/128 microstepping, i.e. 128 microsteps per motor full step. Additionally, as can be seen from the Figure 2, Arm Segment 1 is driven by a 1:3 ratio synchronous belt drive, while Arm Segment 2 is driven by two 1:3 ratio synchronous belt drive in series. Hence, the angular accuracy of Joint 1 is $\frac{2\pi\ rad}{200} \cdot \frac{1}{128} \cdot \frac{1}{3} = 8.18 \cdot 10^{-5}\ rad$ and the angular accuracy of Joint 2 $\frac{2\pi\ rad}{200} \cdot \frac{1}{128} \cdot \frac{1}{3} \cdot \frac{1}{3} = 2.73 \cdot 10^{-5}\ rad$, both of which fulfill the required angular accuracy. Although microstepping will result in a loss of torque, based on our design criterion, there is sufficient margin to ensure that any torque loss will not result in loss of functionality.

3. Cost and Schedule

3.1 Cost Analysis

3.1.1 Labor

Based on team members' personal experiences and research on the salary of the ECE and CS graduate students in the Urbana-Champaign area and the Bay area [12], we estimate an approximate salary of $50 per hour. Based on the time team members' have spent so far per week on the project, we estimate a workload of twenty hours per week for a total of ten weeks, giving a total labor time spent of two hundred hours per person. We compute the salary per person via the following formula as found on the ECE 445 website [13]:

$$Total\ Salary\ Per\ Person = Salary\ (Hourly) \times Hours \times Multiplier \tag{41}$$

Table 11: Labor Cost Estimate

Member	Salary (Hourly)	Hours	Multiplier	Total Per Person
Bingzhe Wei	$50.00	200	2.5	$25000
Chenghao Duan	$50.00	200	2.5	$25000
Tianhao Chi	$50.00	200	2.5	$25000
			Labor Total	$75000

3.1.2 Parts and Materials

Table 12: Bill of Materials

Part ID	Part Description	Manufacturer	Supplier	Assembly
ATmega1284P-PU	Atmel ATmega1284P IC MCU 8BIT 128KB FLASH 40DIP	Atmel	Digikey	Control Board
COM-00097	Mini Pushbutton Switch	Generic	Sparkfun	Control Board
COM-08571	Capacitor Ceramic 22pF 200V	Generic	Sparkfun	Control Board
N/A	5x PCB Control Board	Shenzhen2u	Shenzhen2u	Control Board
PRT-10811	DC Barrel Jack Adapter - Breadboard Compatible	Generic	Sparkfun	Control Board
SEN-00246	Optical Detector / Phototransistor - QRD1114	Fairchild Semiconductor	Sparkfun	Control Board
3296W-1-204LF	200 kOhms 0.5W, 1/2W PC Pins Through Hole Trimmer Potentiometer Cermet 25 Turn Top Adjustment	Bourns Inc.	Digikey	Motor Board
COM-00096	Electrolytic Decoupling Capacitors - 100µF/25V	Generic	Sparkfun	Motor Board

K103K15X7RF5TL2	10000pF ±10% 50V Ceramic Capacitor X7R Radial	Vishay	Digikey	Motor Board
K180J15C0GF5TL2	18pF ±5% 50V Ceramic Capacitor C0G, NP0 Radial	Vishay	Digikey	Motor Board
L6470H	L6470 Stepper Motor Controller IC HTSSOP28	STMicroelectronics	Digikey	Motor Board
N/A	3x PCB Motor Board	OSHPark	OSHPark	Motor Board
PRT-10571	Screw Terminals 2.54mm Pitch (2-Pin)	Generic	Sparkfun	Motor Board
UVP1H470MPD	47µF 50V Aluminum Electrolytic Capacitors Radial, Can 2000 Hrs @ 85°C	Nichicon	Digikey	Motor Board
UVR2AR22MDD	0.22µF 100V Aluminum Electrolytic Capacitors Radial, Can 2000 Hrs @ 85°C	Nichicon	Digikey	Motor Board
COM-00523	Electrolytic Decoupling Capacitors - 10µF/25V	Generic	Sparkfun	Motor Board/Control Board
COM-00536	Crystal 16MHz HC49/US	Generic	Sparkfun	Motor Board/Control Board
COM-08375	Capacitor Ceramic 0.1µF 50V	Generic	Sparkfun	Motor Board/Control Board
COM-08588	Diode Small Signal - 1N4148	Generic	Sparkfun	Motor Board/Control Board
COM-09590	LED (5mm) LED - Basic Red 5mm	Generic	Sparkfun	Motor Board/Control Board
COM-09592	LED (5mm) LED - Basic Green 5mm	Generic	Sparkfun	Motor Board/Control Board
COM-09594	LED (5mm) LED - Basic Yellow 5mm	Generic	Sparkfun	Motor Board/Control Board
COM-10969	Resistor Kit - 1/4W, Includes 0Ω, 1.5Ω, 4.7Ω, 10Ω, 47Ω, 100Ω, 220Ω, 330Ω, 470Ω, 680Ω, 1kΩ, 2.2kΩ, 3.3kΩ, 4.7kΩ, 10kΩ, 22kΩ, 47kΩ, 100kΩ, 330kΩ, 1MΩ	Generic	Sparkfun	Motor Board/Control Board

PRT-00115	Female Headers 0.1" 40 Position	Generic	Sparkfun	Motor Board/Control Board
PRT-00116	Break Away Headers – Straight 0.1" 40 Position	Generic	Sparkfun	Motor Board/Control Board
23HS30-2804S	NEMA 23 Stepper Motor 2.8A 1.9Nm (269oz.in) 76mm Length	Generic	HobbyUnlimited @ Amazon	Robotic Arm
60355K704	Ball Bearing, Sealed, Trade Number R8-2RS, for 1/2" Shaft Diameter	McMaster-Carr	McMaster-Carr	Robotic Arm
6484K228	XL Series Timing Belt, Trade No. 180xL037	McMaster-Carr	McMaster-Carr	Robotic Arm
6484K237	XL Series Timing Belt, Trade No. 280xL037	McMaster-Carr	McMaster-Carr	Robotic Arm
94613A550	100x Off-White Nylon 6/6 Hex Head Screws, 1/4"-20 Thread Size, 2" Long, Fully Threaded	McMaster-Carr	McMaster-Carr	Robotic Arm
94812A700	100x Nylon 6/6 Hex Nut 1/4"-20 Thread Size	McMaster-Carr	McMaster-Carr	Robotic Arm
95606A430	100x Nylon 6/6 Plastic Washer for 1/4" Screw Size, 0.312" ID, 0.5" OD	McMaster-Carr	McMaster-Carr	Robotic Arm
HPM# 07828	4x Compression Spring 5/16" x 1-1/2" x .020"	Hillman	Lowe's	Robotic Arm
ROB-09065	Servo - Generic (Sub-Micro Size)	Generic	Sparkfun	Robotic Arm
ROB-11884	Servo - Hitec HS-422 (Standard Size)	Hitec	Sparkfun	Robotic Arm
TOL-14584	PLA Filament 2.85mm - 1kg (Black)	Generic	Sparkfun	Robotic Arm

Table 13: Parts/Materials Cost Estimate

Part ID	**Part Description**	**Assembly**	**Quantity**	**Cost Per Unit**	**Cost Total Per Part**
ATmega1284P-PU	Atmel ATmega1284P IC MCU 8BIT 128KB FLASH 40DIP	Control Board	1	$5.68	$5.68
COM-00097	Mini Pushbutton Switch	Control Board	1	$0.35	$0.35

COM-08571	Capacitor Ceramic 22pF 200V	Control Board	2	$0.25	$0.50
N/A	5x PCB Control Board	Control Board	1	$25.84	$25.84
PRT-10811	DC Barrel Jack Adapter - Breadboard Compatible	Control Board	1	$0.95	$0.95
SEN-00246	Optical Detector / Phototransistor - QRD1114	Control Board	4	$0.95	$3.80
3296W-1-204LF	200 kOhms 0.5W, 1/2W PC Pins Through Hole Trimmer Potentiometer Cermet 25 Turn Top Adjustment	Motor Board	2	$2.41	$4.82
COM-00096	Electrolytic Decoupling Capacitors - 100µF/25V	Motor Board	4	$0.35	$1.40
K103K15X7RF5TL2	10000pF ±10% 50V Ceramic Capacitor X7R Radial	Motor Board	4	$0.21	$0.84
K180J15C0GF5TL2	18pF ±5% 50V Ceramic Capacitor C0G, NP0 Radial	Motor Board	4	$0.21	$0.84
L6470H	L6470 Stepper Motor Controller IC HTSSOP28	Motor Board	2	$7.72	$15.44
N/A	3x PCB Motor Board	Motor Board	1	$61.70	$61.70
PRT-10571	Screw Terminals 2.54mm Pitch (2-Pin)	Motor Board	8	$0.75	$6.00
UVP1H470MPD	47µF 50V Aluminum Electrolytic Capacitors Radial, Can 2000 Hrs @ 85°C	Motor Board	2	$0.64	$1.28
UVR2AR22MDD	0.22µF 100V Aluminum Electrolytic Capacitors Radial, Can 2000 Hrs @ 85°C	Motor Board	2	$0.24	$0.48
COM-00523	Electrolytic Decoupling Capacitors - 10µF/25V	Motor Board/Control Board	4	$0.45	$1.80
COM-00536	Crystal 16MHz HC49/US	Motor Board/Control Board	3	$0.95	$2.85
COM-08375	Capacitor Ceramic 0.1µF 50V	Motor Board/Control Board	16	$0.25	$4.00
COM-08588	Diode Small Signal - 1N4148	Motor Board/Control Board	5	$0.15	$0.75
COM-09590	LED (5mm) LED - Basic Red 5mm	Motor Board/Control Board	6	$0.35	$2.10
COM-09592	LED (5mm) LED - Basic Green 5mm	Motor Board/Control Board	5	$0.35	$1.75

COM-09594	LED (5mm) LED - Basic Yellow 5mm	Motor Board/Control Board	6	$0.35	$2.10
COM-10969	Resistor Kit - 1/4W, Includes 0Ω, 1.5Ω, 4.7Ω, 10Ω, 47Ω, 100Ω, 220Ω, 330Ω, 470Ω, 680Ω, 1kΩ, 2.2kΩ, 3.3kΩ, 4.7kΩ, 10kΩ, 22kΩ, 47kΩ, 100kΩ, 330kΩ, 1MΩ	Motor Board/Control Board	1	$7.95	$7.95
PRT-00115	Female Headers 0.1" 40 Position	Motor Board/Control Board	10	$1.50	$15.00
PRT-00116	Break Away Headers – Straight 0.1" 40 Position	Motor Board/Control Board	10	$1.50	$15.00
23HS30-2804S	Nema 23 Stepper Motor 2.8A 1.9Nm (269oz.in) 76mm Length	Robotic Arm	2	$24.50	$49.00
60355K704	Ball Bearing, Sealed, Trade Number R8- 2RS, for 1/2" Shaft Diameter	Robotic Arm	6	$8.74	$52.44
6484K228	XL Series Timing Belt, Trade No. 180xL037	Robotic Arm	1	$6.41	$6.41
6484K237	XL Series Timing Belt, Trade No. 280xL037	Robotic Arm	1	$8.05	$8.05
94613A550	100x Off-White Nylon 6/6 Hex Head Screws, 1/4"- 20 Thread Size, 2" Long, Full y Threaded	Robotic Arm	1	$8.00	$8.00
94812A700	100x Nylon 6/6 Hex Nut 1/4"-20 Thread Size	Robotic Arm	1	$6.47	$6.47
95606A430	100x Nylon 6/6 Plastic Washer for 1/4" Screw Size, 0.312" ID, 0.5" OD	Robotic Arm	1	$7.25	$7.25
HPM# 07828	4x Compression Spring 5/16" x 1-1/2" x .020"	Robotic Arm	1	$2.68	$2.68
ROB-09065	Servo - Generic (Sub-Micro Size)	Robotic Arm	3	$8.95	$26.85
ROB-11884	Servo - Hitec HS-422 (Standard Size)	Robotic Arm	1	$9.95	$9.95
TOL-14584	PLA Filament 2.85mm - 1kg (Black)	Robotic Arm	3	$25.00	$75.00
				Parts Total	$435.32

3.1.3 Grand Total

$$Grand\ Total = Labor\ Cost + Parts\ and\ Materials\ Cost$$

$$= \$75000 + \$435.32 = \$75435.32 \quad (54)$$

3.2 Schedule

Please note that we have already completed all circuit design and PCB design required for our project, and have also bought all electrical components needed, so those elements have been excluded from the schedule below.

Table 14: Schedule for Rest of Semester

Week	Objective	Bingzhe Wei	Chenghao Duan	Tianhao Chi
Week 1 (Feb 19th)	Complete PC Software, Finalize 3D Models	Finalize 3D models of SCARA robotic arm's base and arm segments	Finalize 3D models of SCARA robotic arm's transmission and end effector	Implement all PC side image processing algorithms
Week 2 (Feb 26th)	Assemble Control Board, Start on Microcontroller Software	Implement microcontroller-Motor Board communication and status LED functionality	Solder and verify Control Board functionality	Implement PC-microcontroller communication and limit switch reading functionality
Week 3 (Mar 5th)	Finish Microcontroller Software, Print all Physical Parts for Robot Arm	Integrate and test microcontroller software	Print all physical parts needed at UIUC MakerLab, and buy needed mechanical components.	Integrate and test microcontroller software
Week 4 (Mar 12th)	Assemble Both Motor Boards, Test Software	Solder and verify one Motor Board's functionality	Solder and verify one Motor Board's functionality	Integrate and test microcontroller software with actual Motor Boards connected
Week 5 (Mar 19th) Spring Break	Integrate all Components	Assemble the base and transmission and integrate all other assemblies assembled by other group members	Assemble the end effector assembly	Assemble the arm segments
Week 6 (Mar 26th)	Testing and Fine-Tuning	Motion test – tune microcontroller software as needed to allow accurate robot movement	Motion test - tune mechanical design as needed to allow accurate robot movement	Motion test - tune PC image processing algorithms as needed to allow accurate robot movement
Week 7 (April 2nd)	Testing and Fine-Tuning	Full system integration test – test	Full system integration test – test	Full system integration test – test

		drawing capabilities and debug as needed	drawing capabilities and debug as needed	drawing capabilities and debug as needed
Week 8 (April 9th)	Testing and Fine-Tuning	Tune PC style transfer network for best artistic effect	Continue tuning algorithms to increase robot arm movement speed as much as possible	Tune clustering algorithms for best artistic effect
Week 9 (April 16th)	Testing and Fine-Tuning	Train additional style transfer networks	Continue tuning algorithms to increase robot arm movement speed as much as possible	Tune vectorization algorithms for best artistic effect
Week 10 (April 23th)	Work on Final Report	Work on Final Report, focusing on electrical components of the project	Work on Final Report, focusing on mechanical components of the project	Work on Final Report, focusing on software components of the project
Week 11 (April 23th)	Rehearsal and Preparation Work on Final Presentation	Prepare Powerpoint slides and materials and procedures for a demonstration of robot's drawing capability to be shown at during the Final Presentation, and rehearse for the Final Presentation	Prepare Powerpoint slides and materials and procedures for a demonstration of robot's drawing capability to be shown at during the Final Presentation, and rehearse for the Final Presentation	Prepare Powerpoint slides and materials and procedures for a demonstration of robot's drawing capability to be shown at during the Final Presentation, and rehearse for the Final Presentation

4. Ethics and Safety

There are several potential safety hazards with our project, the primary being possible impact and pinch hazards with regards to the moving parts of the robotic arm. We plan to mitigate this by utilizing the stall detection mechanism built into the stepper motor drivers to automatically stop the robot in case arm movement is not as expected, which, given our use case, would indicate a collision with an unexpected entity in the robotics arm's workspace, and also enclosing the transmission and other mechanical components of the robotic arm as much as possible to prevent possible pinch hazards, while the workspace will be covered with a removable transparent shield during robot operation.

There is also a risk with that the stepper motor driver ICs or PCB traces may overheat due to an overcurrent condition either caused by continual high-power motor operation or a short during operation, creating a potential burn and fire hazard. We plan to mitigate this by power supply maximum current limits, overcurrent and over temperature automatic shutdown protocols implemented with the microcontroller and sensors embedded in stepper motor controller ICs, and appropriate PCB design, with suitable heatsinking and trace sizing and spacing, as well as isolation by placement in a separate board from the Control Board and a heat resistant physical enclosure.

It is our responsibility to make sure our project is consistent with the IEEE Code of Ethics [14]. With regards to the potential safety hazards above, as is consistent with Items 1 and 6, we will implement the above mitigation procedures and provide appropriate warnings, warning labels/markings, and training for users of our system. Since our project mainly just serves an artistic purpose, and is unlikely to be a part of critical infrastructure, our project is very unlikely to harm humanity or society in any form, and thus consistent with Item 9. As a non-sponsored project and with group members not having any particular relevant conflicts of interests, our team fulfills Items 2 and 4. As stated earlier in the introduction, we seek explore and to advance the state of the art of artificial intelligence and its interactions with society, as is consistent with Item 5. Finally, as engineers, all the members of our team will develop this project under a code of utmost integrity, mutual support and nondiscrimination, fulfilling Items 3, 7, 8, and 10.

References

[1] S. K. Saha, Introduction to Robotics, New Delhi: Tata McGraw-Hill, 2008.

[2] The Franklin Institute, "Maillardet's Automaton," 2018. [Online]. Available: https://www.fi.edu/history-resources/automaton. [Accessed 8 February 2018].

[3] S. Colton, "The Painint Fool," 2018. [Online]. Available: http://www.thepaintingfool.com. [Accessed 8 February 2018].

[4] H. Cohen, "The Further Exploits of AARON, Painter," *Stanford Humanities Review,* vol. 4, no. 2, 1995.

[5] S. Monnier, "Algorithmic worlds," 12 September 2014. [Online]. Available: http://www.algorithmic-worlds.net/. [Accessed 2 February 2018].

[6] Center of Excellence Digital Art, "compArt daDA: the database Digital Art," [Online]. Available: http://dada.compart-bremen.de. [Accessed 8 February 2018].

[7] R. Verostko, "The Algorists," 2011. [Online]. Available: http://www.algorists.org/algorist.html. [Accessed 8 February 2018].

[8] A. Krizhevsky, I. Sutskever and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *25th International Conference on Neural Information Processing Systems*, Lake Tahoe, NV , 2012.

[9] K. Hornik, "Approximation Capabilities of Multilayer Feedforward Networks," *Neural Networks,* vol. 4, no. 2, pp. 251-257, 1991.

[10] MIT edX, "2.008x: Fundamentals of Manufacturing Processes," [Online]. Available: https://www.edx.org/course/fundamentals-manufacturing-processes-mitx-2-008x-0. [Accessed 8 February 2018].

[11] M.-A. Belabbas, "ECE 486: A note on sensitivity," 2015. [Online]. Available: https://courses.engr.illinois.edu/ece486/fa2017/documents/ECE486note2_edited.pdf. [Accessed 26 Feburary 2018].

[12] College of Engineering, University of Illinois at Urbana-Champaign, "Salary Information 2015-2016," 2016. [Online]. Available: https://engineering.illinois.edu/documents/Salary.Info.Sheet.pdf. [Accessed 26 Feburary 2018].

[13] ECE 445 Course Staff, "Design Document," 2018. [Online]. Available: https://courses.engr.illinois.edu/ece445/guidelines/design-document.asp. [Accessed 26 Feburary 2018].

[14] IEEE, "IEEE Code of Ethics," 2018. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed 8 February 2018].