NORTHWESTERN UNIVERSITY

The Design and Control of an Actuated Prosthetic Elbow

A DISSERTATION

SUBMITTED TO THE GRADUATE SCHOOL

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

for the degree

MASTER OF SCIENCE

Field of Mechanical Engineering

By

Michael Jenz

EVANSTON, ILLINOIS

June 2026

ABSTRACT

The Design and Control of an Actuated Prosthetic Elbow

Michael Jenz

Whole-body angular momentum regulation requires reciprocal arm swing during walking, however, for individuals with upper limb loss this regulation is disrupted. This project aims to develop an actuated prosthesis to reintroduce arm swing during walking and restore whole-body angular momentum regulation. This work covers the entire design process and presents a new prototype that can imitate normal elbow arm swing during walking at variable speeds. The device, Prototype 3, actively adapts to changes in user walking speed using heel strike to align prototype arm swing with user intention. Results show Prototype 3 tracks elbow arm swing trajectories with average error of 1.5°. Prototype 3 tested under the condition of using a spring showed reduced average error and energy expenditure with a spring, establishing it as a critical component in the design of such systems. Finally, results conditioned on the weight of the controlled prosthesis show the estimated torque required for this system is higher than expected.

Acknowledgements

I would like to thank my advisors, Professors Matt Elwin and Matty Major, for the opportunity to work under their guidance and mentorship.

I would also like to thank the CRB lab members that have helped me over the past couple years. Thank you, Luke Batteas, for the extra time you offered to help me integrate NUControl into the project.

Finally, I would like to thank my family and friends for their support during my studies at Northwestern. I could not have made it through without you all.

Table of Contents

List of Tables

List of Figures

CHAPTER 1

Introduction

1.1. Background

Previous research has shown that periodic, reciprocal arm swing during walking is a critical because it balances generated contralateral leg momenta [**1**]. Furthermore, since individuals with unilateral upper limb absence experience bilateral asymmetry in both upper extremity body mass and motion [**2, 3, 4**] the whole-body angular momentum regulation is affected by upper limb absence [**5, 6**]. The high rate of falls in persons with upper limb absence that occur during walking, found to be 67%, is likely tied to this asymmetry in whole-body angular momentum regulation [**7**].

Individuals with upper arm limb absence exhibit a small amount of affected limb arm swing during walking —nearly a third of the intact limb [**2, 3, 4**]. Conventional prostheses do not restore the upper limb arm swing movement required to restore whole-body angular momentum regulation [**3, 6**]. Furthermore, the range of motion of the affected side during walking is the same regardless of if a conventional arm prosthesis is worn [**2, 3**].

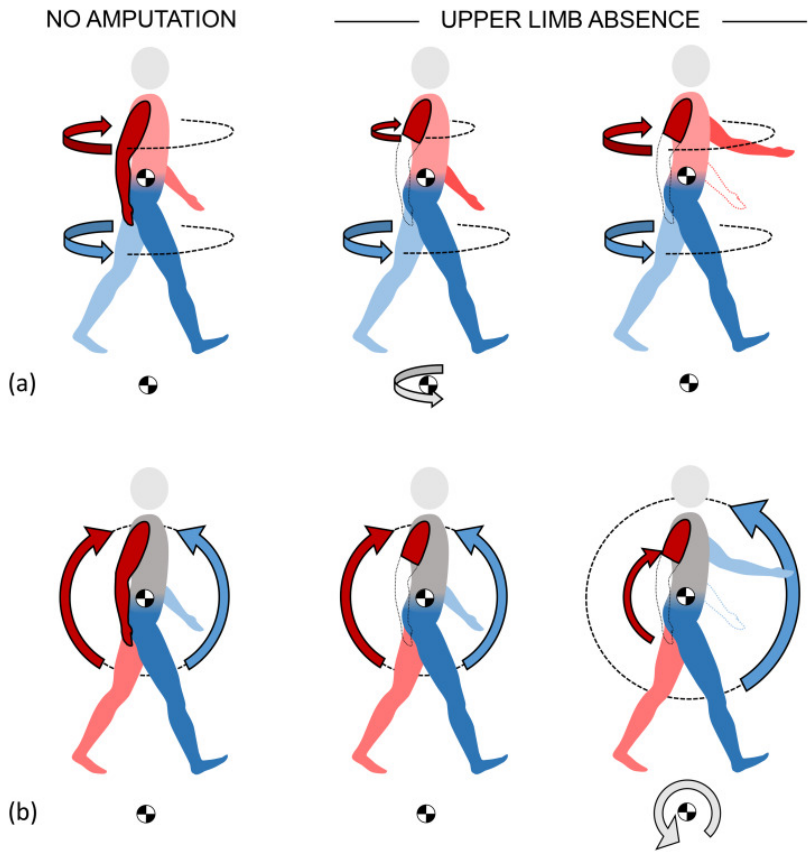


Figure 1.1. The top and bottom rows show the angular momentum generation from left and right sides in the transverse and sagittal planes respectively. The left most column shows individuals with intact arms, and the right two columns show individuals with unilateral upper limb loss. The right most column shows the effect of over-extension of the intact limb to balance transverse plane angular momentum generation and dissipation and the resulting imbalance in the sagittal angular momentum regulation. [**5**].

Figure 1.1 encapsulates the information presented as it presents the effect of upper limb loss on whole-body angular momentum regulation and how the available biomechanical remedy results in other asymmetries in the whole-body angular momentum. The reader can further internalize the effect of walking with asymmetric whole-body angular momentum generation by walking without arm swing on one or both sides.

1.2. Motivation and Assumptions

These findings have motivated this project, with the goal of developing an actuated prosthesis to restore the arm swing required to balance whole-body angular momentum regulation [**8**]. Actuation is required to inject energy into the dynamical system to reintroduce these stereotypical arm swings because individuals with upper limb loss to not show the same range of motion in their affected limb. Therefore, we have assumed that by replicating typical arm swing at the elbow joint via an actuated prosthesis, sufficient energy is introduced to restore stereotypical arm swing and therefore restore whole-body angular momentum generation. While other methods of restoring whole-body angular momentum generation are possible, without experimental data it is impossible to determine if this or other methods produce better results. Therefore, this project focused on the design and control of a prototype transhumeral prosthetic arm for individuals that generates typical arm swing trajectory behavior during walking at variable speeds.

CHAPTER 2

Mechanical Design

2.1. Motor Selection

2.1.1. Lessons from Prototypes 1 and 2

The methods and results that lead to the major learnings from Prototypes 1 and 2 are covered in depth in Appendix A, however, these key takeaways are worth reviewing here as they were key factors in the motor selection for Prototype 3.

The simulation results in this following section refer to the required elbow torque derived through a dynamic simulation of an arm as a double pendulum. This simulation is described in more depth in Appendix A.

2.1.1.1. Motor Torque Requirements. Prototype 1 used position control to track an arm swing trajectory during normal walking and was capable of following this trajectory signal with low error. The actual torque required by Prototype 1 is contrasted with the theoretical required torque as derived from the dynamic simulation of a double pendulum in Figure 2.1 below.

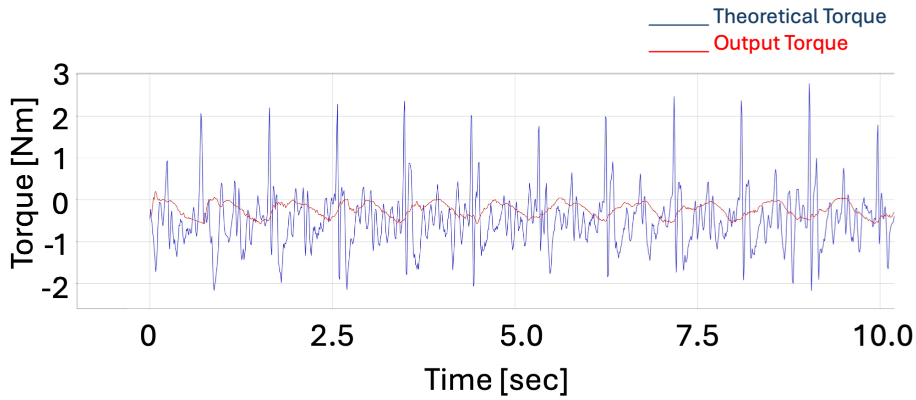


Figure 2.1. Theoretical vs. Actual Torque required by Prototype 1 at 1.4m/s. The actual torque as shown in red is periodic and far smaller in magnitude (amplitude of $0.25Nm$) in great contrast to the high and erratic torque that is theoretically required as shown in blue. These results contrast the torque requirements from two different approaches for control the system [**9**].

The motor used in Prototype 1 is a Cubemars G60 that weighs $226g$ and has a peak torque output of $1.75Nm$ [**10**]. The amplitude of the periodic torque signal is roughly $0.25Nm$. This is far smaller than the torque that the G60's peak torque specification indicating that the system was not torque limited. These results above in Figure 2.1 suggest that the torque magnitude required to track an arm swing trajectory are smaller than previously hypothesized based on simulation results. Deeper analysis of the final results from Prototype 1 as seen in Table 2.1 show that this proposed torque specification of $0.25Nm$ is valid.

Some trials from these results indicate a higher torque amplitude required. These results, however, also show nearly double the amount of error as other trials, hinting that unfavorable dynamics may have been present that made control of the system more difficult and thus require higher amplitudes of torque.

Table 2.1. Required torque output for position trajectory tracking from Prototype 1 at various walking speeds with error [**11**].

Walking Speed [m/s]	**0.7**	**0.8**	**1.0**	**1.1**	**1.4**
Torque Amplitude [Nm]	0.125	0.5	0.25	0.5	0.25
Torque Vertical Shift [Nm]	-0.25	-0.30	-0.25	-0.25	-0.25
Peak Error* [deg]	2.0	4.0	2.0	5.0	2.5

The periodic torque profiles are not zero mean, resulting in the actual required motor torque output to be equal to the signal amplitude added to the signal vertical shift. To keep the required motor torque as low as possible, the periodic signal needs to be shifted to be zero mean. This can be done, as supported by previous work in simulation.

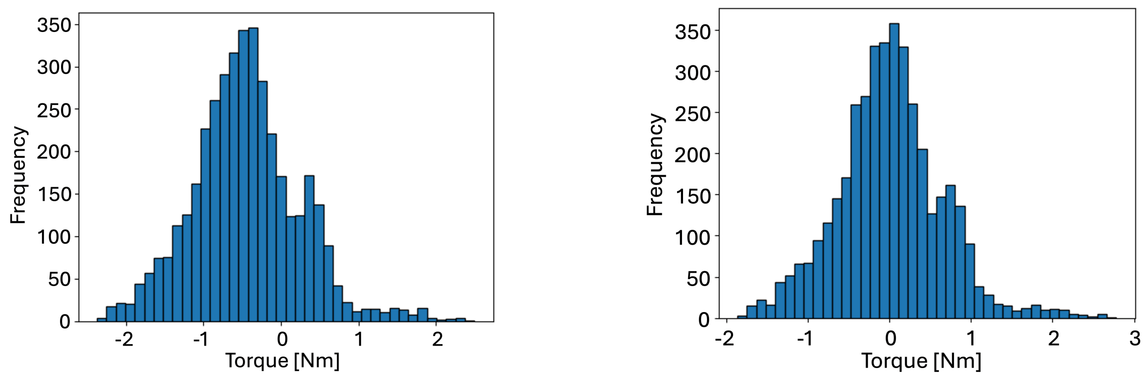


Figure 2.2. Comparison of histograms of theoretical motor output torque over a 40s trial collected at 100Hz with a bucket size of 0.125Nm. The results on the left are as is from the double pendulum simulation. The results on the right incorporate a torsional spring with stiffness $k = 0.5\frac{\text{Nm}}{\text{rad}}5$, which shifts the distribution's center toward zero, moving much of the torque output into the $\pm$1Nm range [**9**].

2.1.1.2. Spring Element to Reduce Torque Requirement. These results as shown in Figure 2.2 demonstrate that with the use of a spring, the torque profiles required to mimic arm swing is shifted to have zero mean. While these results are under the assumption of a torque control approach, as opposed to position control, they still support the idea that the torque profile can be shifted using a spring. Thus, this idea may be applied to the torque profiles observed from Prototype 1 and limit the required torque to the amplitude of the periodic signal as previously suggested.

2.1.1.3. Motor Speed Requirements. Figure 2.3 shows the required angular arm speed at the elbow via two separate analyses. Both theoretical and experimental results support the speed requirement for elbow arm swings to be at maximum $4.8\frac{rad}{s}$ or $46rpm$.

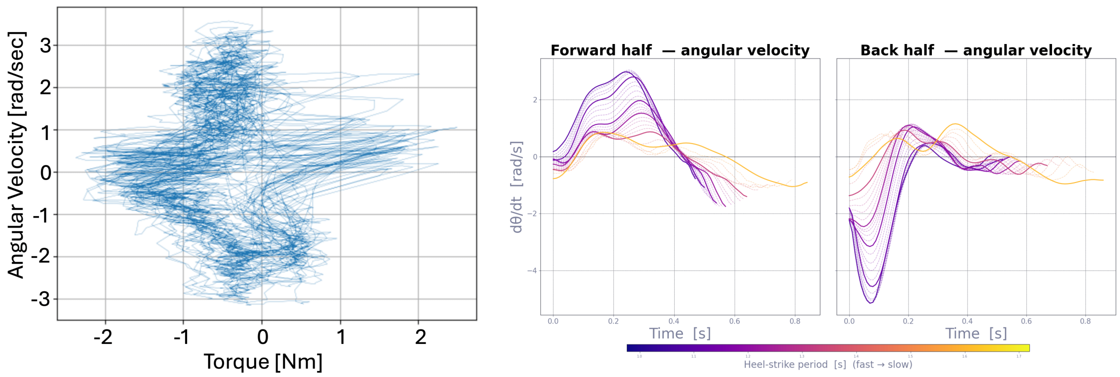


Figure 2.3. Theoretical and experimental results for angular velocity requirements. On the left is the torque speed curve derived from the double pendulum simulation showing maximum of $3.5\frac{rad}{s}$ required. On the right is elbow velocity derived from data collection at varied speeds between $0.5\frac{m}{s} - 1.5\frac{m}{s}$ showing maximum required elbow angular velocity as $4.8\frac{rad}{s}$ [**9**].

2.1.1.4. Transmission Requirement. The last relevant finding from previous work is that gearboxes with high gear ratios introduce difficult-to-control friction into the system. These results are shown in Figure 2.4 which displays bench testing results from Prototype 2. These results clearly show the adverse effects of static friction within the system working directly against the tracking of a sinusoidal signal.

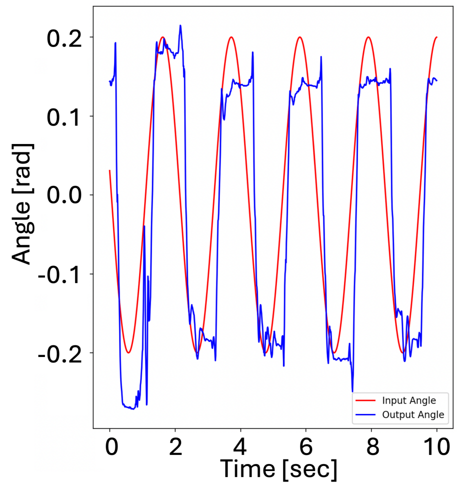


Figure 2.4. Bench testing results from Prototype 2 following a sinusoid. Prototype 2 is unable to follow these movements due to the gearbox friction resulting from the high gear ratio in its servo motor. This inability to follow precise movements, in addition to the added control challenge suggests using a highly geared motor is not possible for this project despite possible weight reduction [**9**].

These results show that future iterations should not include highly geared transmissions to avoid similar challenges.

2.1.2. Motor Requirements Summary

Given the lessons learned from previous work, the motor for this new prototype should satisfy the following requirements:

(1) $0.25Nm$ continuous torque

(2) $46rpm$ continuous speed

(3) Direct drive

(4) Minimal weight and size [**12**]

Although it was not explicitly found that a direct drive motor is more advantageous than a motor with a smaller gear ratio, this requirement is in line with the finding that the injection of friction into the system makes it more difficult to control. Therefore, it follows that a motor without gearing will introduce the least amount of friction possible to the system.

2.1.3. The Motor Selected

The Cubemars GL40 was selected as it satisfies all the selection criteria while being small and relatively lightweight. Motor specifications and resources for the GL40 are provided in Appendix B [**13**].

2.2. Mechanical Design

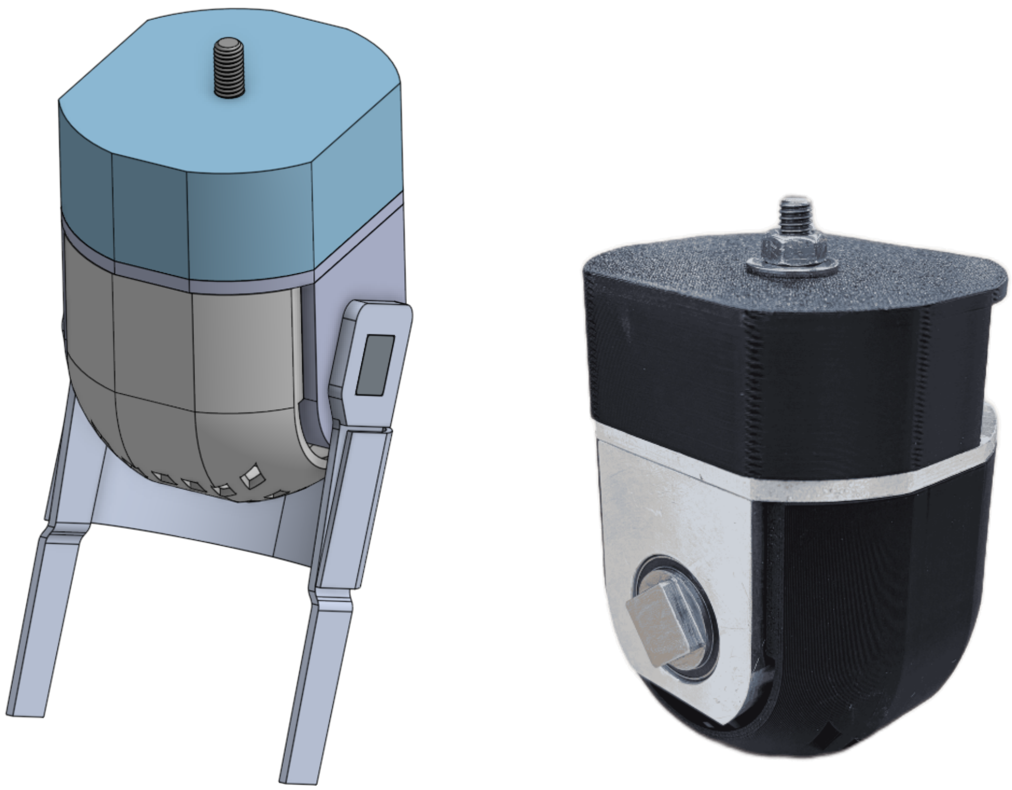


Figure 2.5. Complete elbow assemblies. On the left is the full assembly as designed in CAD. On the right is the manufactured and assembled elbow joint.

2.2.1. Design Considerations and Goals

The mechanical design of the elbow joint was driven by the following design choices: the torque transmission method, the motor selection, and the minimization of friction and weight. The torque transmission method selected is a bracket with square holes to transmit torque from the elbow joint to the prosthetic arm as used in the Fillauer

E-400 elbow [**14**]. This choice of bracket drives the width of the elbow joint, as well as the geometry of the motor shaft. The motor drives the design of the motor mounting, as well as the geometry of the motor flange shaft which connects the motor output to the transmission bracket. Finally, the goals to reduce friction and minimizing weight to respectively make the system easier to control and more comfortable for the user drove the rest of the design process.

2.2.2. Bill of Materials

Table 2.2 below lists all the parts required to build one assembly of Prototype 3. The bill of materials does not include stock required to machine the components for the elbow joint.

Table 2.2. Bill of Materials

Part	**Qty**	**Use**	**Cost [USD]**
Motor	1	Actuate prosthesis	97.99
Bearing	2	Constrain motor flanges	70.88
M3×6mm countersink screws	4	Motor flange to motor	0.22
M3×8mm countersink screws	4	Motor to motor mount	0.23
4-40, 3/8" screws	6	Motor side housing to top piece	0.75
Thin-profile hex nut	1	Nut for user mounting screw	0.05
Washer	1	Washer for user mounting nut	5.94
M6×40mm screw	1	Screw for mounting user to prosthetic	1.00
4-40, 5/8" screws	2	PCB mounting	0.26
4-40 nylon nuts	2	PCB mounting	0.25
Total			**177.61**

2.2.3. Elbow Design Breakdown

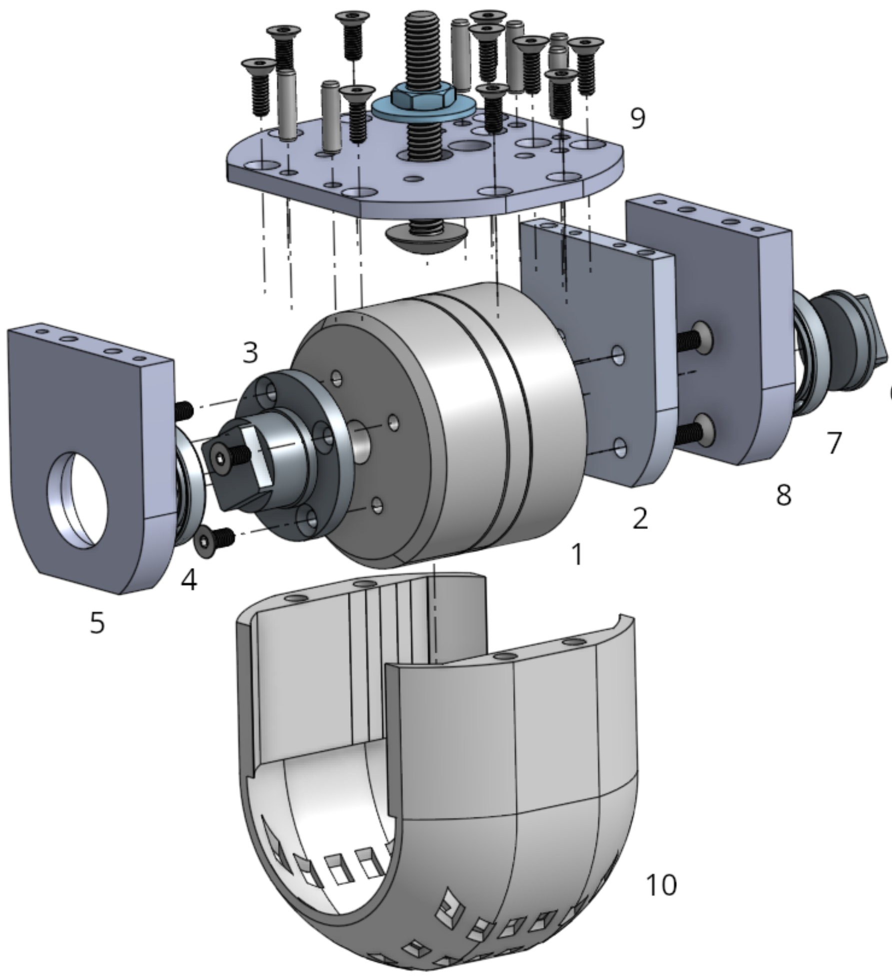


Figure 2.6. Exploded view of the elbow design. The motor (1) is seated in the center of the elbow joint, mounted to the motor mounting plate (2) via countersink screws. The motor output is connected to the bracket (not shown) by the motor flange shaft (3) that is seated in a bearing (4) within the elbow joint side housing (5). This design is mirrored behind the motor (6-8), however, the flange shaft (6) is not connected to the motor. The side housings and motor mounting plate (5, 8, 2) are fixtured to the top plate (9) via screws and located using dowel pins. Finally, the motor housing (10) protects the user from the moving internal components.

The mechanical design for Prototype 3 is broken down into two parts. The first, shown in Figure 2.6, contains the critical mechanical components for the elbow. The Onshape CAD and drawings can be found in Appendix C.

2.2.4. Electronics Mounting and Housing

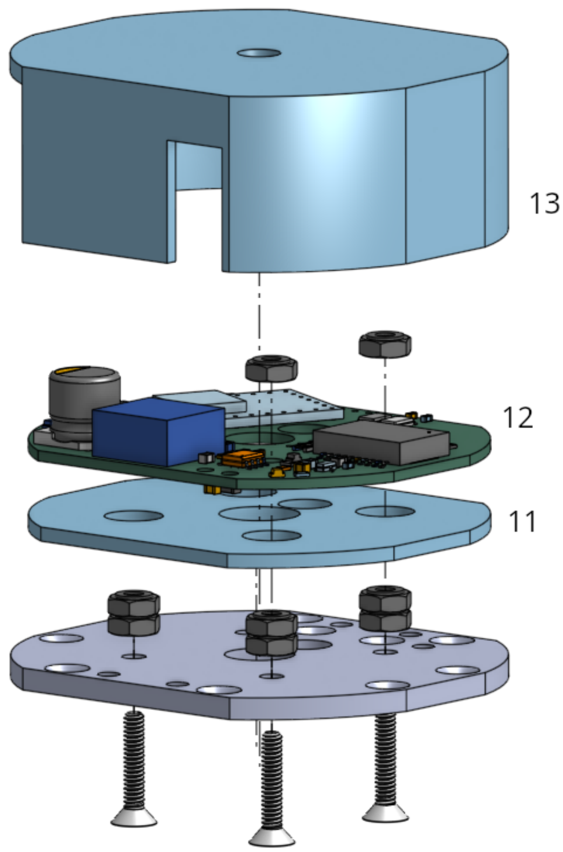


Figure 2.7. Exploded view of the electrical mounting design. This assembly connects to the top of the elbow joint via the top plate (9). The insulation layer (11) keeps the elbow joint electrically insulated from the PCB (12). The electronics housing (13) protects the PCB and has a hole for the mounting screw to pass through as do all the other components shown (9, 11, 12).

The electronics mounting and housing can be seen in Figure 2.7. The electronics housing is the surface at which the user interfaces with the prosthetic elbow joint via the mounting screw shown in Figures 2.5, 2.6. This screw passes through the electronics components and housing. The thin top to the electronics housing acts as a diaphragm spring which maintains the elbow joints fixturing during use.

2.2.5. Other components

External to the elbow joint there are three other components of Prototype 3. These include the prosthesis itself (along with the mounting bracket), the spring as described previously, and the intact arm mounting. Prototype 3 uses a traditional plastic prosthesis similar to that of Prototype 2 with a steel hook end-effector. It was found later in testing that the mass of this prosthesis was too great to control, thus the end-effector was removed in testing to reduce the mass of the prosthesis. The spring used to shift the torque profile to be zero mean was a rubber band terminated on the user mounting and the bottom part of the prosthesis as seen in Figure 2.8. Finally, the intact arm mounting was used for testing with people with intact limbs. This design is included in the full Onshape CAD linked in Appendix C and uses velcro to strap the Prototype to the user.

2.2.6. Design Comparison

Prototype 3 is a culmination of results from Prototypes 1 and 2 as covered in Appendix A. This is clear in a side by side comparison of the three prototypes thus far. In Figure 2.8 we can see that Prototype 3 is an iteration on the form factor of Prototype 2. Prototype 3 has the most complete design thus far with a custom PCB that is fully enclosed in the elbow joint. It also uses a traditional prosthesis, similar to Prototype 2. The electronics and motor choice are akin to that of Prototype 1.

Figure 2.8. Comparison of Prototypes 1, 2, and 3 from left to right.

Prototype 3 is lighter than Prototype 1 but heavier than Prototype 2. Prototype 2 is lighter since the motor and battery were much smaller than in both Prototypes 1 and 3. The battery was not attached to the prosthesis in any of these Prototypes, so the swinging mass in each of these cases is equivalent to the prosthesis weight. A comparison of prototype masses and components is shown in Table 2.3

Table 2.3. Prototype Comparison

	Prototype 1	**Prototype 2**	**Prototype 3**
Motor	BLDC: Cubemars R60	Servo: HiWonder Coreless	BLDC: Cubemars GL40
	Peak torque = 1.75 Nm	Peak torque = 3.4 Nm	Peak torque = 0.5 Nm
	Gear ratio = 1:1	Gear ratio = 342:1	Gear ratio = 1:1
	Drive method = ODrive	Drive method = PWM	Drive method = FOC
Weight	Total = 1538 g	Total = 855 g	Total = 1079 g
	Motor = 248 g	Motor = 60 g	Motor = 107 g*
	Prosthesis = 550 g	Prosthesis = 400 g	Prosthesis = 352 g**
	Electronics/Housing = 320 g	Electronics/Housing = 225 g	Elbow Joint = 386 g
	Battery = 420 g	Battery = 170 g	Battery = 256 g

* Not included in total since motor weight accounted for in elbow joint weight.

** Does not include end-effector of mass 85g.

CHAPTER 3

Electrical Design

3.1. Electronic System Design

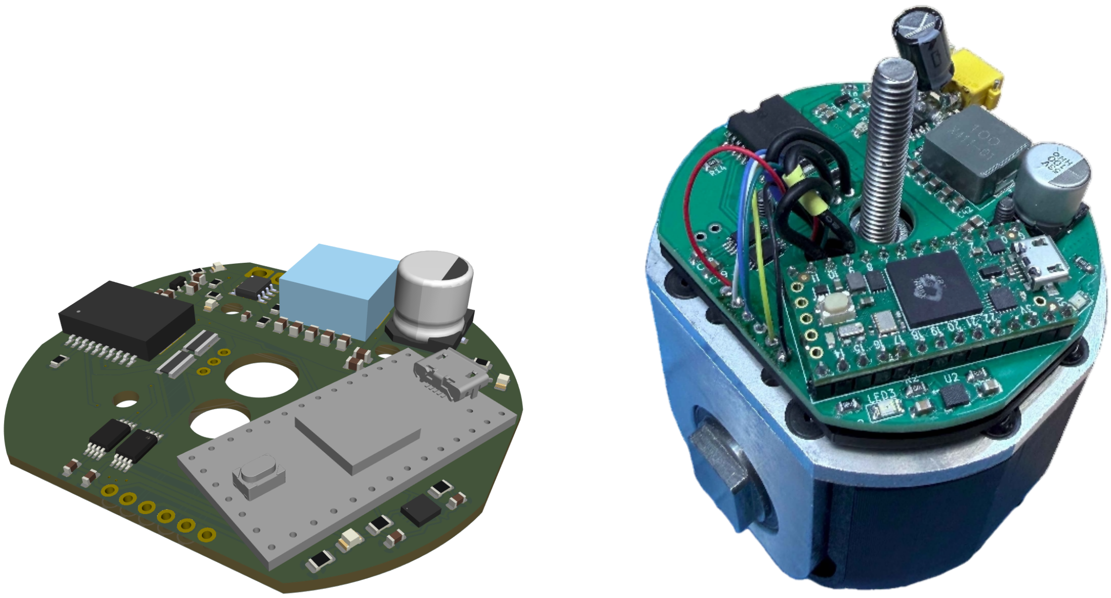


Figure 3.1. Full PCB assemblies. On the left is the visualization of the PCB design. On the right is the manufactured version of the PCB mounted on the elbow assembly.

3.1.1. Design Requirements

The electronic system for Prototype 3 must run the controls algorithms, drive the brushless motor, and collect acceleration feedback. The system needs to be remotely powered using a battery and should be self-contained, not requiring connection to external compute via wire or wireless connection. The electronic system should fit within the elbow joint and be electrically isolated from the elbow joint.

3.1.2. Bill of Materials

The components required for one PCB are listed below in Table 3.1 not including the cost of the PCB itself.

Table 3.1. Electrical Bill of Materials

Part	**Qty**	**Use**	**Cost [USD]**
Microcontroller			
Teensy 4.0	1	Microcontroller	24.17
Power			
6S 1500 mAh LiPo battery	2	Power supply	63.89
LMR33640ADDAR	1	Buck converter, adjustable 4A	2.41
TLV76133DCYR	1	3.3 V LDO regulator	0.49
TLV76150DCYR	1	5 V LDO regulator	0.49
SCIHP1367-100M	1	10 μH power inductor	1.85
Battery voltage monitor	4	Monitor battery level	10.89
Motor Drive & Sensing			
L6234PD	1	3-phase BLDC motor driver	7.67
INA240A2PW	2	Current sense amplifier	6.66
RMCF0805FT15TR	2	15 mΩ current sense resistor	0.17
Sensing			
LSM6DSVTR	1	IMU	4.05
Passives			
Capacitors (various)	26	Decoupling, filtering, bulk storage	7.79
Resistors (various)	13	Biasing, feedback, LED limiting	1.10
Diodes (various)	1	Status LED's and Shottkey diodes	0.96
Total			**132.59**

3.1.3. Schematic Analysis

The schematic along with other documentation for the electronic system design is located in Appendix D. The schematic contains five subsystems: the MCU, power supply and regulation, current sense, motor driver, and IMU. The design of the MCU and IMU subsystems was straightforward in that the required connections were driven by pin assignments and datasheet recommendations. The design for the power supply and regulation was more involved since both the voltage divider and capacitance required calculations guided by the datasheet. The electrical system design for the current sense and motor driver subsystems was inspired largely in part by the SimpleFOC Shield v2 [**15**]. The current sense resistor value needed to be tuned for the Teensy 4.0's ADCs and the calculations for this can be found in Appendix D.

3.1.4. PCB Layout Analysis

The layout of the PCB is designed to fit within the prosthetic arm and thus the form of the PCB is in the same shape as the top plate as seen in Figure 3.2. The PCB has mounting holes in three locations, however, the through hole that is close to the pins of the Teensy 4.0 is inaccessible leaving the PCB unconstrained. The main concern with this is that the IMU signal would be corrupted by extra PCB movement, but this never became an issue. The PCB also has two larger through holes for the prosthesis mounting screw and for motor leads and encoder wires to pass through to the top of the PCB, as visible in Figure 3.1.

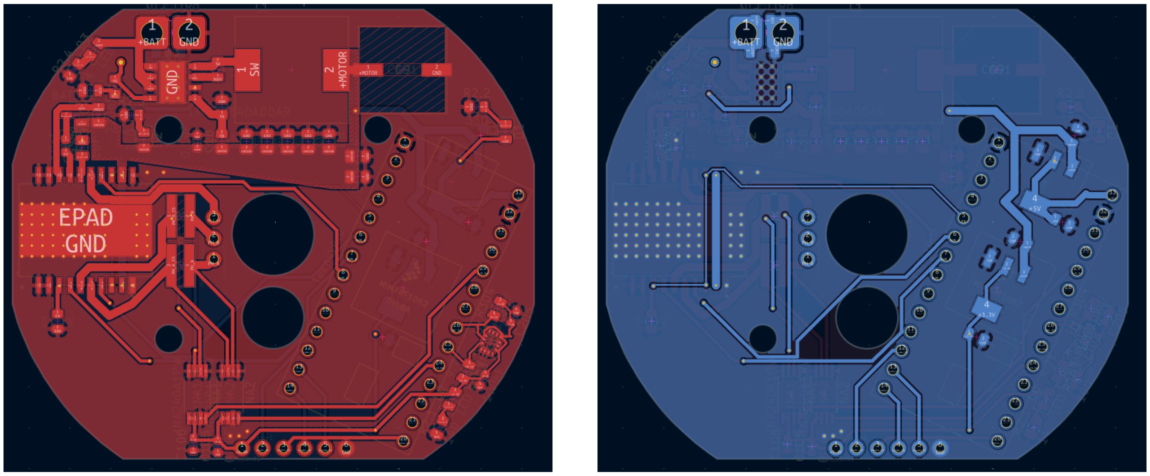


Figure 3.2. The PCB layout. The top side of the PCB is on the left and the bottom side is on the right.

3.1.5. Battery Selection

The goal of battery selection is to balance the competing interests of minimizing and maximizing the continuous operation time. This means we need a battery with high energy density, leading directly to lithium-polymer (lipo) batteries. I compared a set of these lipo batteries against one another to find the battery with the best capacity to weight ratio. To compare, I tested how long each battery would last while operating at the motors maximum continuous current draw after DCDC conversion. This is in essence comparing the power capacity of each battery. I found that a high voltage battery with lower capacity balances well the minimization of weight with longevity of operation time. I selected a 6S lithium-polymer battery with 1500mAh of capacity which can power the system at maximum continuous torque for 1.28 hours. The full lipo comparison is provided in Appendix D.

CHAPTER 4

Data Collection

The methodology for replicating arm swings with these actuated prototypes has been using previously collected arm swing trajectories as command signals for a motor position controller. This requires having arm swing trajectories to use. This section details the collection process used to obtain these elbow angle trajectories. The full implementation of this process is provided in Appendix E.

4.1. Collection Procedure

Arm swing trajectories were collected with Xsens motion capture IMUs. These sensors collect acceleration and orientation data which is used to extract elbow angle using the following procedure. This procedure requires two IMUs, one located in the middle of the upper arm and one located in the middle of the lower arm. Parts of this decomposition process are inspired by the previous work on lower limb angle decomposition [**16**].

4.1.1. Calibration

A calibration must be derived to extract the elbow angle from the relative orientation of the two IMUs. In practice this means extracting both the axis of rotation of the elbow, which is a revolute joint with a single degree of freedom, and the relative orientation of the IMUs at full extension called the neutral offset. With this information elbow angle can be extracted from the IMU dataset.

4.1.1.1. Elbow Axis of Rotation. First, the elbow axis of rotation needs to be identified. The participant should stand upright with their arm at their side and their palm inward with their arm fully extended at start. Then, they should make three slow sweeps from full elbow extension to 90° elbow flexion. The result of this procedure is shown in Figure 4.1.

The data can then be processed using the procedure described below in Equations 4.1- 4.4 where $r(t)$ represents a rotation vector (Euler vector) representation of an orientation at time t. This procedure omits some filtering and other representation-specific implementation details that can be found in Appendix E.

$$r_{\text{rel}}(t) = r_{\text{upper}}(t)^{-1} \cdot r_{\text{lower}}(t) \tag{4.1}$$

$$R_{\text{rel}} = \begin{bmatrix} r_{\text{rel}}(t_0) \\ \vdots \\ r_{\text{rel}}(t_{\text{f}}) \end{bmatrix} \tag{4.2}$$

$$U \Lambda V^T = \text{SVD}(R_{\text{rel}}) \tag{4.3}$$

$$r_{\text{flex}} = V_{:,1} \tag{4.4}$$

Rotation vectors were used as the representation of the rotation for this part of the procedure since they encode rotations as rotations around an axis. We are interested in the axis of rotation of the elbow joint. So, with this representation the first vector of V from the Singular Value Decomposition (SVD) is the axis of strongest rotation from the dataset and therefore a good estimate of the elbow axis of rotation.

4.1.1.2. Neutral Offset. Next, to obtain the neutral offset rotation, the participant should stand upright with their arm at their side and their palm inward with their arm fully extended. They should hold this position for 5 seconds as data is collected.

The offset rotation is computed by first averaging the quaternion orientation of each IMU during the 5-second trial denoted as $\overline{Q}_{\text{upper-base}}$ and $\overline{Q}_{\text{lower-base}}$ since they are in reference to the base station. Then, the elbow offset angle θ_{offset} is computed below in Equation 4.5- 4.8.

$$\overline{q}_{\text{base-upper}} = \text{avg}(Q_{\text{base-upper}}) \tag{4.5}$$

$$\overline{q}_{\text{base-lower}} = \text{avg}(Q_{\text{base-lower}}) \tag{4.6}$$

$$q_{\text{calibrated}} = \overline{q}^{-1}_{\text{base-upper}} \cdot \overline{q}_{\text{base-lower}} \tag{4.7}$$

$$\theta_{\text{offset}} = 2 \cdot \arctan 2(q^{\text{v}}_{\text{calibrated}} \cdot r_{\text{flex}}, q^{\text{w}}_{\text{calibrated}}) \tag{4.8}$$

The final step of Equation 4.8 is a projection of the calibrated elbow angle at rest onto the axis of rotation derived previously. This gives us the angle of the arm when extended to use as an offset or reference point in during the decomposition of actual arm swing data. The argument $q^{\mathrm{v}}_{\mathrm{calibrated}}$ is the vector part of the quaternion and $q^{\mathrm{w}}_{\mathrm{calibrated}}$ is the scalar part of the quaternion.

4.1.1.3. Validation. To validate that the calibration is successful we can analyze the performance of the decomposition method during a validation trial. In this validation trial the user flexed their arm to 90°, laterally rotated out to 90°, medially rotated back to neutral, and then extended back to neutral. The results for this validation test are shown on the right in Figure 4.1.

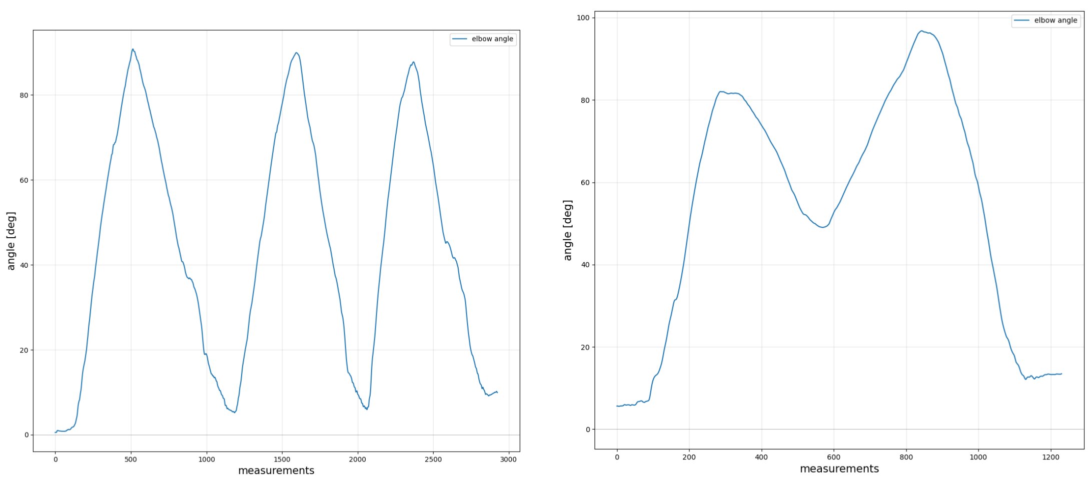


Figure 4.1. Elbow angle over time derived during IMU calibration process. On the left is the elbow angle during the calibration flexion extension sweeps. The calibration is able to cleanly extract the elbow angle during these controlled swings. On the right are results from a calibration validation. This validation does not indicate that this method for calibration is stable in more complex settings, however, results on the left indicate this method is stable enough for simple walking conditions.

The results for this validation test are poor and suggest that the calibration procedure is not stable for decomposing off axis elbow angles. This is expected since the calibration procedure extracts the axis of rotation under specific conditions. It is clear that during the calibration trial itself as shown in Figure 4.1 the procedure can extract the elbow angle well. This indicates that this sort of decomposition calibration is only valid under these conditions which match those of normal gate well enough for the purposes of this application.

4.1.2. Collection

The actual data collection procedure involves recording Xsens data during normal gait on a treadmill at controlled speeds. The speeds used were evenly distributed between $0.5\frac{m}{s}$ (a slow walking speed) and $1.5\frac{m}{s}$ (a fast walking speed).

4.1.3. Elbow Angle Decomposition

With the raw IMU acceleration and orientation data for walking trials, the next step was to decompose the elbow arm angle over time during these trials. The procedure is described below in Equations 4.9- 4.10 which follows closely with the neutral angle procedure previously described.

$$q_{\text{calibrated}}(t) = \overline{q}_{\text{base-upper}}(t)^{-1} \cdot \overline{q}_{\text{base-lower}}(t) \tag{4.9}$$

$$\theta_{\text{elbow}}(t) = 2 \arctan 2(q^{\text{v}}_{\text{calibrated}}(t) \cdot r_{\text{flex}}, q^{\text{w}}_{\text{calibrated}}(t)) - \theta_{\text{offset}} \tag{4.10}$$

The same projection of calibrated rotation onto the rotation axis is used at every time t, and the offset angle is subtracted from this angle thus giving us the elbow angle at every time t where full extension is $0°$.

4.2. Post Processing

The control is updated at a step by step basis thus requiring the command elbow angle signals to be segmented by heel-strike.

4.2.1. Cleaning and Segmentation

The IMU acceleration data can be used to identify heel strike during the trials using a peak detection algorithm. The Xsens sensors provide a clean signal which allows for easy identification of heel strike. Even so, we first filter this data with a moving average before identifying and extracting timestamps of heel-strike.

A researcher must then verify this initial heel-strike identification as seen in Figure 4.2. The goal of this step is to remove any falsely identified or unwanted heel strikes (such as heel strikes that occurred before or after the user was walking) and to add back in any missed heel-strikes. The period or time between steps should be approximately constant during these trials since they are collected at a single walking speed, therefore, the period of heel strike is a great indicator if a heel-strike has been missed or falsely identified.

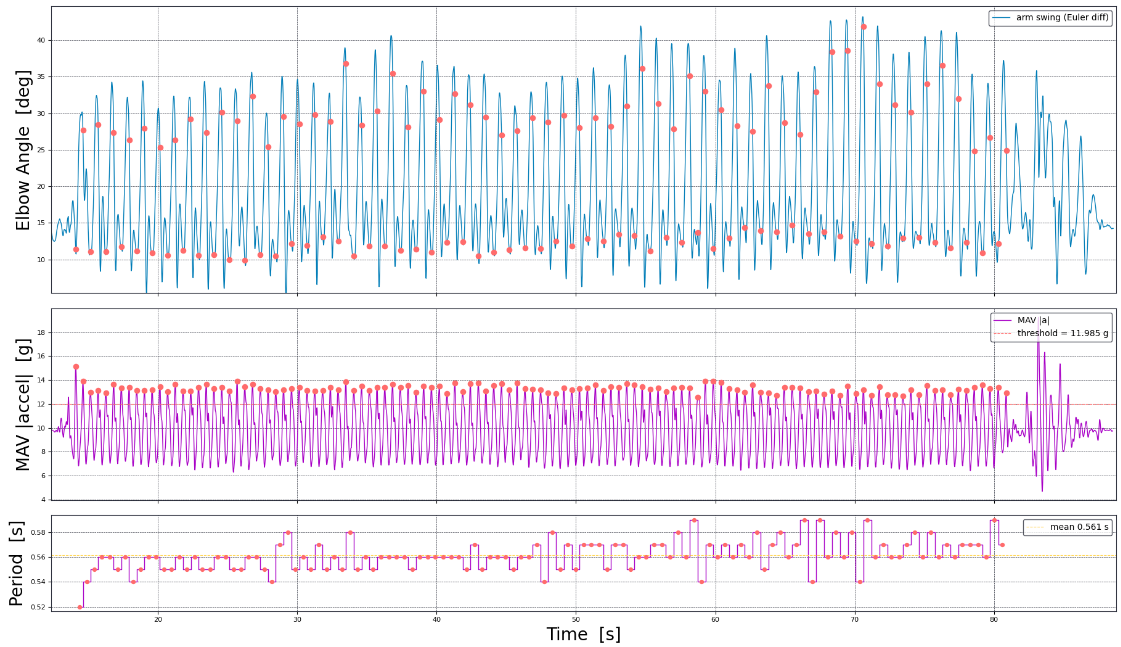


Figure 4.2. The view while cleaning walking data. The top plot shows the decomposed elbow angle over time. The middle plot shows the moving average of the acceleration during the trial, used to identify heel strike as denoted by the orange dots on both the middle and top plot. The bottom plot shows the time between consecutive steps in seconds as a measure of walking cadence. An initial pass identifies heel strikes which is then verified and cleaned by a researcher for verification.

After the heel strikes cleaned, the elbow arm swing trajectories are then segmented by heel-strike in an alternating fashion such that they are divided into forward swings and backwards swings.

4.2.2. Averaging

These forward and backward swings for each walking speed are then averaged into a single forward and backward swing pair that represents the typical arm swing at that walking speed. A visualization of this process is shown in Figure 4.3.

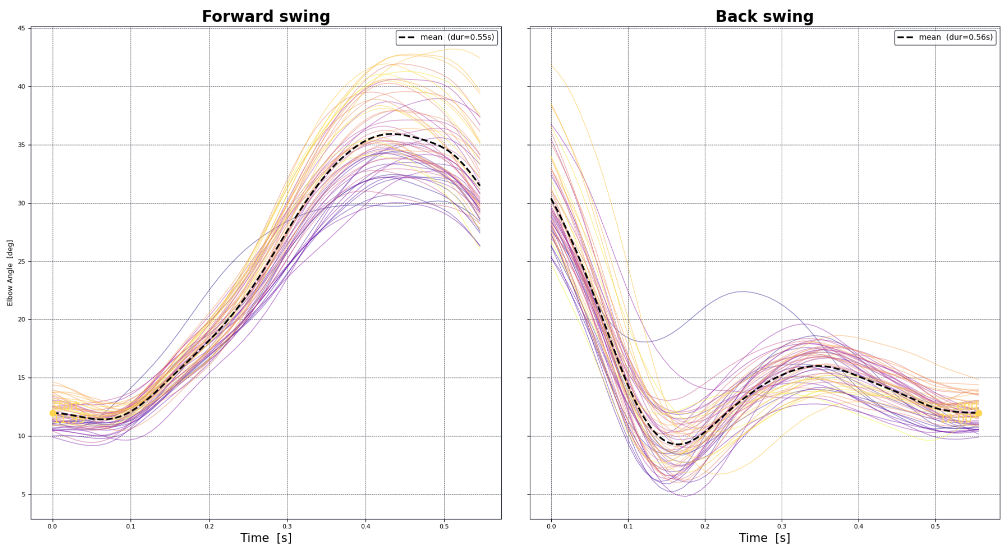


Figure 4.3. The collected forward and backward swings for a trial collected at $1.1\frac{m}{s}$ walking speed with the average of these swings shown in a dotted black line.

This procedure is repeated for each walking speed so that a set of forward and backwards arm swing pairs is compiled.

4.2.3. Building the Look-Up Table

Once there is a set of arm swing trajectories for a range of speeds from slow to fast a final look up table can be compiled. The look-up table relates the time between heel-strike to the average arm swing trajectory that was used at that speed. If we only use the trajectories from the collected data the table would be sparsely populated. We can use spline interpolation between each of the collected trajectories to fill in the look-up table. A complete look-up table is shown in Figure 4.4.

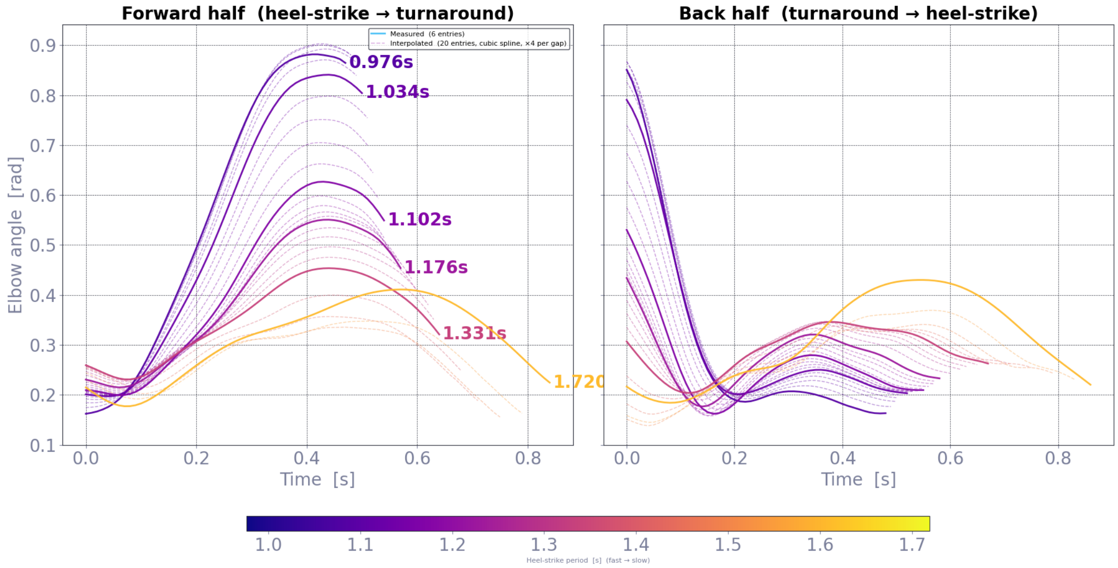


Figure 4.4. Look up table visualization relating period between heel-strike to expected elbow angle trajectory. The solid lines show the collected elbow angle trajectories while the dotted show the spline interpolated trajectories. The look-up table started with 6 entries and was populated up to 26 entries by interpolating four curves between each collected elbow angle trajectory.

CHAPTER 5

Controls Design

This section overviews all the controls that are deployed on Prototype 3. The full implementation of these methods can be found in Appendix F.

5.1. Sensing and Filtering

The first part of the controls involves taking in data from the user and filtering it to make a meaningful interpretation of the users actions that will inform the behavior of the device in later controls steps.

The sensor that is used in this section is an Intertial Measurement Unit (IMU) that collects acceleration data from the elbow joint. This raw acceleration data is noisy and thus the first filter that is applied is a moving average filter to this acceleration data. The following algorithms operate on this moving average (MAV) of the acceleration.

5.1.1. Heel Strike Detection

The first step is to identify heel-strikes from the acceleration data. Heel-strike is the moment of impact when your heel hits the ground during walking. The impact created by this event travels through your body and is sensed by the IMU onboard Prototype 3.

The heel-strike impact is not in any one particular direction relative to the IMU, so instead of examining a particular axis of acceleration (x, y, or z) we instead use the magnitude of the acceleration signal as computed below in Equation 5.1 where α is the moving average of the acceleration.

$$|\alpha| = \sqrt{{\alpha_x}^2 + {\alpha_y}^2 + {\alpha_z}^2} \tag{5.1}$$

Heel-strike is detected using a simple peak detection algorithm. For a point in time t to be considered a heel strike it must satisfy the following rules.

(1) $|\alpha(t)| > |\alpha(t-1)|$ and $|\alpha(t)| > |\alpha(t+1)|$

(2) $|\alpha(t)| > \alpha_{\mathrm{hs}}$ and $|\alpha(t)| < \alpha_{\mathrm{noise}}$

(3) $t - t_{\mathrm{prev.\ hs}} > t_{\mathrm{refractory}}$

The algorithm is dependent on the tuning of three threshold parameters: α_{hs}, α_{noise}, and $t_{\mathrm{refractory}}$. α_{hs} is the minimum $|\alpha(t)|$ to be considered a heel strike and needs to be tuned depending on the environment. A harder walking surface with harder shoes will transmit the impact of heel-strike better to the IMU, and in these cases a higher α_{hs} may be used. In other cases where the user is walking on a more compliant surface such as a treadmill, α_{hs} should be lowered.

The parameter α_{noise} is a ceiling to the allowed $|\alpha(t)|$ that can be considered a heel strike. This term was added since motor jitter and other mechanical vibrations can introduce noise into the sensor that needs to be filtered out.

The parameter $t_{\text{refractory}}$ sets the refractory period or the minimum amount of time that we would expect heel strikes to occur after one another. I set this parameter at a time period slightly longer than the maximum expected period between heel-strikes by the look-up table previously generated. To see the algorithm work in practice see Figure 5.1.

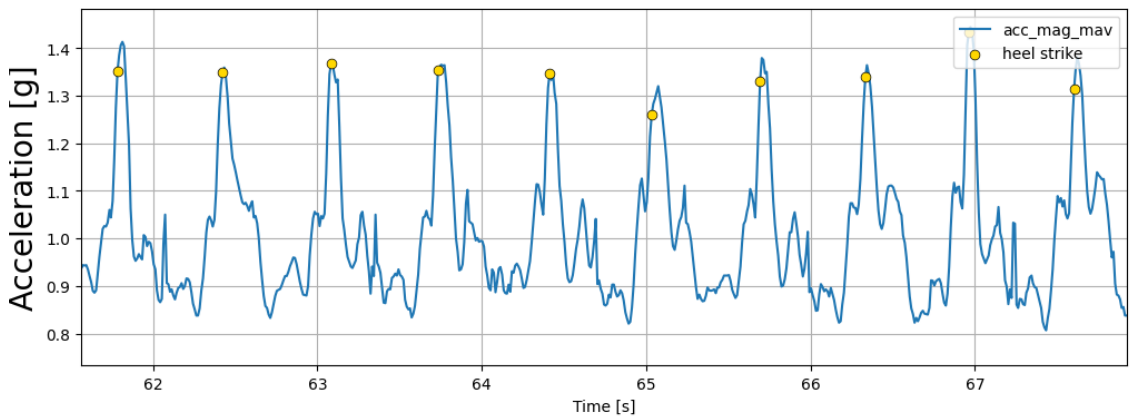


Figure 5.1. Example results from heel-strike detection algorithm. Time of detection is marked in yellow dots over the MAV acceleration signal in blue. The heel-strike algorithm is able to clearly identify the peaks which indicate the user's heel has hit the ground.

5.1.2. Heel Strike Classification

Once heel-strike has been identified, the next task is to classify the heel-strike as the left or right foot. To classify the step in such a way, we need to use the medio-lateral acceleration α_{ML} which is the left-to-right acceleration. We can decompose this acceleration from the IMU signal using the simple trigonometric conversion described in Equation 5.2 where θ_{IMU} is the orientation of the IMU on the PCB (65°).

$$\alpha_{\mathrm{ML}} = \alpha_x * \sin(\theta_{\mathrm{IMU}}) + \alpha_y * \cos(\theta_{\mathrm{IMU}}) \tag{5.2}$$

Although the medio-lateral acceleration is periodic in shape, its peaks do not correspond to the time of heel-strike. Therefore, classifying based on the $\alpha_{ML}(t_{\mathrm{hs}})$ is not possible. So we need to extract a useful feature from this signal, as can be done using the procedure described in Equation 5.3- 5.4.

$$x_{\mathrm{hs}} = \int_{t_{\text{prev. hs}}}^{t_{\mathrm{hs}}} \alpha_{\mathrm{ML}}(t)dt \tag{5.3}$$

$$f_{\mathrm{hs}} = \frac{x_{\mathrm{hs}} - x_{\text{prev. hs}}}{t_{\mathrm{hs}} - t_{\text{prev. hs}}} \tag{5.4}$$

With this feature f_{hs} in hand we can apply the following rules to classify the heel strike.

(1) If $f_{\mathrm{hs}} > f_{\mathrm{thresh}}$ then the heel-strike is LEFT
(2) If $f_{\mathrm{hs}} < -f_{\mathrm{thresh}}$ then the heel-strike is RIGHT
(3) Otherwise the heel-strike is UNKNOWN

This algorithm is a sign based classification with a small threshold built in that ensures certainty about the classification. The parameter f_{thresh} should be tuned such that classifications with insufficient confidence are discarded and set as UNKNOWN.

The sign convention that is used here is dependent upon the orientation of the IMU. In the case of Prototype 3 which was worn on the left arm during testing, a

$\alpha_{ML}(t_{\text{hs}})$ with a positive sign indicated leftward acceleration. Therefore, for systems where the rightward acceleration is positive would need the opposite convention that which is used here.

To see the algorithm in action, Figure 5.2 shows the $\alpha_{ML}(t_{\text{hs}})$ as well as the resulting f_{hs} and the final heel-strike classification.

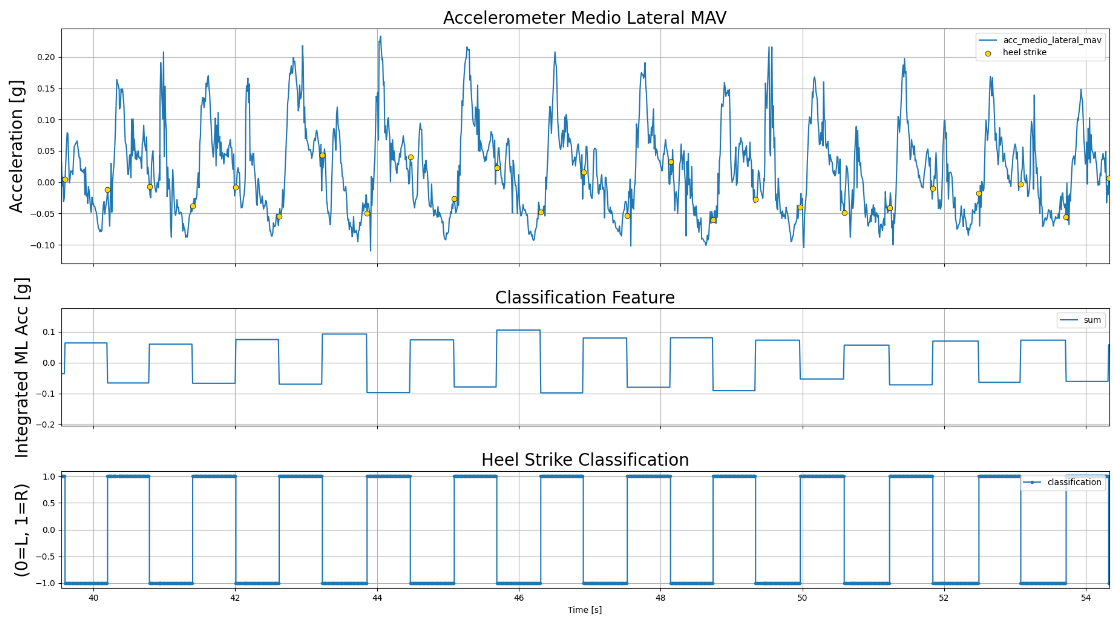


Figure 5.2. Results above show the heel-strike classification algorithm working in real time. The top plot shows the medio-lateral acceleration. The middle plot shows the resulting classification feature. The bottom plot shows the final heel-strike classification where a 1 indicates a RIGHT classification and 0 indicates a LEFT classification.

5.1.3. Confidence Update

The last element of sensing and filtering is the confidence update step. This step keeps track of the expected heel-strike classification and updates a confidence metric based on the results from the heel-strike classification algorithm.

First, the confidence update algorithm initializes a guess of the expected heel-strike classification as LEFT (this initialization is arbitrary). Then, at every new heel-strike classification the confidence update algorithm runs following these rules described below.

(1) If classification matches expectation, increment confidence and flip expectation
(2) If classification is UNKNOWN, do not change confidence
(3) If classification is the opposite of expectation, decrement confidence
(4) If classification doesn't match expectation, and confidence is greater than zero, then flip expectation
(5) If the user stops walking, set confidence to 0
(6) Clamp confidence between 0 and the maximum confidence

This algorithm updates both the confidence metric and also the expectation during each run. The algorithm does not switch the expectation until the confidence has reached zero, since that is the only case in which the expectation is not flipped. The only parameter that can be tuned in this algorithm is the maximum confidence which I have set to 10. It should not be too high since if the confidence metric is allowed to integrate to a huge number it will take many opposite heel strikes to correct the expectation. To see this algorithm working see Figure 5.3

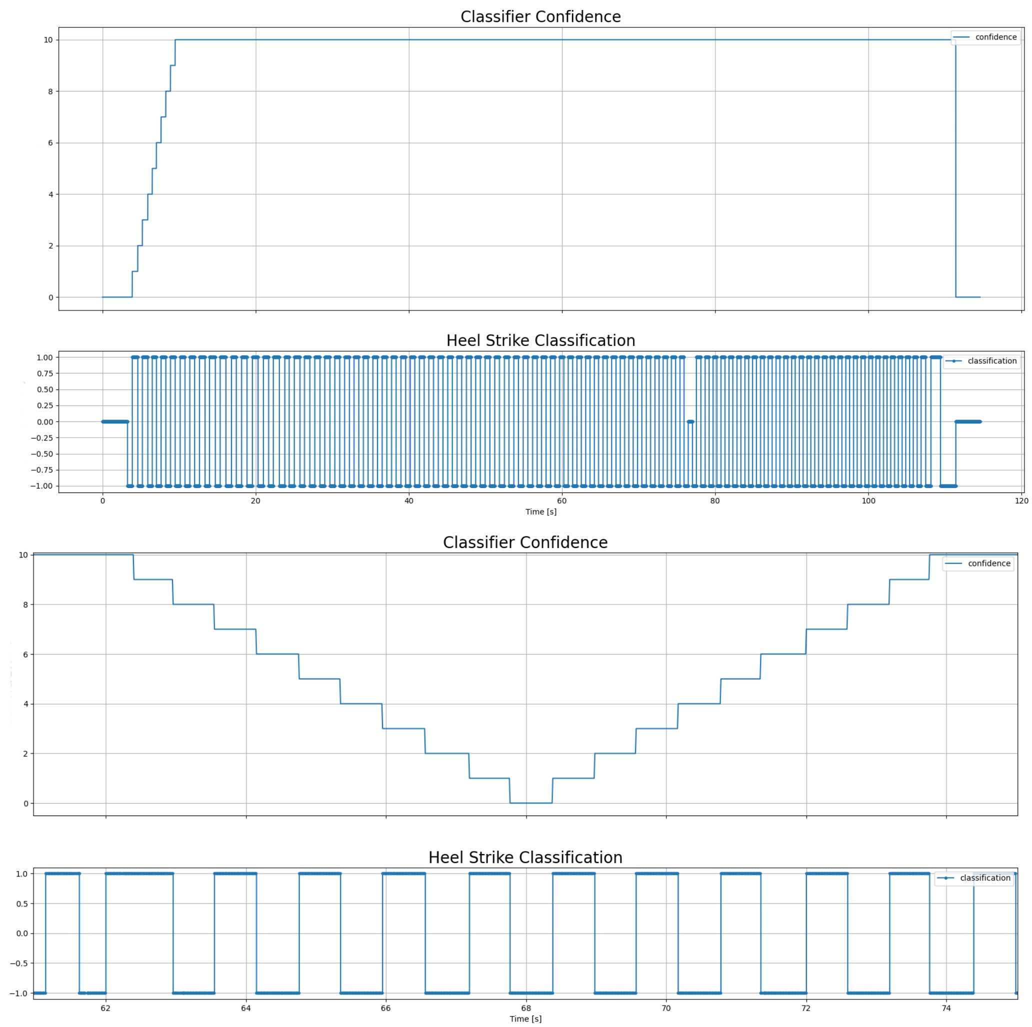


Figure 5.3. The confidence update can be seen running in two difference scenarios above. In the top plot the expectation aligns with classification and the confidence rises to its maximum and stays there during the entire trial. In the plot on the bottom there is a misidentified heel strike and the confidence decreases until reaching zero where its expectation changes allowing the expectation to rise up again.

5.2. Mid-Level Control

With sensing and filtering complete, the control system can now process these incoming data to control the state of the prosthetic device itself.

5.2.1. Prosthesis State Machine

The prosthesis state machine manages both the state of the position controller (on or off) and the trajectory selection. The controller state is determined by a simple algorithm with the following rules.

(1) If the confidence > 5, enable control

(2) If control is enabled and confidence reaches 0, stop control

This state machine algorithm ensures that there is confidence in the walking pattern of the user before starting control and requires the confidence metric to lose all confidence in the given expectation before stopping control.

Next, once control has been enabled the prosthesis state machine must select a trajectory. A new trajectory is selected at each heel strike based on the sum of the period between heel-strikes for the most recent two steps. This is in essence the time of one complete gait cycle. The arm swing trajectory that has a gait cycle period closest to the calculated gait cycle period is selected. When a new trajectory is selected, the controller starts using this trajectory immediately from the start of the forward or backward swing (whichever is next given the internal expectation). This constant update of the command trajectory is the mechanism which maintains the temporal alignment of the command signal with the user's actual walking cadence. The performance of this prosthesis state machine can be seen in Figure 5.4.

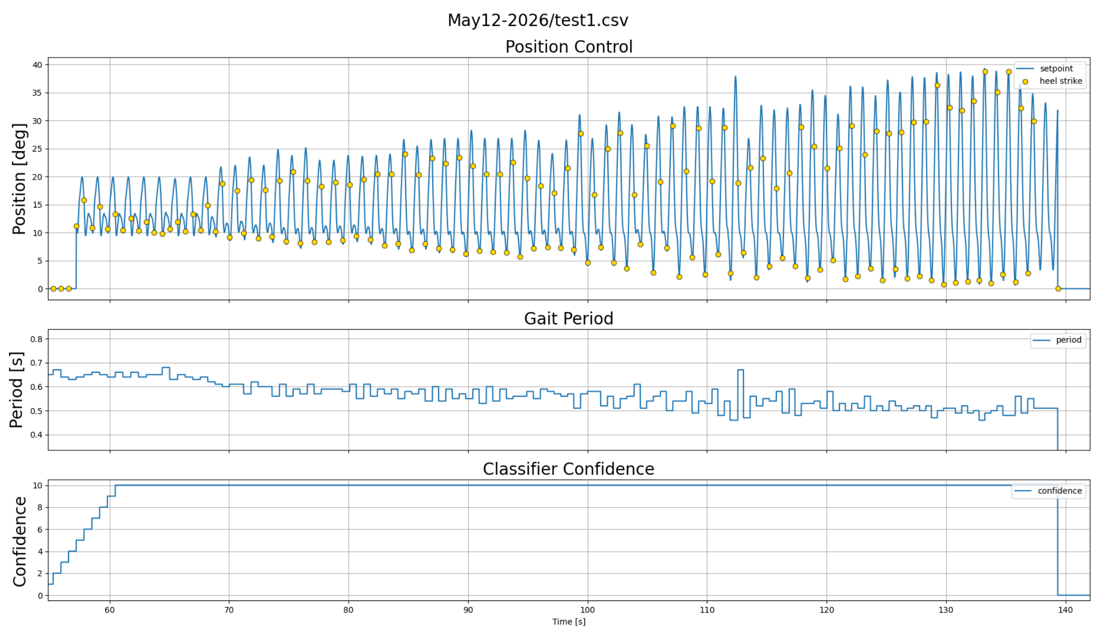


Figure 5.4. The results from the prosthesis state machine show how the controller enabler and trajectory selection work in real time during operation. The controller enabler starts control once confidence has reached five, as the command position starts oscillating at that point in time. Also, as the user increases their walking speed, and therefore decreases the time between steps as seen in the middle plot showing period, the arm swing trajectory increases in frequency and amplitude as it should for faster walking speeds.

5.2.2. Arm Swing Trajectory Transitioning

Arm swing command trajectories are updated constantly in this formulation. If this were implemented directly without any filtering, there would be discontinuities in the command signal. Therefore, some sort of transitioning is required between command signals.

In this case we use an exponential moving average (EMA) filter that transitions from the previous trajectory to the newly selected trajectory. The formulation for this EMA filter is shown in Equation 5.5 where β is the transition parameter.

$$\theta_{\text{command}}(t) = (1-\beta)\cdot\theta_{\text{prev}}(t) + (\beta)\cdot\theta_{\text{new}}(t) \tag{5.5}$$

The transition is executed based on the update of the transition parameter β. The value of β indicates the level of weighting towards the new trajectory. A β of 1 indicates full weighting of the new trajectory. In this formulation β is initialized to 0.1 when a new trajectory is selected. Then at each timestep throughout the trajectory β is updated using Equation 5.6 until it reaches its maximum of 1.

$$\beta = \beta + m\cdot\frac{0.9}{len(\theta_{\text{new}})} \tag{5.6}$$

The numerator of this β update is set as 0.9 because that is the remaining value for β to rise from its starting value of 0.1 to 1.0. The scalar multiplier m of this fraction indicates how much faster β will reach saturation than the total period of the trajectory. For example, if m is set as 2, β will saturate halfway through the trajectory. This value m can be tuned to saturate β faster or slower, we used 2 in our implementation. The result of the trajectory transitioning algorithm is shown in Figure 5.5.

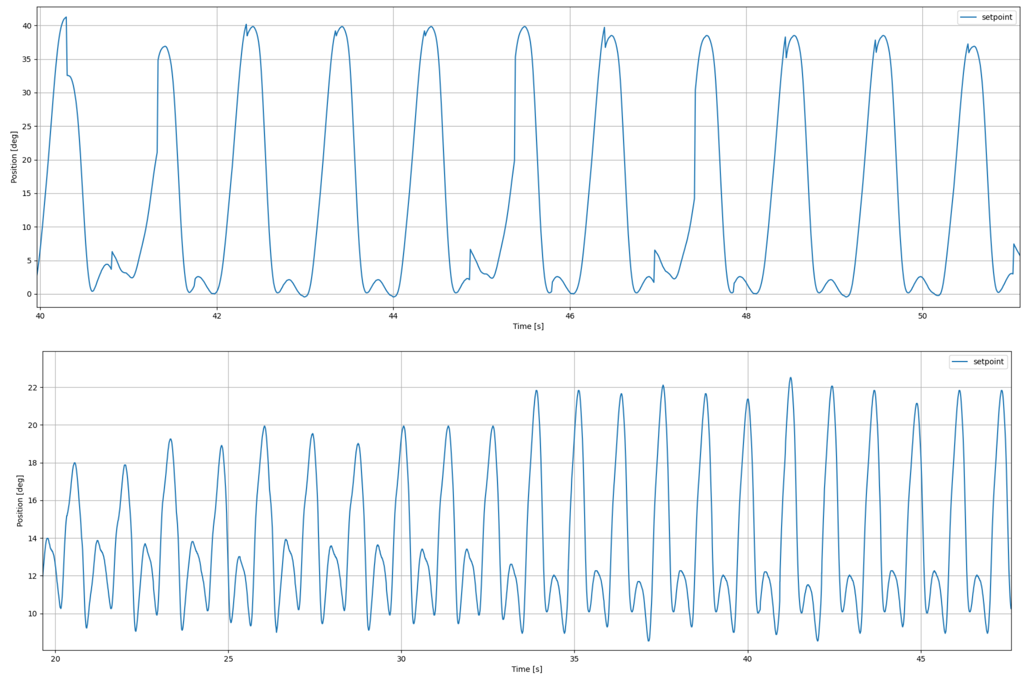


Figure 5.5. The plots above show trajectory update without (top) and with (bottom) trajectory transitioning. The top plot has clear discontinuities at the break points between different trajectories while the bottom plot smoothly transitions from a slower walking speed elbow angle trajectory to a faster walking speed one.

5.3. Motor Control

The final component of the control system is the low level motor control. We used a double nested control loop with an outer loop that generates torque commands for an inner loop.

5.3.1. Position Trajectory Tracking

The outer position control loop uses PD control to track the trajectories of elbow angle over time. We experimented with a variety of more complex control strategies including the use of gravity compensation feed forward terms, however, nothing more than PD was required to get good control results.

The position control loop updated at a rate of $1000Hz$ while the command that it was given was updated at a rate of $100Hz$.

5.3.2. Brushless Motor Control

For low level torque control on the Cubemars GL40 we used the NUControl codebase [**17**]. NUControl is an implementation of FOC control which attempts to align the d and q axes of the motor such that all current sent to the BLDC motor is used to generate torque. The NUControl implementation worked well for this device since it allowed for the loading of previous calibrations for the motor. It also provided good control of the motor only using a feed forward model generated by the motors specifications. Therefore, NUControl took in the torque command generated by the PD position controller, transformed it, and sent the command to the BLDC motor.

5.4. Control Block Diagram

All of these control system elements come together to control Prototype 3 to adaptively respond to user walking cadence and output arm swing trajectories. The complete control block diagram is shown in Figure 5.6.

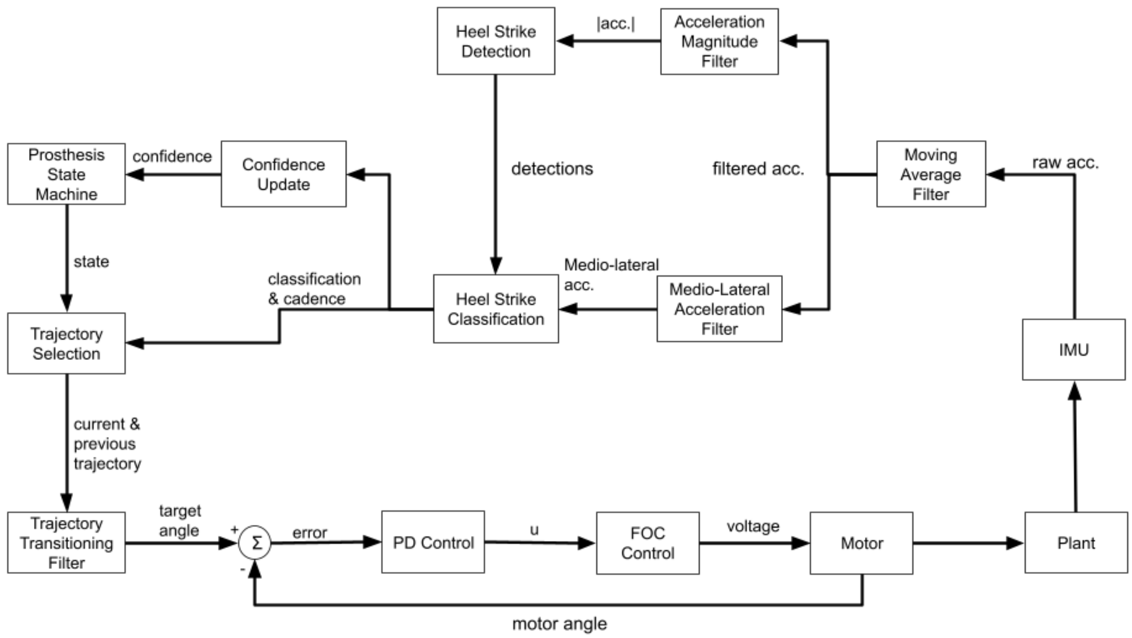


Figure 5.6. This diagram demonstrates how the control loop is closed, from IMU sensor data passed through filtering into the mid-level control, and finally back to the motor control system which sends torque into the plant. The plant in this case represents all the dynamics of the user from their heel-strike impact to how their natural arm swing interacts with the prosthetic arm.

CHAPTER 6

Results and Conclusions

6.1. Results

The performance of the arm was tested under two varied conditions. The first condition was with or without the end-effector attached to the prosthesis. The second case controlled for is with or without the spring. The code used to collect data and create these visualizations can be found in Appendix F.

6.1.1. Result Visualizations

6.1.1.1. Without Control. The result in Figure 6.1 shows the behavior of the prosthetic arm if no control is applied and the arm swings freely at the users side during normal gait.

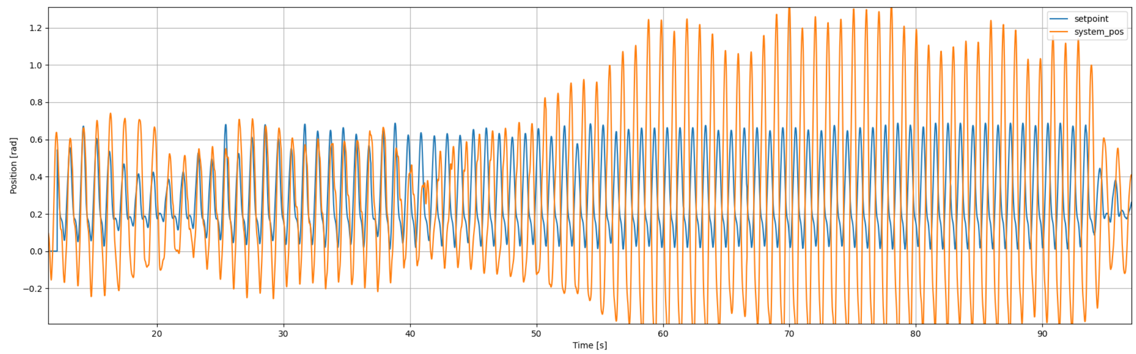


Figure 6.1. Arm swing behavior without control applied shows chaotic prosthetic elbow swings in orange that are completely out of phase and amplitude with the expected arm swing trajectory that is shown in blue. The end-effector and spring were attached for this trial.

6.1.1.2. Without End-Effector, With Spring. See the results in Figure 6.2.

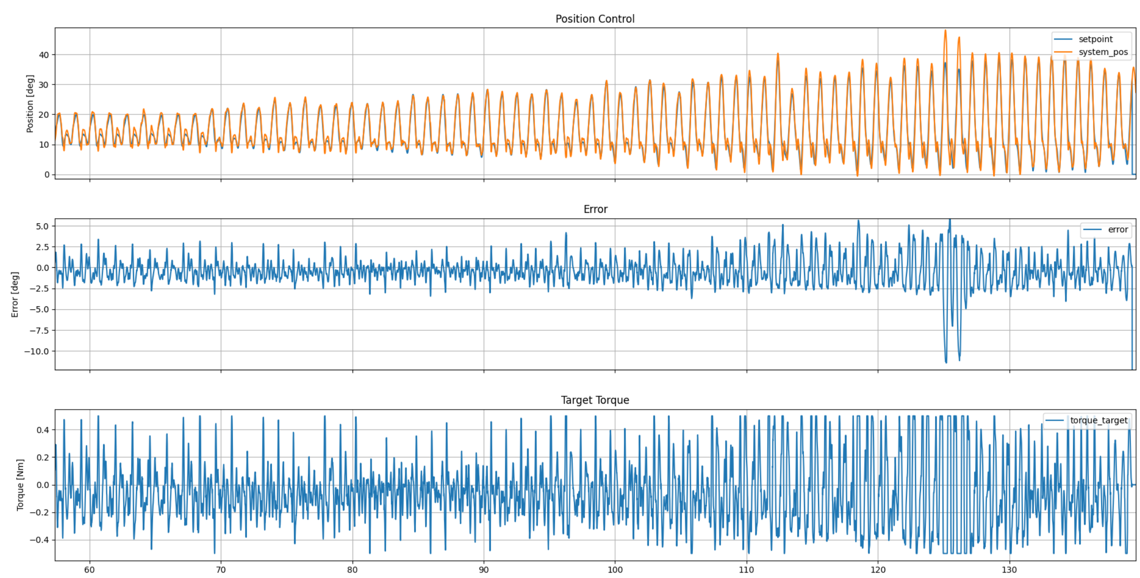


Figure 6.2. Results without end-effector attached and with a spring show good tracking of the command signal with minimal error throughout the trial and an unsaturated torque command signal.

6.1.1.3. Without End-Effector, Without Spring. See the results in Figure 6.3.

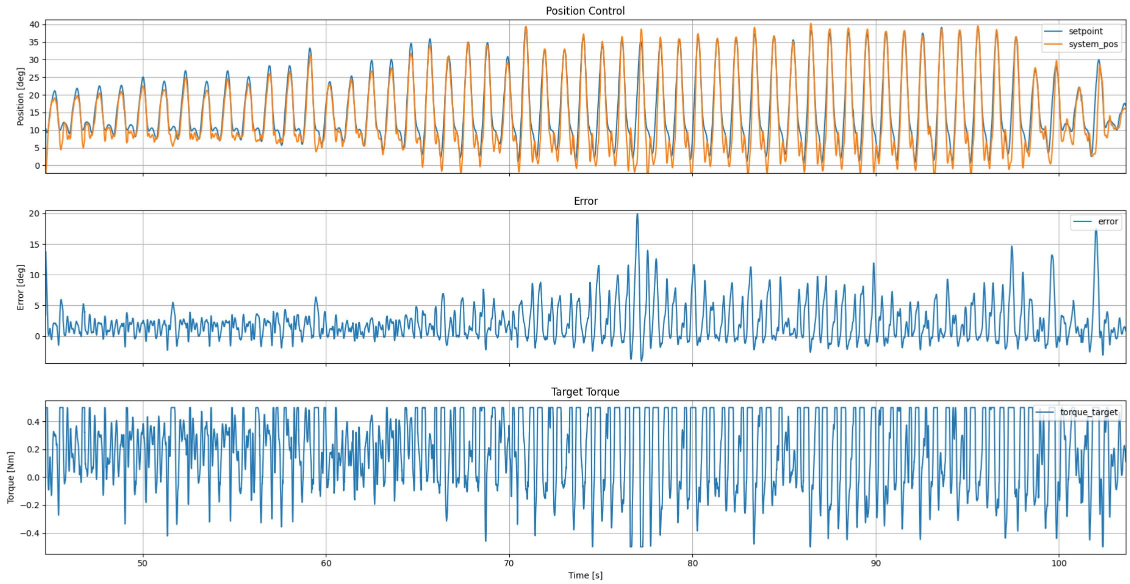


Figure 6.3. The results without the end-effector attached at without the spring show undershooting the command signal at slower walking speeds and overshooting on the backswing during the faster walking speeds. The torque profile is noticeably biased in one direction and error is elevated throughout the trial.

6.1.1.4. With End-Effector, With Spring. See the results in Figure 6.4.

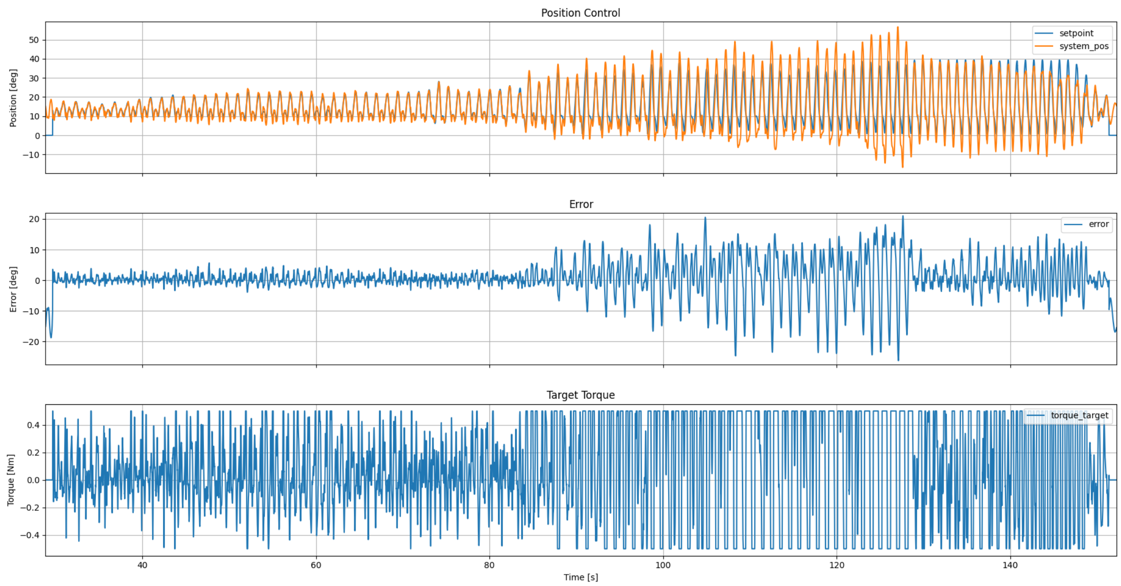


Figure 6.4. The results with the end-effector and spring attached show good control during slow walking speeds. However, as the walking speed increases the error increases the torque command signal is saturated.

6.1.1.5. With End-Effector, Without Spring. See the results in Figure 6.5.

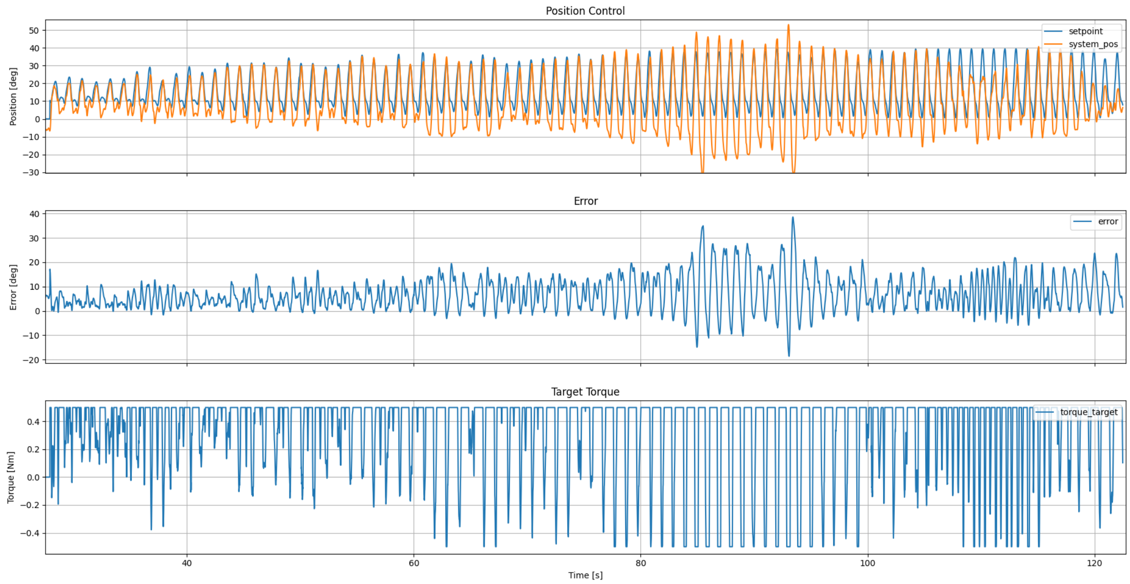


Figure 6.5. The results with the end-effector and without the spring attached show moderate to high levels of error throughout the trial. The torque command signal is saturated throughout the entire trial.

6.1.2. Result Statistics

To summarize the results above we can observe the distribution of key signals across trials. Table 6.1 shows these results collected from the trials above in addition to other trials under the same conditions.

Table 6.1. Summary of tracking error and motor torque statistics across all test conditions.

	Tracking Error (°)		Motor Torque (Nm)			
Condition	$\overline{\lvert\epsilon\rvert}$	$\sigma_{\lvert\epsilon\rvert}$	$\overline{\tau}$	σ_τ	$\overline{\lvert\tau\rvert}$	$\sigma_{\lvert\tau\rvert}$
No EE, Spring	1.551	1.4499	−0.045	0.264	0.220	0.152
No EE, No Spring	2.620	2.632	0.2299	0.234	0.277	0.173
EE, Spring	4.071	5.2993	0.0499	0.352	0.305	0.183
EE, No Spring	7.0994	5.9945	0.338	0.274	0.408	0.151

The mean magnitude of error $\overline{|\epsilon|}$ and mean magnitude of torque $\overline{|\tau|}$ increase as the conditions become increasingly more difficult to control (from top to the bottom of the table). The average torque $\overline{\tau}$ is nearly zero mean when the spring is used, and when it is not used the $\overline{\tau}$ is shifted dramatically. The standard deviation of the magnitude of the error $\sigma_{|\epsilon|}$ also increases as the conditions become more difficult to control, although the standard deviation for the torque σ_{τ} and magnitude of the torque $\sigma_{|\tau|}$ are generally consistent throughout trials.

6.2. Discussion

6.2.1. Analysis of Prototype Performance

The results show good control of the prosthetic arm under the most favorable conditions (without the end-effector, with the spring). The device is able to output the desired trajectory at all walking speeds with minimal error and without saturating the controller. This indicates that the system is fully controllable under these conditions for the motor.

When the end-effector is attached to the prosthesis the results remain good at slow walking speeds. However, at faster walking speeds Prototype 3 is not able to control the swing of the prosthetic. This is clear based on the saturation of the torque command signal.

6.2.2. Analysis of Spring

The theory of integrating a spring into the device design to shift the mean of the torque profile to zero is supported by the findings as shown in Table 6.1. It is clear that the trials which used the spring had means that were near zero, especially compared to the trials that did not use the spring.

The trials that used the spring not only had torque profiles of near zero mean, they also had lower averages for the magnitude of torque and magnitude of error indicating better control and lower energy required to control the prosthesis.

These results are critical to the success of the actuated prosthetic arm project since they both lower the amount of torque required and reduce the amount of energy required. Reducing the magnitude of torque required will allow for comparatively smaller actuators to be used, and reducing the amount of energy consumed will allow for the prosthesis to last for longer durations of use.

One qualification to these findings is that the same PD gains were used for the position controller between all these trials. It is possible that the device could have been tuned to perform better on trials that did not use the spring. However, it still remains that the magnitude of torque required is reduced and the mean of the torque profile is shifted to zero under this condition.

6.2.3. Comparison with Prototype 1

These results conflict with the results from Prototype 1 in several ways. First, the prosthesis used in Prototype 3 is far lighter than that used in Prototype 1. Therefore, we would expect that the required torque to track an arm swing trajectory would be lower than the $0.25Nm$ amplitude that was observed from the results of Prototype 1. Furthermore, the shape of the output torque that is exhibited from Prototype 3 does not match the smooth periodic torque signal that Prototype 1 exhibited. The results shown above show a far more chaotic torque command signal.

These conflicts may arise from a series of sources. For example, it is possible that the conditions under which Prototype 1 and 3 were tested were different. Or it is even possible that the custom FOC driver board used for Prototype 3 is not behaving as expected.

Prototype 3 is not yet well characterized enough to overwrite findings from Prototype 1 and requires more testing and analysis to be sure of these findings.

6.2.4. Confounding Factors

During user testing it became clear that the participant behavior during testing can have a huge effect on the results of the device tracking the trajectory. I decided to constrain my upper arm from moving during testing since allowing my arm to swing resulted in far worse results from the prototype. Allowing my arm to swing likely introduced difficult to account for dynamics into the control system. Furthermore, I theorize that allowing my arm to swing freely removed the reaction force produced by my constrained upper arm that helped the device move the prosthesis.

It is both unclear how a user with upper limb loss may behave while wearing the device and whether pursuing this goal of tracking an arm swing trajectory will produce the cancellation of whole body angular momentum that is desired.

6.3. Future Work

6.3.1. Control Method

The decision to pursue elbow angle trajectory tracking was made in part because the torque requirements were lower than those anticipated by the double pendulum dynamic simulation, as shown by the results of Prototype 1.

The question of if this method of control will achieve the desired result of whole body angular momentum regulation is still not answered. It is difficult to assert that controlling for one variable (the elbow position) will properly control another (momentum).

Future work should investigate better methods to integrate the goal of restoring whole-body angular momentum regulation into the control system used on the prosthetic arm. For example, we could scale the position trajectories by the ratio between the angular inertia of the prosthetic arm and the users intact arm. Equations 6.1- 6.3 below derive the relationship between angular momentum and elbow angle.

$$I_{\text{limb}} \cdot \omega_{\text{limb}} = I_{\text{pros}} \cdot \omega_{\text{pros}} \tag{6.1}$$

$$\int \frac{I_{\text{limb}}}{I_{\text{pros}}} \cdot \omega_{\text{limb}} dt = \int \omega_{\text{pros}} dt \tag{6.2}$$

$$\frac{I_{\text{limb}}}{I_{\text{pros}}} \cdot \theta_{\text{limb}} = \theta_{\text{pros}} \tag{6.3}$$

Other interesting pathways may include tracking velocity trajectories or performing more advanced modeling to understand the effects of the different methods of control.

6.3.2. Analysis of Required Torque

In the case that position control is used in future work, a deeper analysis is needed on the torque requirements at the elbow joint for position control. Results from Prototype 1 indicated that only $0.25Nm$ of torque is required for good control, however, Prototype 3 seems to contradict this finding.

Taking time to evaluate this question either experimentally or through simulation would be worthwhile. One approach would be to build a basic prosthetic arm, similar to that of Prototype 1, that can have a hot swappable motor at the elbow joint actuated by a well known motor driver. With a simulated prosthesis of known weight, the control system used in Prototype 3 could be applied to evaluate the torque needed to track the trajectory. Another approach would be to re-evaluate the double pendulum simulation.

6.4. Conclusions

The mechanical and electrical system design of Prototype 3 are solid improvements on previous iterations of the design. Furthermore, the control system integrates the user into the operation of the device more ever before. Together, these three elements of Prototype 3 come together to produce great elbow trajectory tracking results with a reduced mass on the prosthesis. Furthermore, testing with an external spring confirmed the theory that the torque profile could be shifted to be zero mean thus reducing the torque requirement and reducing the energy required to actuate the prosthesis.

Testing with the end-effector prosthesis revealed that the device may be underpowered despite findings from Prototype 1. More investigation is required into the discrepancies between the results of Prototypes 1 and 3.

References

[1] H. Herr and M. Popovic, "Angular momentum in human walking," *Journal of Experimental Biology*, vol. 211, pp. 467–81, Feb 2008.

[2] M. J. Major, S. M. McConn, J. L. Zavaleta, R. Stine, and S. A. Gard, "Effects of upper limb loss and prosthesis use on proactive mechanisms of locomotor stability," *Journal of Electromyography and Kinesiology*, vol. 48, pp. 145–51, Oct 2019.

[3] J. A. Dunn, N. G. Gomez, B. Wong, S. K. Sinclair, H. B. Henninger, K. B. Foreman, and K. N. Bachus, "Transhumeral prosthesis use affects upper body kinematics and kinetics," *Gait & Posture*, vol. 112, pp. 59–66, Jul 2024.

[4] S. Topuz, E. Kirdi, A. I. Yalcin, O. Ulger, H. Keklicek, and G. Sener, "Effects of arm swing on spatiotemporal characteristics of gait in unilateral transhumeral amputees," *Gait & Posture*, vol. 68, pp. 95–100, Feb 2019.

[5] J. A. Kent and M. J. Major, "Asymmetry of mass and motion affects the regulation of whole-body angular momentum in individuals with upper limb absence," *Clinical biomechanics*, vol. 76, p. 105015, Jun 2020.

[6] J. A. Dunn, N. G. Gomez, B. Wong, S. K. Sinclair, K. B. Foreman, K. N. Bachus, and H. B. Henninger, "Transhumeral prosthesis use and disuse affects whole-body angular momentum," *Clinical Biomechanics*, vol. 120, p. 106365, Dec 2024.

[7] M. Major, "Fall prevalence and contributors to the likelihood of falling in persons with upper limb loss," *Physical therapy*, vol. 99, pp. 377–87, Apr 2019.

[8] M. Major, R. Knox, Y. Ben Shalom, and M. Elwin, "Method and system to activate limb movement in amputees," U.S. Patent 2021-035-02 17/705,665, May 3, 2021.

[9] M. Jenz, A. Garcia-Casal, Y. B. Shalom, M. L. Elwin, and M. J. Major, "Design and control of an actuated prosthetic elbow to restore arm swing for persons with upper limb absence," *2025 International Conference On Rehabilitation Robotics (ICORR)*, pp. 730–735, 05 2025. [Online]. Available: https://ieeexplore.ieee.org/document/11063135

[10] "G60 kv25/kv55 gimbal motor," CubeMars, 2026. [Online]. Available: https://www.cubemars.com/product/g60-kv25-kv55-gimbal-motor.html

[11] Y. Ben-Shalom, "Motorized-prosthetic-arm results," GitHub, 2025. [Online]. Available: https://github.com/YaelBenShalom/Motorized-Prosthetic-Arm/tree/master/motor_control/graphs/treadmill_test

[12] L. C. Smail, C. Neal, C. Wilkins, and T. L. Packham, "Comfort and function remain key factors in upper limb prosthetic abandonment: findings of a scoping review," *Disability and Rehabilitation: Assistive Technology*, vol. 16, pp. 1–10, 03 2020.

[13] "Gl40 kv70 gimbal motor," CubeMars, 2026. [Online]. Available: https://www.cubemars.com/product/gl40-kv70-gimbal-motor.html

[14] "E-400 elbow," Fillauer — Innovative Orthotic and Prosthetic Products, 05 2025. [Online]. Available: https://fillauer.com/product/e-400-elbow/#product-specifications

[15] A. Skuric, "Arduino simple field oriented control project," Simple FOC, 2024. [Online]. Available: https://www.simplefoc.com/simplefoc_shield_product_v2

[16] Z. Hoegberg, S. Donahue, and M. J. Major, "An open-source wearable system for real-time human biomechanical analysis," *Sensors*, vol. 25, p. 2931, 05 2025.

[17] L. Batteas, "Github - nu-aiet/nucontrol," GitHub, 2025. [Online]. Available: https://github.com/NU-AIET/NUControl

APPENDIX A

ICORR 2025 Submission

This appendix contains the conference paper submitted at the 2025 ICORR conference in Chicago, Illinois.

Design and Control of an Actuated Prosthetic Elbow to Restore Arm Swing for Persons with Upper Limb Absence

Michael Jenz
Dept. of Mechanical Engineering
Northwestern University
Evanston, United States
michaeljenz2026@u.northwestern.edu

Allan Garcia-Casal
Dept. of Mechanical Engineering
Northwestern University
Evanston, United States
AllanGarcia-casal2023@u.northwestern.edu

Yael Ben Shalom
Dept. of Mechanical Engineering
Northwestern University
Evanston, United States
yael.bshalom@gmail.com

Matthew L. Elwin
Dept. of Mechanical Engineering
Northwestern University
Evanston, United States
elwin@northwestern.edu

Matthew J. Major
Dept. of Physical Medicine and Rehabilitation
Northwestern University
Chicago, United States
matthew-major@northwestern.edu

***Abstract*—Dynamic balance in human locomotion relies on the regulation of whole-body angular momentum. Asymmetry in upper extremity body mass and motion that occurs from upper limb absence disrupts this regulation. Conventional transhumeral prostheses do not restore controlled arm swing, leaving this asymmetry unresolved. This project aimed to develop an actuated prosthetic elbow that mimics natural elbow rotations to restore arm swing during walking. Two prototypes with contrasting designs were fabricated and tested. Prototype 1, featuring a direct drive motor, tracked arm swing trajectories based on able-bodied data with less than 5% position error. It enabled evaluation of predicted elbow torque requirements, showing that a double-pendulum dynamic model overestimates torque needs. This finding supports using smaller, lighter direct drive motors for natural arm swing. While Prototype 1 performed well, its motor was too large for practical integration. Prototype 2 successfully fit the drive and control system into a standard prosthetic envelope for enhanced applicability. Testing revealed that gearbox friction hindered control, and successful operation required synchronizing elbow movement with natural arm swing frequency. Despite using the same control system, both prototypes produced notably different outcomes. These results highlight potential for future designs incorporating a small direct drive motor to match Prototype 1's performance while maintaining Prototype 2's compact form.**

***Index Terms*—Robotic prostheses, Amputation, Prosthesis, Upper extremity, Mechatronics, Gait, Arm swing, Balance**

I. Introduction

Reciprocal arm motion is essential for balancing leg momenta to minimize whole-body angular momentum that facilitates locomotor balance in bipedal human gait [1]. Recent evidence has revealed that individuals with unilateral upper limb absence experience bilateral asymmetry in both upper extremity body mass and motion [2]–[4], which affects the tight regulation of whole-body angular momentum during walking [5], [6]. This asymmetry and impact on dynamic balance may contribute to the high prevalence of falls in persons with upper limb absence, in which 67% of reported falls occur during walking [7]. While persons with unilateral transhumeral-level limb absence exhibit a small amount of affected limb arm swing during walking (nearly a third of the intact limb) [2]–[4] that gives rise to this asymmetry, conventional transhumeral prostheses are not designed to restore controlled arm swing and facilitate the inter-limb segment momenta cancellation required to minimize whole-body angular momentum [3], [6]. In fact, the limited arm swing of the affected side during walking remains the same whether or not a conventional arm prosthesis is being worn [2], [3].

Therefore, interventions are needed to restore the biomechanical symmetry of angular momentum and aid dynamic balance in persons with unilateral upper limb absence. This project aimed to develop a mechatronic prosthetic elbow that generates stereotypical elbow rotations to restore affected limb arm swing of persons with transhumeral-level limb absence during walking [8]. This paper describes steps for the design, development, and preliminary testing of two mechatronic systems that showcase contrasting design methods.

II. Methods

Two mechatronic designs of a robotic elbow were implemented to accommodate different motors for actuation. Prototype 1 is powered by a larger, brushless, direct-drive motor. Prototype 2 is powered by a small brushed motor with a high gear ratio, extracted from a servo motor, developed in response to the high cost and complexity required to use a brushless motor. Prototype 2 aimed to replicate Prototype 1 function with reduced complexity. Since Prototype 2 was developed after Prototype 1, it features a more compact design and prosthetic arm attached to the elbow, rather than a mock prosthesis that is weighted to mimic anatomical inertia (as seen in Prototype 1). These prototypes were developed

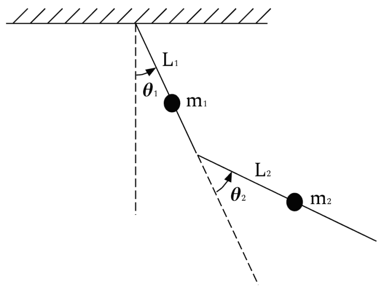


Fig. 1. Model representation of arm-body dynamics during walking as a double pendulum. System configuration is defined by shoulder angle θ_1 and elbow angle θ_2. The model predicts the required torque output at the elbow joint to generate desired swing trajectories.

only to assess the control and actuation platforms. This section contrasts the prototypes designs.

A. Prototype 1

1) Arm-Body Dynamics Simulation: The arm-body dynamics for arm swing during walking can be modeled as a double pendulum system as seen in Fig. 1. The system can be fully defined using the configuration $q = [\theta_1, \theta_2]$, the angles of shoulder and elbow flexion.

In this model, the arm is treated as a separate entity from the body, with energy input at the shoulder joint to simulate body-generated movement. The model calculates the torque required at the elbow joint to generate desired arm swing trajectories.

The Euler-Lagrange equations relate the system configuration q and its first and second derivative to the external forces acting on the system at a given time. This relationship can be used to forward integrate the behavior of the system and simulate its dynamics.. In this case, user data with trajectories (that is position, velocity, and accelerations) of the elbow are substituted into the Euler-Lagrange equations for a double pendulum to derive the required torque:

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = \tau \tag{1}$$

Here, τ is the joint torque vector and L is the Lagrangian defined as the difference between the kinetic and potential energy of the system.

2) Motor Selection: The power specifications for the motor can be estimated by the results from the Euler-Lagrange back-calculated torque. The motor specifications will be defined by the maximum output from the model. The arm swing trajectory that produces these maximum values is the most energetic arm swing. After examination of arm swing trajectory data collected from an able-bodied person, it was determined that the maximum speed at which data was collected, $1.4\frac{m}{s}$, was when the person produced the most energetic arm swing trajectory. Therefore, the largest torque output required from a motor was calculated from these configuration data. Fig. 2 displays the torque vs angular velocity profile, where the torque is the output from the Euler-Lagrange equations and the velocity is the time derivative of the angular position.

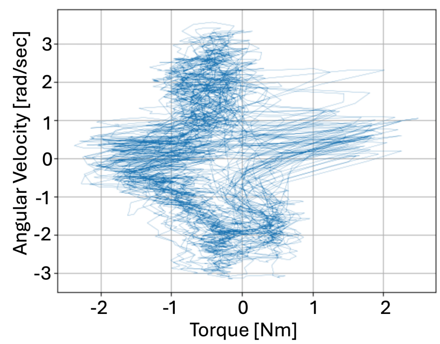


Fig. 2. Angular velocity versus torque landscape for an example user trajectory collected at walking speed $1.4\frac{m}{s}$ for 40s. The area of this curve represents the required power at the elbow. The maximum motor torque required nears 2.9Nm and maximum velocity nears $3\frac{rad}{s}$.

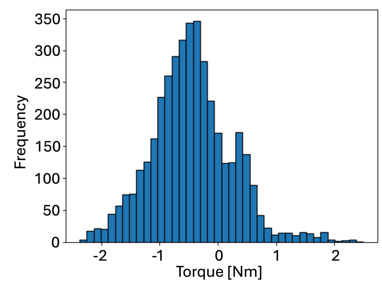


Fig. 3. Histogram of motor output torque over 40s trial collected at 100Hz with bucket size of 0.125Nm. Much less time was spent on torques outside $\pm 1\text{Nm}$. This challenges the torque specifications required from the motor.

The motor power envelope should encompass the entire angular velocity-torque landscape such that it encompasses the largest entries of elbow torque and angular velocity. Approximately, the largest angular velocity required is $3\frac{\text{rad}}{\text{s}}$ while the peak torque required is 2.9Nm.

To enable the arm to swing according to the natural arm-swing trajectory while minimizing energy use, the elbow joint should have minimal friction. A direct drive brushless DC motor is relatively low friction because there is no transmission: the only friction involved is that in the bearings.

The size of the motor was constrained by its mechanical design and prosthesis design concerns. Excessive weight of an upper limb prosthesis is often cited as a comfort-related reason for device abandonment [9], so a design goal was to select a motor that minimized mass of the system.

To discover the true motor torque requirement, the calculated torque trajectory was decomposed into time spent at various torque output levels. This leads to Fig. 3 which shows the frequency at torque buckets of size 0.125Nm.

3) Control System: The most direct way to generate human arm swing during walking is to command the motor along predefined swing trajectories. However, motors take in current as their input. So, to use the desired positions as a set point for the motor we added a second layer to control the actual input into the motor. Each controller takes in an error determined by the difference between the desired and actual value. Then, the controller transforms the error and outputs it to the next step in the controller (Fig. 4). This describes a closed loop feedback system that drives the motor based on

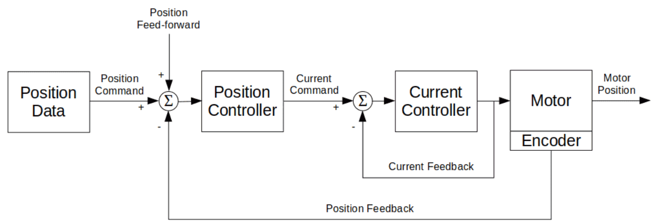


Fig. 4. Double nested closed loop feedback system. The Σ junctions represent the controllers.

the error generated. With control loops running fast enough and arm swing trajectories to follow this system will output arm swing trajectories.

4) Evaluation: Prototypes were tested to assess functionality as the device was carried on a person's arm while walking on a treadmill (qualified as development activity and *not human subjects research*). Prototype 1 was designed to validate arm swing trajectories at several fixed walking speeds that affected cadence: 0.7, 0.8, 1.0, 1.1, and $1.4 \frac{m}{s}$. Torque trajectories for each speed were predefined from unpublished inertial measurement data walking at each of these speeds. Performance of the elbow was evaluated by the expected versus actual angle output, and current data was collected to analyze the actual output torque.

B. Prototype 2

1) Motor Selection: Prototype 2 uses a motor with internal gearing, enabling use of a smaller motor that can still produce sufficient torque to enable arm swing. Although a motor with power capable of reproducing a natural swing trajectory could be selected, there was concern that a motor would exhibit excess speed but insufficient torque to complete the swing task. This disparity means that a motor with minimal power, geared down significantly, can satisfy the torque and angular velocity requirements at a considerably lower weight than a motor with sufficient torque but no gearing.

2) Controller: Prototype 2 uses the same control system as Prototype 1. Although performance disadvantages are anticipated with a geared motor, some of these may be alleviated by the controller. The position controller should track arm trajectories regardless of the gearing of the motor because any response that is slowed by the transmission characteristics will generate more error that the controller will account for in its response.

3) Step Detection and Adaptive Control: People do not walk at a fixed cadence during community ambulation. Therefore, a usable prosthetic arm should be able to track the wearer's cadence and match arm swing trajectory to that cadence such that it is synchronized with contralateral leg swing. Cadence was estimated through registration of foot initial contact (i.e., collision) with the ground. This real-time registration was performed through data collected from an accelerometer attached to the prosthetic elbow and a peak signal detection algorithm, meaning that the control system should be able to adapt to a change of cadence and update its dynamics accordingly (with some reasonable lag).

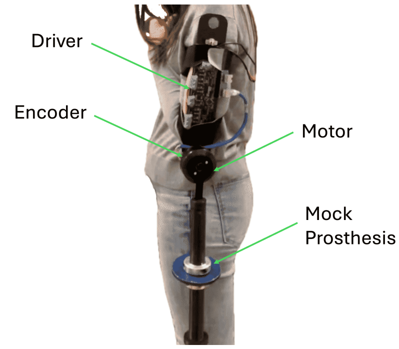


Fig. 5. The final Prototype 1 design features the motor and mock forearm that mimics the mass and inertia of a conventional prosthesis. The prototype is suitable for simulation based controller testing.

4) Evaluation: As with Prototype 1, Prototype 2 was tested as a person carried the prosthesis on their arm and walked on a treadmill. Arm swing trajectories of the wearer were collected at walking speeds of 0.7, 0.9, 1.1, and $1.3 \frac{m}{s}$ to be used in the controller and tested during trials at the same speeds. Tests were conducted both at individual speeds to test the performance of the controller as well as at variable speed to test the step detection and adaptive control. The prototype was also tested on the bench without the prosthetic forearm and with the axis aligned with gravity, thus removing any external force on the motor.

III. Results

A. Prototype 1

1) Final Design Overview: The final design of Prototype 1 as seen in [10, Fig. 5] is composed of a driver, encoder, motor, and mock prosthesis. The motor selected is a direct drive system (R60, CubeMars, Jiangxi, China) with a peak torque of $2.3\mathrm{Nm}$, making it capable of outputting almost any torque required based on the Euler-Lagrange estimation. The mock prosthesis consisted of a plastic tube and disc weights that were positioned to simulate the length, mass, and center-of-mass position of a conventional prosthetics forearm/terminal device, which matched the forearm component in the arm-body dynamics model.

The prosthesis was controlled using a PIDF (proportional, integral, derivative, feed forward) controller to correct for error in position and current. The gains were tuned by hand during bench testing. The PIDF controller is simple and widely used, making it the first choice for this system. In addition to error generated response, the feed forward term generates output based on the desired value, resulting in a faster response.

2) Evaluation Results: The results from testing show that Prototype 1 performed well in tracking the desired trajectories. As seen in Fig. 6 the output position matches closely with the desired, and hovers between $\pm 5\%$ absolute position error.

When comparing the actual and theoretical torques as calculated previously, we discovered that the prosthesis does not require as much torque as was predicted to accurately track the trajectory. Fig. 7 shows that the actual torque output

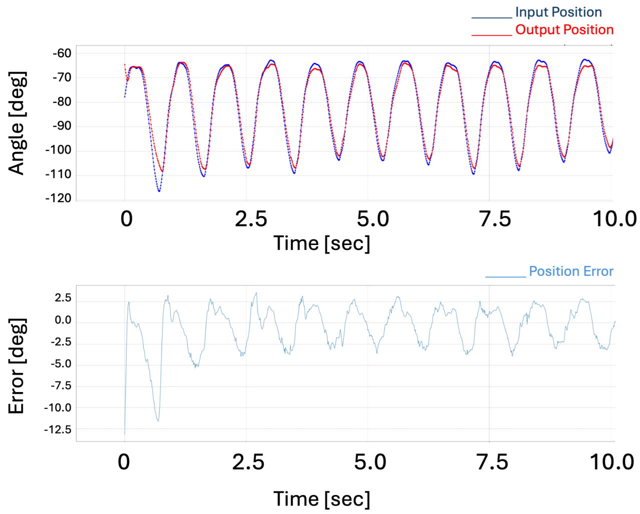


Fig. 6. Actual and desired position for Prototype 1 during testing (top), and error (bottom). The desired position was tracked with less than 5% error.

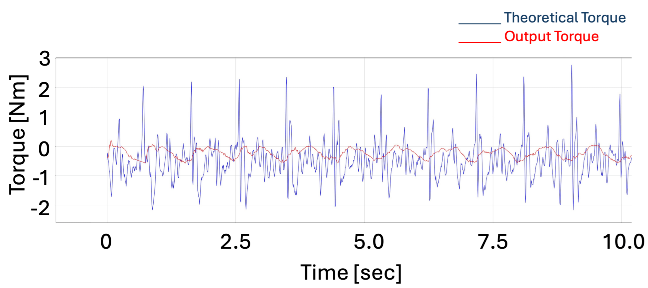


Fig. 7. Actual torque output superimposed on theoretical torque. The red signal displays the actual torque required to follow trajectories during testing, from its smallest to largest torques it ranges a total of 1Nm. This result suggests use smaller motors to achieve the same results achieved before.

varies at most 1Nm from the smallest to the largest actual torque outputs (compared to the simulations predicted range of 4.6Nm from smallest output torque to largest).

B. Prototype 2

1) Final Design Overview: Prototype 2 in Fig. 8 takes on a form similar to typical prosthetic arms, featuring an elbow that interfaces with a prosthetic socket, a prosthetic forearm with terminal device, and an external electronics center. The elbow joint motor interface and housing were designed around a commercial elbow joint (E400, Fillauer, Chattanooga, TN). Future iterations will condense the electronics such that it can be discretely fit inside a hollow forearm segment.

The motor selected for this iteration was a Coreless Servo (HiWonder, Shenzhen, China) with peak torque output of 3.4Nm. This high output torque from a smaller motor is achieved via its gearbox with a gear ratio of 1:342. Servo motors are typically controlled by a PWM signal that corresponds to a position. Instead of using this capability, which is sometimes inaccurate, the servo potentiometer was used as an encoder and the motor was controlled directly via its positive and negative terminals. Modifying the servo motor

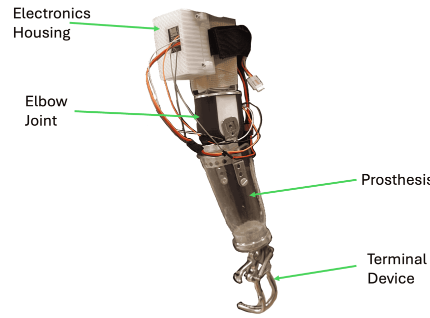


Fig. 8. Prototype 2 features an industry standard elbow envelope, conventional prosthetic socket and forearm/terminal device for universal application, as well as housed mechanical and electrical components.

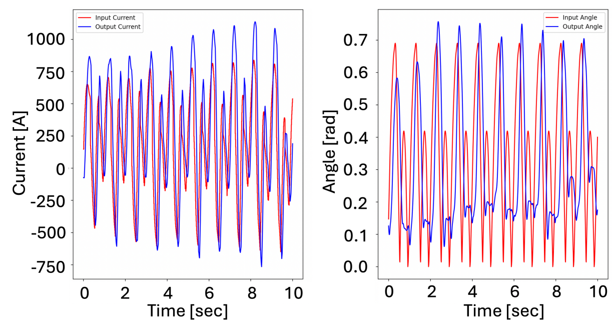


Fig. 9. Prototype 2 testing results with position control collected at $1.3\frac{m}{s}$ walking speed, displaying current (left) and position (right) output. There is deadband between the desired and actual output position. The controller tracks the desired trajectory with both overshoot and undershoot depending on the flexion or extension phase.

and incorporating current sensing allows for the same double nested control system as in Prototype 1 to be used for this motor (although the high gear ratio makes current sensing and torque control relatively inaccurate).

The controller for Prototype 2 uses a PIDF controller with a modified version of feed forward control to help overcome gearbox friction. Regular feed forward control is proportional to the desired output. However, the desired output is always nonnegative (since the elbow is restricted from hyperextension), so, it generates positive torque output from this term. Therefore, the feed forward coefficient was modified to be positive when the desired position was increasing (elbow flexion) and negative when the desired position was decreasing (elbow extension). This adjustment helps the motor overcome static gearbox friction when changing direction and kinetic gearbox friction during dynamic operations.

2) Evaluation Results: Exemplary results from testing with Prototype 2, displayed in Fig. 9, demonstrate success in tracking the shape of the arm swing trajectory. However, there is pronounced deadband between the desired and the actual trajectory, and the output does not track smaller arm swings presented in the desired trajectory.

Other results from Prototype 2 testing displayed less desirable behavior from the controller. In Fig. 10 the output position is overshooting and undershooting as it comes in and out of phase with the target arm swing frequency. The phase difference was mitigated if the wearer aligned their cadence with the prosthesis output. However, if there was any discrepancy between the controller and the target arm swing trajectory, results like Fig. 10 were produced.

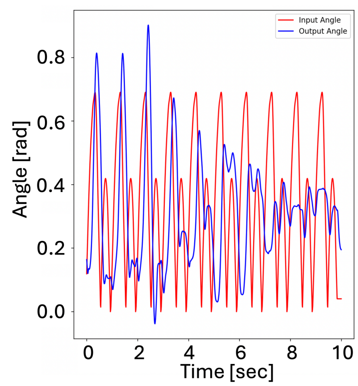


Fig. 10. Prototype 2 testing results with position control collected at $1.3\frac{m}{s}$ walking speed. The controller was unable to track the desired trajectory as it comes in and out of phase with the target arm swing.

The results in Fig. 11 are from testing with external forces removed from the motor to reveal the pure behavior of the motor within the control system.

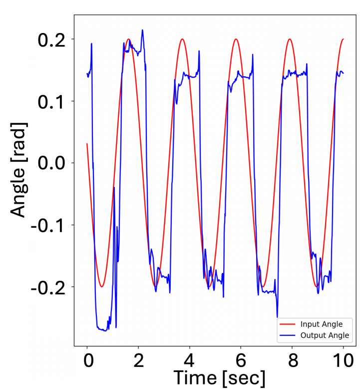


Fig. 11. Bench testing results on elbow joint tracking a sinusoidal trajectory with prosthesis and external forces due to gravity removed. The fast acceleration plateaus magnify the problems presented by gearbox friction that prevents the motor from moving at slow speeds from rest.

3) Adaptive Control and Step Detection: Initial results from testing of the step detection and adaptive control were promising. The step detection algorithm was able to accurately identify cadence when the wearer walked on solid ground, however, during treadmill walking the impact of footfalls was diminished and not registered by the automatic algorithm, thereby reducing its efficacy and challenging adaptive control. Future work will continue to refine this algorithm to enhance accuracy across different terrain.

IV. Discussion and Conclusion

1) Controller: The results of Prototypes 1 and 2 indicate that the same controller can have vastly different results depending on the prosthesis design. The promising results from Prototype 1 indicate that a double-nested position and current controller can work. The difficulties manifested in Prototype 2 are therefore likely resulting from the motor and transmission selection.

For Prototype 2, desynchronization with the wearer's cadence was likely due to the absence of control on the timing of the response. The desired arm swing trajectory was not aligned with any biomechanical marker such that the output phase is correctly timed. Issues with phase synchronization should be addressed in future iterations such that the arm auto-aligns its swing frequency with that of the user. This improvement would solve problems with undershooting the desired arm swing trajectory: phase alignment would prevent destructive interference between the elbow output torque and natural arm swing.

The overshoot of the desired arm swing trajectory during testing of Prototype 2 in Fig. 9 and Fig. 10 may be caused by the control system itself. The position control system should, in theory, be able to compensate for the additional energy added to the system by shoulder extension during walking as it monitors the position output in real time. However, the trajectory overshoot may be explained by the controller not anticipating extra energy added to the system by shoulder flexion/extension. When the torque output and arm momentum combine it generates a trajectory that overshoots the desired position. This interpretation is supported by the current output in Fig. 9. The motor current suddenly jumps during exaggerated arm swing, indicating that external forces drove the motor farther and required the controller to send more current through the motor to maintain the trajectory.

2) Motors: Prototype 2 testing demonstrated that use of a direct drive motor as in Prototype 1 has advantages. The sudden accelerations observed in Fig. 11 during bench testing likely resulted from internal gearbox friction. The controller is sending current to the motor proportional to the desired output, but the motor does not move until the torque output can overcome static friction. These jumps are then a manifestation of the limits of the motors operating torque from rest. If the motor was moving in one direction continuously, this characteristic would not be problematic. In this case the motor could not reflect the periodic flexion/extension of natural arm swing.

Sudden accelerations after plateaus did not appear during walking tests, likely due to the comparatively higher arm swing trajectory frequency. There was a deadband which manifested in the output difference between the desired and actual position. As observed in bench test results (Fig. 11) the motor is unable to track a sinusoidal trajectory without external forces. Since testing did not reveal firmware issues, we theorize that gearbox friction is the source of this deadband.

A. Future Work

1) Motor Selection: A direct-drive brushless motor has proven essential, offering significant advantages over geared motors. While direct-drive motors tend to be larger, Prototype 1 results suggest that a smaller, lighter brushless motor

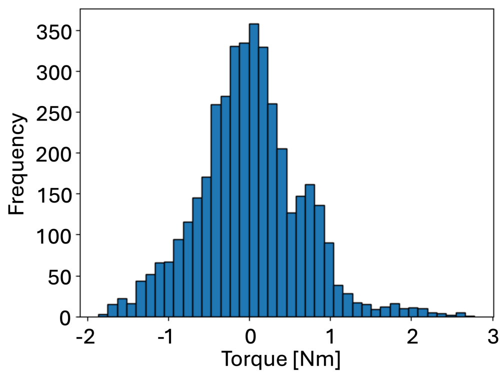


Fig. 12. Histogram of motor output torque over a 40s trial collected at $100\mathrm{Hz}$ with a bucket size of $0.125\mathrm{Nm}$. These results incorporate a torsional spring with stiffness $k = 0.5\frac{\mathrm{Nm}}{\mathrm{rad}}$, which shifts the distribution's center toward zero, moving much of the torque output into the $\pm 1\mathrm{Nm}$ range. This design would allow use of a smaller motor.

could achieve similar torque output without compromising performance and still reduce device weight.

As seen in Fig. 3, the torque output was not centered at zero and the range is negative biased. Hence, a motor tasked with tracking the torque trajectory must output higher magnitudes than if the distribution was zero-centered. A design solution would be to add a torsional spring that reduces the maximum torque output required from the motor and allows a smaller motor to be used. The simulated histogram results from adding a spring with stiffness $k = 0.5\frac{\mathrm{Nm}}{\mathrm{rad}}$ to the arm-body model can be seen in Fig. 12, which is now shifted towards zero.

2) Control System: As for controlling the arm, there was success in using the existing control system. However, it has been shown that the controller output needs to be aligned with some wearer input such that its frequency matches the wearer's dynamics. Foot initial contact as estimated through data collected by the embedded accelerometer is a simple biomechanical marker to record and would correlate to desired time signatures in an arm swing trajectory.

3) Adaptive Response: Adapting the trajectory of the prosthetic arm to the user's walking cadence is a critical step to improve gait symmetry and natural movement. This adaptation can be achieved by reliably estimating the walking cadence through real-time measurement of the initial contact of the foot in different terrains and walking speeds. Consequently, the arm can dynamically adjust its trajectory to match the user's cadence, ensuring a smoother and more synchronized motion.

One possible approach to implementing this adaptive response is to use large datasets of arm swing trajectories collected from wearers at various walking speeds. The data would serve as a lookup table, such that the prosthesis would select the trajectory at a speed most similar to the user's walking speed. However, a library of generalized behavior would have to be generated that may not always map well to wearer dynamics, potentially requiring interpolation.

An alternative approach involves training a predictive model to generate arm swing trajectories based on the wearer's anthropometry (e.g. height, leg length, arm length). This model would generalize arm swing trajectories across users, and require easily measurable parameters to prepare (i.e., calibrate) the prosthesis for use. This solution reduces the need for extra data collection and enhances applicability of the prosthesis. Future work will implement these design approaches and conduct human subject trials to assess functionality of the prosthesis and its affect on user gait dynamics.

B. Conclusion

We have designed two embodiments of a mechatronic prosthetic elbow that has demonstrated the potential to restore symmetry in arm swing during walking for persons with transhumeral-level limb absence. The success of Prototype 1 demonstrated the viability of using a position-based controller with a brushless motor. Prototype 2 demonstrated the ability to integrate the elbow with a conventional prosthetic system and highlighted the potential for deeper user integration, paving the way for more intuitive and self-sufficient control systems. By combining insights from both prototypes, future iterations will focus on refining these features and conducting human subject trials to assess the impact of a motorized prosthetic arm on improving dynamic balance during walking and potentially reducing the risk of falls for upper limb prosthesis users.

Acknowledgment

We thank Zachary Hoegberg and Lily Centra for assistance with testing, and support from the NU Departments of Physical Medicine & Rehabilitation and Mechanical Engineering.

References

[1] H. Herr and M. Popovic, "Angular momentum in human walking," *Journal of Experimental Biology*, vol. 211, pp. 467–81, Feb 2008.

[2] M. J. Major, S. M. McConn, J. L. Zavaleta, R. Stine, and S. A. Gard, "Effects of upper limb loss and prosthesis use on proactive mechanisms of locomotor stability," *Journal of Electromyography and Kinesiology*, vol. 48, pp. 145–51, Oct 2019.

[3] J. A. Dunn, N. G. Gomez, B. Wong, S. K. Sinclair, H. B. Henninger, K. B. Foreman, and K. N. Bachus, "Transhumeral prosthesis use affects upper body kinematics and kinetics," *Gait & Posture*, vol. 112, pp. 59–66, Jul 2024.

[4] S. Topuz, E. Kirdi, A. I. Yalcin, O. Ulger, H. Keklicek, and G. Sener, "Effects of arm swing on spatiotemporal characteristics of gait in unilateral transhumeral amputees," *Gait & Posture*, vol. 68, pp. 95–100, Feb 2019.

[5] J. A. Kent and M. J. Major, "Asymmetry of mass and motion affects the regulation of whole-body angular momentum in individuals with upper limb absence," *Clinical biomechanics*, vol. 76, p. 105015, Jun 2020.

[6] J. A. Dunn, N. G. Gomez, B. Wong, S. K. Sinclair, K. B. Foreman, K. N. Bachus, and H. B. Henninger, "Transhumeral prosthesis use and disuse affects whole-body angular momentum," *Clinical Biomechanics*, vol. 120, p. 106365, Dec 2024.

[7] M. Major, "Fall prevalence and contributors to the likelihood of falling in persons with upper limb loss," *Physical therapy*, vol. 99, pp. 377–87, Apr 2019.

[8] M. Major, R. Knox, Y. Ben Shalom, and M. Elwin, "Method and system to activate limb movement in amputees," U.S. Patent 2021-035-02 17/705,665, May 3, 2021.

[9] L. Smail, C. Neal, C. Wilkins, and T. Packham, "Comfort and function remain key factors in upper limb prosthetic abandonment: findings of a scoping review." *Disability and rehabilitation. Assistive technology*, vol. 16, pp. 821–30, Nov 2021.

[10] M. Major, Y. Ben Shalom, and M. Elwin, "A motorized prosthetic elbow for restoring arm swing," *49th American Academy of Orthotists & Prosthetists Annual Meeting & Scientific Symposium*, Mar 2023.

APPENDIX B

Cubemars GL40 Parameters and Resources

This appendix describes the GL40 KV70 from Cubemars and provides resources given from the manufacturer [**13**].

- **Product Listing:** GL40 KV70

B.1. Motor Drawing

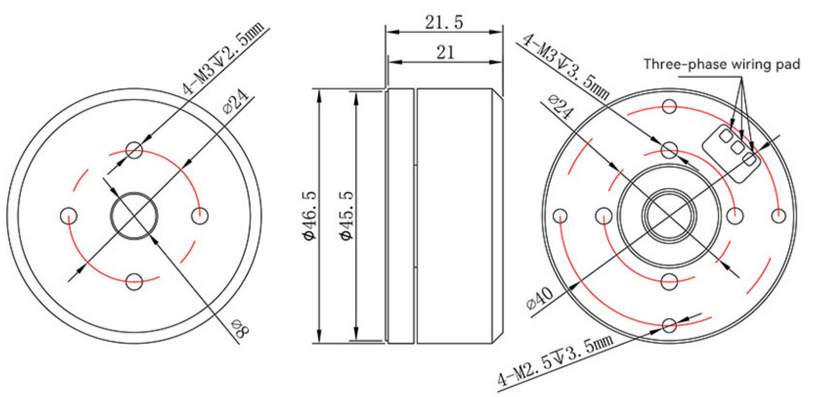


Figure B.1. GL40 Mechanical Drawing

B.2. Motor Specifications

Table B.1. KV70 Motor Specifications

General Specifications		
Application	Gimbal, Radar	
Driving Way	FOC	
Operation Ambient Temperature	−20 °C to 50 °C	
Winding Type	Star	
Insulation Class	H	
Insulation High-Voltage	500 V	5 mA / 2 s
Insulation Resistance	500 V	10 MΩ
Phase	3	
Pole Pairs	14	

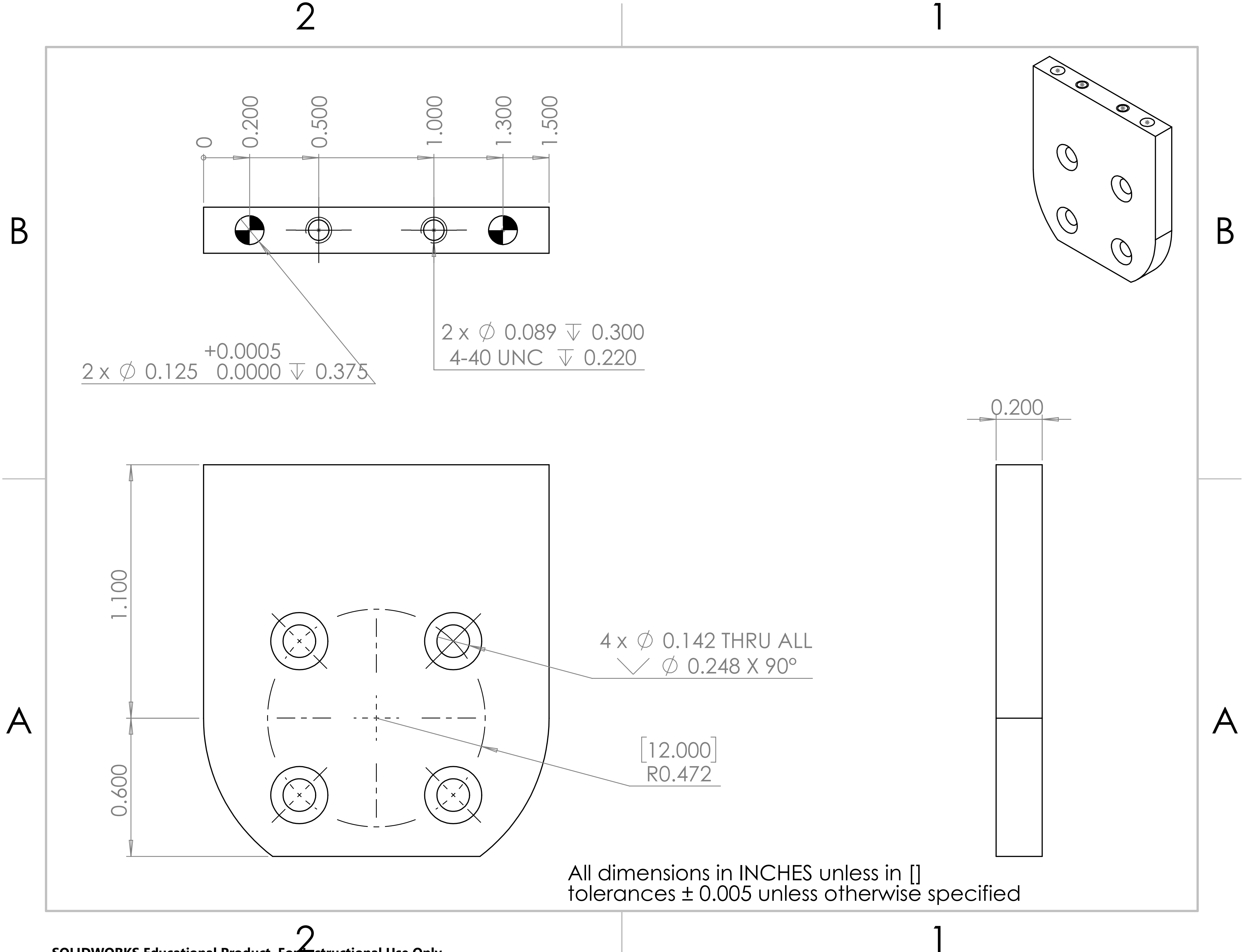
Electric Parameters (KV70)					
Rated Voltage	16	V	Rated Torque	0.25	N·m
K_e	15.00	V/krpm	Rated Speed	430	rpm
No-Load Speed	1015	rpm	Inertia	74	g·cm^2
Phase-to-Phase Resistance	4500	mΩ	Rated Current	1.62	A_{DC}
Phase-to-Phase Inductance	1800	μH	K_m	0.0707	N·m/$\sqrt{W}$
Peak Torque	0.5	N·m	Mechanical Time Constant	1.48	ms
Peak Current	3.3	A_{DC}	Electrical Time Constant	0.40	ms
K_v	63.5	rpm/V	Weight	107	g
K_t	0.150	N·m/A	Max Torque/Weight Ratio	4.67	N·m/kg

APPENDIX C

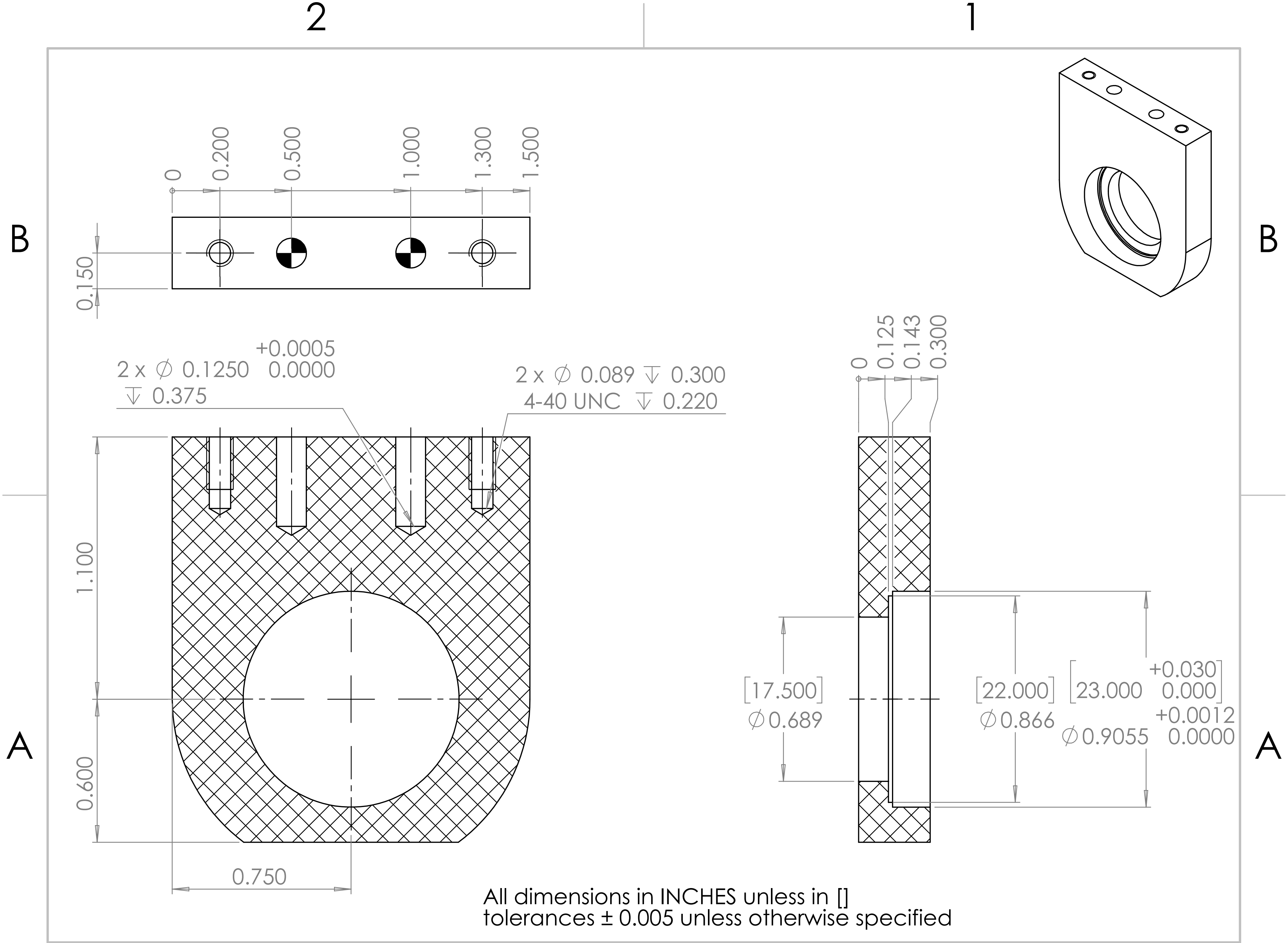
Prosthetic Elbow Mechanical Design Support Material

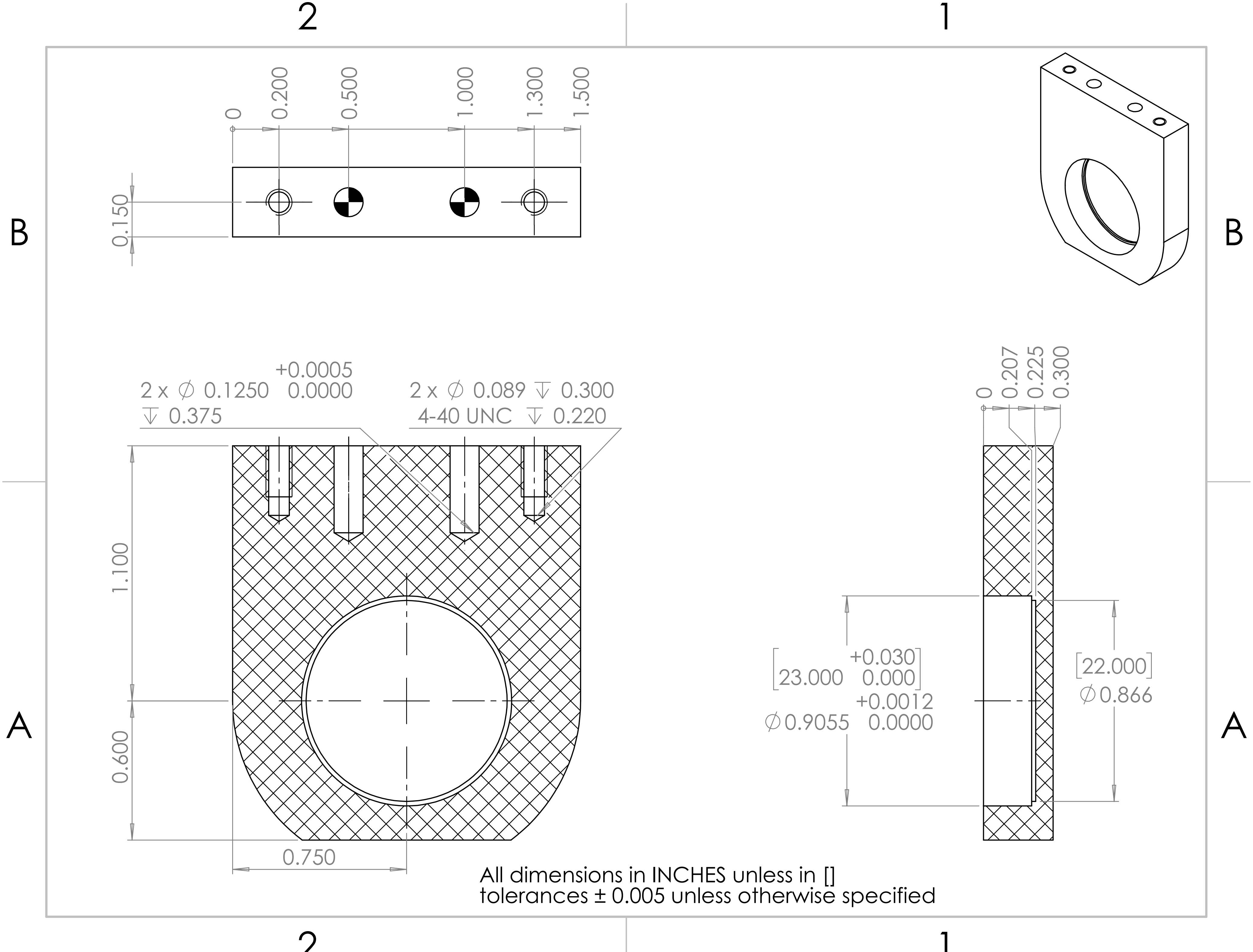
The CAD and other mechanical design files for this project are available at the following links:

- **GitHub Repository:** mechanical design
- **OnShape Assembly:** Onshape assembly

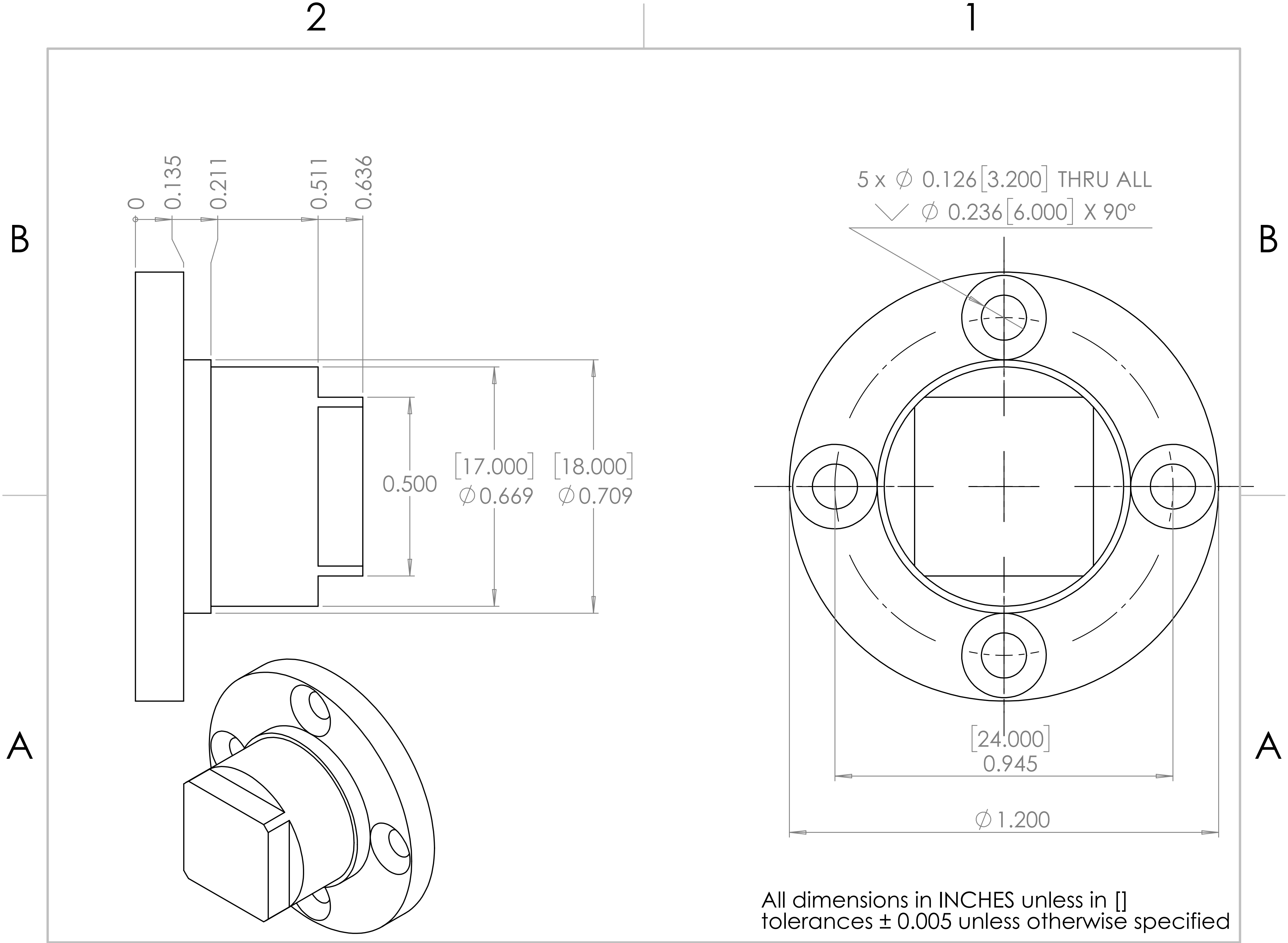


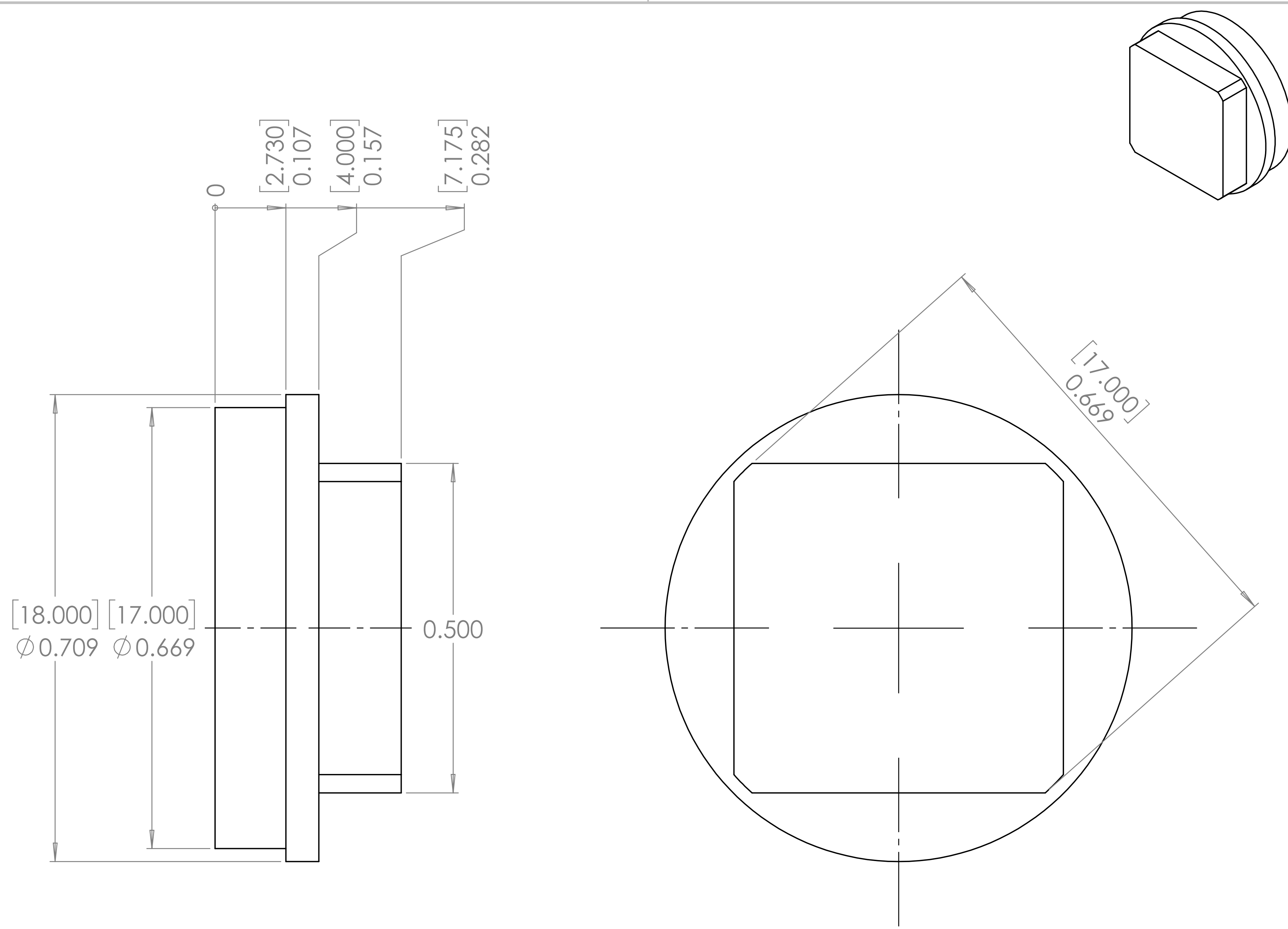
2
1
B
A
0
0.200
0.500
1.000
1.300
1.500
2 x ∅ 0.089 ↧ 0.300
4-40 UNC ↧ 0.220
+0.0005
2 x ∅ 0.125 0.0000 ↧ 0.375
0.200
1.100
0.600
4 x ∅ 0.142 THRU ALL
⌵ ∅ 0.248 X 90°
[12.000]
R0.472
All dimensions in INCHES unless in []
tolerances ± 0.005 unless otherwise specified


2
1
B
A
0
0.200
0.500
1.000
1.300
1.500
0.150
2 x ∅ 0.1250 +0.0005 0.0000
↧ 0.375
2 x ∅ 0.089 ↧ 0.300
4-40 UNC ↧ 0.220
1.100
0.600
0.750
0
0.125
0.143
0.300
[17.500]
∅0.689
[22.000]
∅0.866
[23.000 +0.030 0.000]
∅0.9055 +0.0012 0.0000
All dimensions in INCHES unless in []
tolerances ± 0.005 unless otherwise specified

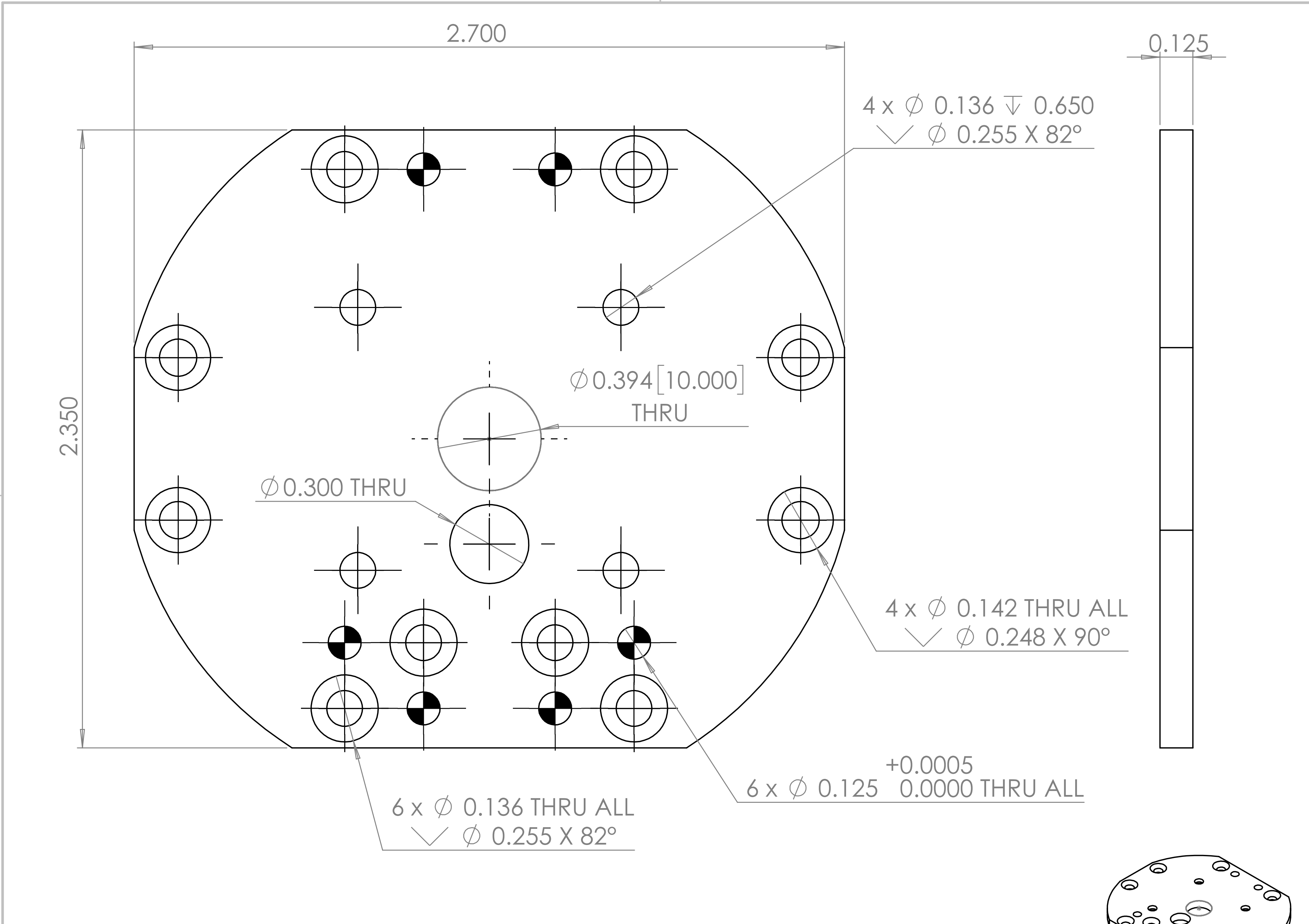


All dimensions in INCHES unless in []
tolerances ± 0.005 unless otherwise specified


2
1
B
A
0
0.135
0.211
0.511
0.636
0.500
[17.000]
∅0.669
[18.000]
∅0.709
5 x ∅ 0.126 [3.200] THRU ALL
⌵ ∅ 0.236 [6.000] X 90°
[24.000]
0.945
∅1.200
All dimensions in INCHES unless in []
tolerances ± 0.005 unless otherwise specified



All dimensions in INCHES unless in []
tolerances ± 0.005 unless otherwise specified



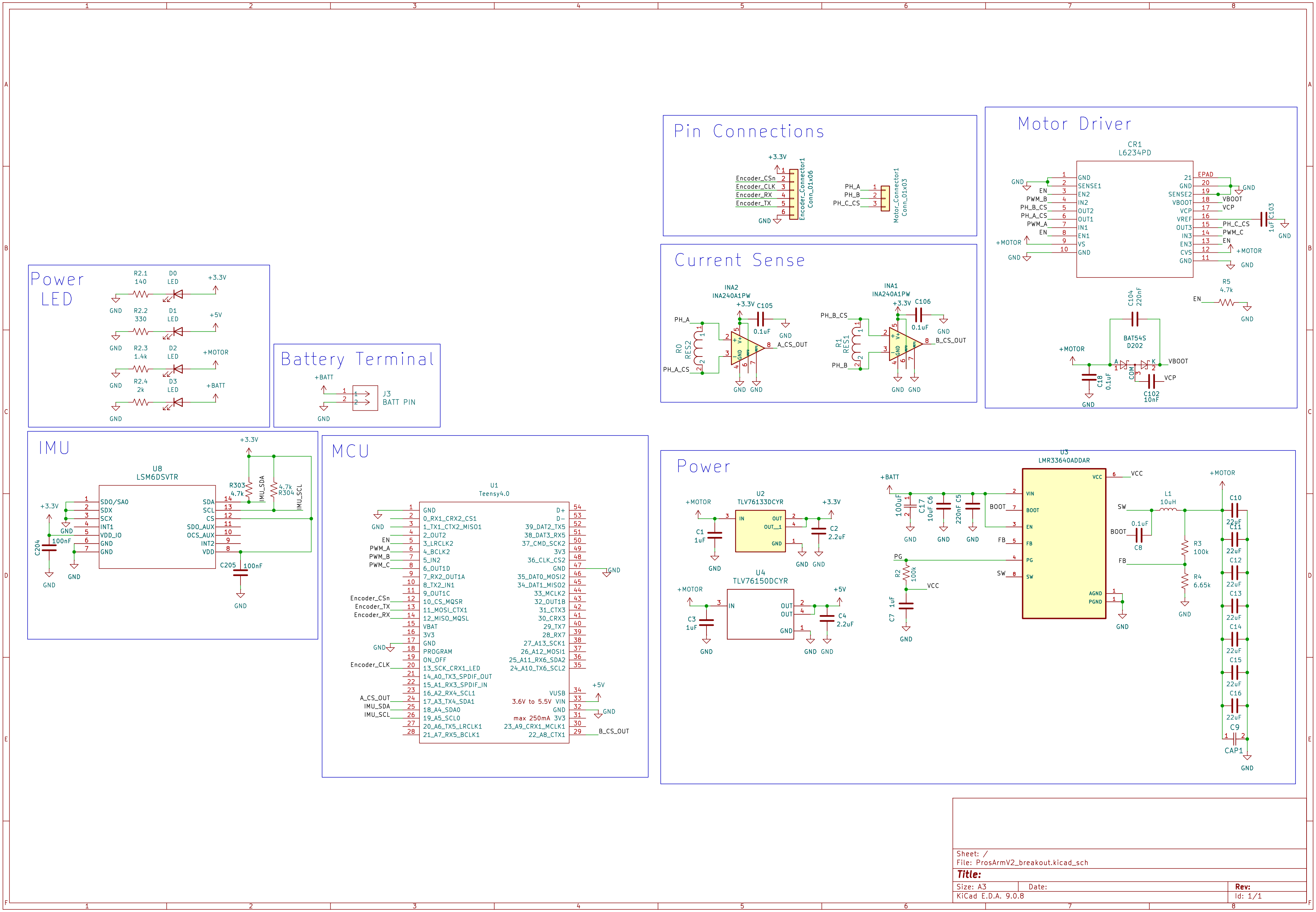
All dimensions in INCHES unless in []
tolerances ± 0.005 unless otherwise specified

APPENDIX D

Prosthetic Elbow Electrical Design Support Material

The KiCad and other electrical design files can be found at the following links:

- **GitHub Repository:** electrical design



Pin Connections
+3.3V
Encoder_CSn
Encoder_CLK
Encoder_RX
Encoder_TX
GND
Encoder_Connector1
Conn_01x06
PH_A
PH_B
PH_C_CS
Motor_Connector1
Conn_01x03
Motor Driver
CR1
L6234PD
GND
SENSE1
EN2
IN2
OUT2
OUT1
IN1
EN1
VS
EPAD
SENSE2
VBOOT
VCP
VREF
OUT3
IN3
EN3
CVS
EN
PWM_B
PH_B_CS
PH_A_CS
PWM_A
+MOTOR
PH_C_CS
PWM_C
C103
1uF
R5
4.7k
C104
220nF
BAT54S
D202
COM
C18
0.1uF
C102
10nF
Current Sense
INA2
INA240A1PW
C105
0.1uF
A_CS_OUT
R0
RES2
INA1
C106
R1
RES1
B_CS_OUT
Power
LED
R2.1
140
R2.2
330
R2.3
1.4k
R2.4
2k
D0
D1
D2
D3
LED
+5V
+BATT
Battery Terminal
J3
BATT PIN
IMU
U8
LSM6DSVTR
SDO/SA0
SDX
SCX
INT1
VDD_IO
SDA
SCL
CS
SDO_AUX
OCS_AUX
INT2
VDD
R303
4.7k
R304
IMU_SDA
IMU_SCL
C204
100nF
C205
MCU
U1
Teensy4.0
0_RX1_CRX2_CS1
1_TX1_CTX2_MISO1
2_OUT2
3_LRCLK2
4_BCLK2
5_IN2
6_OUT1D
7_RX2_OUT1A
8_TX2_IN1
9_OUT1C
10_CS_MQSR
11_MOSI_CTX1
12_MISO_MQSL
VBAT
3V3
PROGRAM
ON_OFF
13_SCK_CRX1_LED
14_A0_TX3_SPDIF_OUT
15_A1_RX3_SPDIF_IN
16_A2_RX4_SCL1
17_A3_TX4_SDA1
18_A4_SDA0
19_A5_SCL0
20_A6_TX5_LRCLK1
21_A7_RX5_BCLK1
D+
D-
39_DAT2_TX5
38_DAT3_RX5
37_CMD_SCK2
36_CLK_CS2
35_DAT0_MOSI2
34_DAT1_MISO2
33_MCLK2
32_OUT1B
31_CTX3
30_CRX3
29_TX7
28_RX7
27_A13_SCK1
26_A12_MOSI1
25_A11_RX6_SDA2
24_A10_TX6_SCL2
VUSB
3.6V to 5.5V VIN
max 250mA 3V3
23_A9_CRX1_MCLK1
22_A8_CTX1
U2
TLV76133DCYR
IN
OUT
OUT_1
C1
1uF
C2
2.2uF
U4
TLV76150DCYR
C3
C4
U3
LMR33640ADDAR
VIN
BOOT
FB
PG
SW
VCC
AGND
PGND
C17
100uF
C6
10uF
C5
220nF
R2
100k
C7
1uF
L1
10uH
C8
R3
100k
R4
6.65k
C10
C11
C12
C13
C14
C15
C16
22uF
C9
CAP1
Sheet: /
File: ProsArmV2_breakout.kicad_sch
Title:
Size: A3
Date:
KiCad E.D.A. 9.0.8
Rev:
Id: 1/1

APPENDIX E

Data Collection Support Material

The implemented data processing pipeline can be found here:

- **GitHub Repository:** data collection pipeline

APPENDIX F

Controls Support Material

The control code can be found here:

- **GitHub Repository:** motor control code
- **GitHub Repository:** prosthetic arm data collection code