2017

A Novel 3D printed leg design for a Biped Robot

Matthew A. Haywood
mxh7801@rit.edu

A Novel 3D printed leg design for a Biped Robot

by

Matthew A. Haywood

A Thesis Submitted in Partial Fulfillment of the Requirements for the
Degree of Master of Science in Electrical and Microelectronic Engineering

Supervised by

Professor Dr. Ferat Sahin
Department of Electrical and Microelectronic Engineering
Kate Gleason College of Engineering
Rochester Institute of Technology
Rochester, New York
2017

Approved by:

Dr. Ferat Sahin, Professor
Thesis Advisor, Department of Electrical and Microelectronic Engineering

---

Dr. Sildomar Monteiro, Assistant Professor
Committee Member, Department of Electrical and Microelectronic Engineering

---

Dr. Sohail Dianat, Professor
Committee Member, Department of Electrical and Microelectronic Engineering

Thesis Release Permission Form

Rochester Institute of Technology
Kate Gleason College of Engineering

Title:

A Novel 3D printed leg design for a Biped Robot

I, Matthew A. Haywood, hereby grant permission to the Wallace Memorial Library to reproduce my thesis in whole or part.

Matthew A. Haywood

Date

Dedication

This thesis is dedicated to my family, friends, and colleagues for their continued support and assistance.

Acknowledgments

I would like to thank Dr. Ferat Sahin for advising my thesis.

I am thankful to my parents for giving support and encouraging my engineering interests from a young age.

I am appreciative of my colleagues in the Multi Agent Bio-Robotics Laboratory. To everyone, I am grateful for the interesting conversations and useful input.

Abstract

A Novel 3D printed leg design for a Biped Robot

Matthew A. Haywood

Supervising Professor: Dr. Ferat Sahin

This paper proposes a novel leg design for a humanoid robot that can be 3D printed. More explicitly, the efforts of this paper are to bring some of the more complex leg designs seen in large scale bipedal robot into the realm of smaller bipeds while still allowing for it to be easily reproducible or modified. In order to accomplish this 3D printing technology was utilized, as well as an iterative design process. An ankle and knee powered by linear actuators were first constructed to test the conceptual design of the leg. This was followed by a complete leg design with improved ankle and knee, along with the rest of the leg.

List of Contributions

- Implementation of a 3D printed 7 DoF biped leg inspired by large scale humanoid robot
- Provided a bases for a complex leg design that can be easily reproduced.
- M. Haywood, F. Sahin, “A Novel 3D printed leg design for a Biped Robot,” *System of Systems Engineering (SoSE), 2017 IEEE International Conference on*, 2017. Accepted for publication.

Contents

List of Figures

Chapter 1

Introduction

The advancements of humanoid robots in recent years has been impressive but there is still research yet to be explored. The reason for interest in humanoid robotics is because legged robots offer a major advantage over wheeled counterparts because they can maneuver in difficult terrain. This has been seen in recent years with ATLAS from Boston Dynamics[4]. Outside of their capability in maneuvering, humanoid robotics can play a role in aiding elderly or disabled individuals who wish to continue living independently or be utilized in hazardous jobs where there is a high risk for loss of life.

Currently, humanoid robots range from large scale design with a high number of Degree-of-Freedom (DoF), to a small-scale design with few DoF often relying on a simple servo for each joint. Large scale models are often very complex with customized hardware, for example, LOLA [5] or WABIAN-2[6] humanoid robots. These types of robots utilize Harmonic drives[1] for each joint and typically allow for multiple DoF joint such as 3-DoF hip or ankle joint. In case of LOLA, a customized linear actuator design was used to produce a 2-DoF ankle. These customized components are either hard to reproduce or expensive to have them manufactured.

[1] A gearing system that allows for high torque with no backlash, discussed in Chapter 2

On the other hand, small-scale models are often too simplistic in design. For example, a direct drive servo for joint articulation: as is the case for DARwIn-OP[7] or the 7-DoF leg design presented in [8]. These designs are limited to the amount of torque the servo can generate and as with all things the more power needed, the higher the cost. The exception to the aforementioned robots is the Humanoid Robot NAO. This is due to its custom electrical and mechanical hardware, thus making it impossible to modify [9].

With the advancement and popularity of 3D printers in recent years, there has been an increase in printable bipedal robots. One of the more popular 3D printed humanoids is Poppy, which is also an open source platform [10]. The creator of Poppy discusses how 3D printers can aid in exploring morphological variants in bipedal robots. They have explored this with variations in shin length and with different foot designs. However, they still kept to a rather simple design approach of having the servos directly control the joints.

The objective of this paper is to bring some of the more complex leg designs seen in large scale bipedal robot into the realm of smaller bipeds while still allowing for ease of manufacturing. This paper first discusses some background information that will be needed through out in Chapter 2. There are two design iterations of the lower portion of a bipedal leg presented in the mechanical design, Chapter 3 along with a completed hip design. The electrical system is then be discussed in Chapter 4. Derivation of the forward and inverse kinematics will then be covered in Chapter 5, followed by a discussion of the simulation and modeling done in Chapter 6. This will lead into the results covered in Chapter 7. Conclusions are drawn in Chapter 8 along with Future work.

Chapter 2

Background Literature

2.1 Bipedal Walking Terminology

Here is a list of some basic terminology used in the study of Bipedal locomotion with a short definition of terms. This terminology will be used through out this paper.

Anatomical Planes

"An anatomical plane is a hypothetical plane used to transect the human body, in order to describe the location of structures or the direction of movements."[1] The planes used for the human body (and hence any humanoid robot) are shown in Figure 2.1. There are three anatomical planes and they are listed as follow:

- *Sagittal (Lateral)*: Divides the body into left and right
- *Coronal (Frontal)*: Divides the body into front and back sections.
- *Transverse (Horizontal)*: Divides the body into upper and lower segments.

[1] Source: en.wikipedia.org/wiki/Anatomical_plane

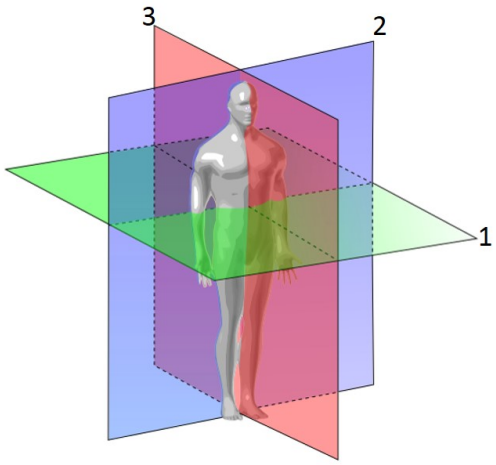


Figure 2.1: Anatomical planes: (1) Sagittal, (2) Coronal, (3) Transverse

2.1.1 Walking Gait

Walking Gait refers to a sequence of movements that propels a biped Center of Mass (COM) forward. The simplest walking gait is comprised of a DSP and SSP, but this can be further broken down into sub phases. Double Support Phase (DSP) is the state in which the biped is supported by both feet. The contact with the ground where as Single Support Phase (SSP) occurs when only one foot is in contact with the ground.

- The **swing leg** is defined as the leg that moves through the air during SSP
- The **stance leg** is the one that provides support during SSP.
- **Support Polygon** is the bounded area of the foot or feet, that has contact with the ground.
- **Center of Pressure (CoP)** is a point on the support polygon where the total sum of the tangential forces act.

2.2 Harmonic Drives

Harmonic drives are comprised of three basic components: a wave generator ①, flex spline ②, and circular spline ③. The number of teeth on the flex spline (N_f) are slightly less than the number of teeth on the circular spline (N_c). When the wave generator makes a full rotation the flex spline is shifted by this difference in teeth. The gear reduction ratio (R) is given by

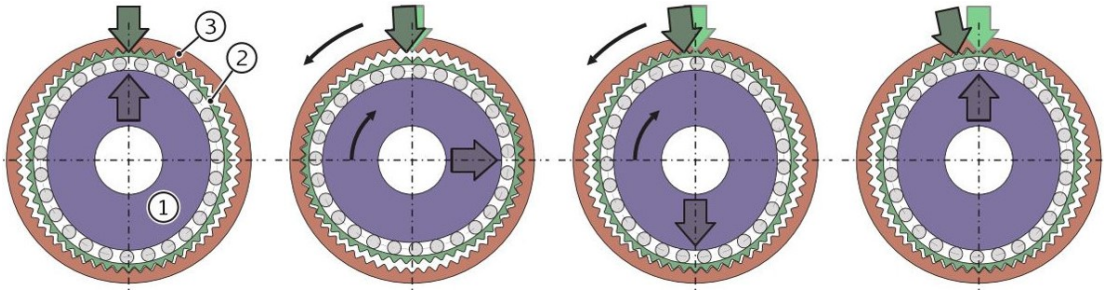


Figure 2.2: Operating principle of a Harmonic Drive

$$R = \frac{N_c - N_f}{N_c} \tag{2.1}$$

Advantages for harmonic drives include: no backlash, high gear ratios, small size, excellent repeatability, high efficiency, and high torque capability. The main disadvantage to them is they are expensive. More detailed information can be found in the book [11].

2.3 Lead Screw

Lead screw is a mechanism that transfer rotational actuation into linear motion. It is comprised of a screw shaft and a nut. The *Lead* is defined as the axial distance traveled for one complete rotation. (Also refer to Figure 3.4a ③ & ④)

2.4 RC Servo

Radio Controlled (RC) servos were developed for small remote control models but are also commonly used in small robotic applications. They care comprised of a motor, gearbox, feedback and a control board as the block diagram shown in Figure 2.3a shows. As it can be seen feedback is typically done with a potentiometer and is connected directly to the output shaft.

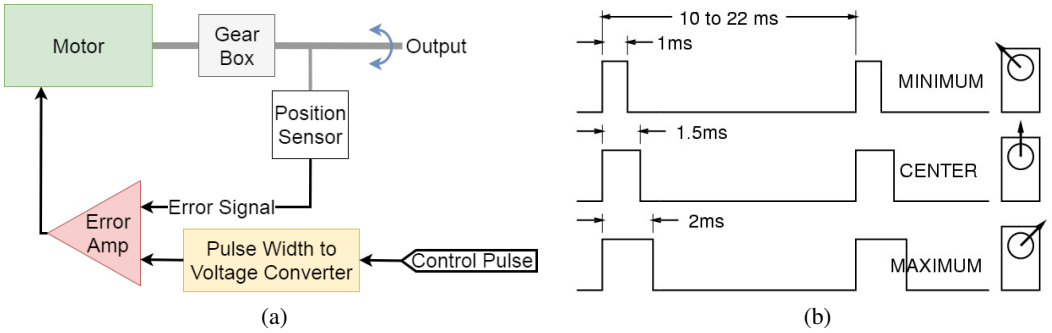


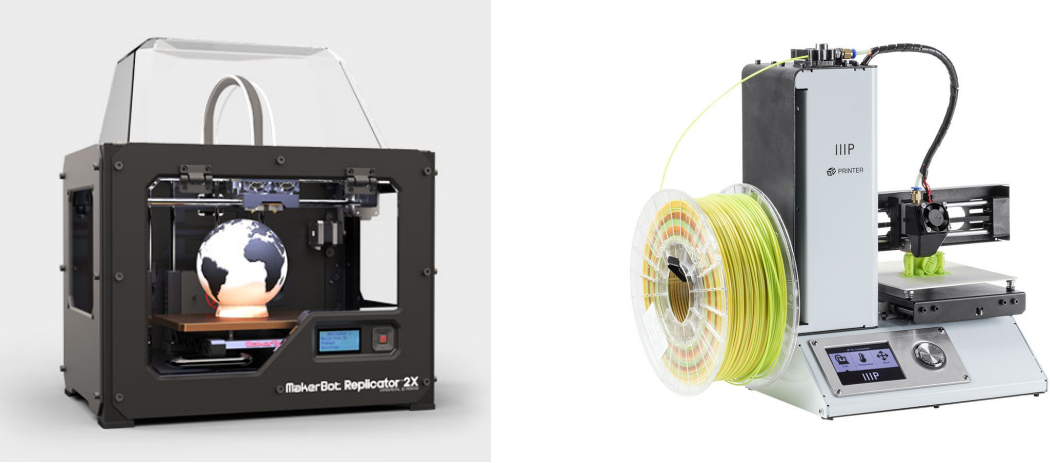
Figure 2.3: (a) Servo block diagram (b) RC Control signal

The control signal for a RC Servo is a form of Pulse Width Modulation (PWM), in which the angle is determined by time of the high pulse as shown in Figure 2.3b.

2.5 3D Printing

The technology of 3D printing has made small scale manufacturing and prototyping easy and accessible to hobbyist and researchers. It is an additive manufacturing process in which an object is created by laying down material one layer at a time. This is typically done with a polymer. The thickness of a layer is defined as the layer height and can be varied within a range (typical $0.1mm$ to $0.2mm$) based on desired amount.

The two printers used to create various parts of this project were Makerbot Replicator



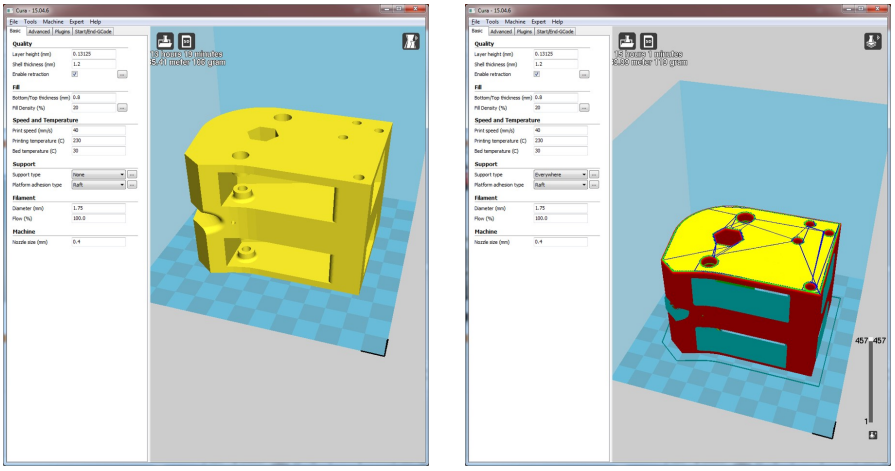
(a) (b)

Figure 2.4: (a) MakerBot Replicator 2X [1] (b) MP Select Mini [2]

2x, shown in Fig. 2.4a and MP Select mini, shown in Fig. 2.4b. Both of these are based on Fused Deposition Modeling (FDM) technology to create a part layer by layer. Acrylonitrile Butadiene Styrene (ABS) or Polylactic Acid (PLA) are the two commonly used materials for FDM type 3D printers. A spool of the chosen plastic is fed into the printing head where it is melted and pushed through the extruder in 2D layers that stack vertically, ultimately resulting in a 3D part. In order to build a part, it must first be modeled in CAD software, exported as a *.stl* file, and then brought into a slicer program that breaks the model into individual 2D layers.

Several parameters must be considered before choosing how to print a part. ABS requires high extruder and build plate temperatures while printing and should be printed with the build volume inside a closed container. Maintaining a constant build chamber temperature allows for this material to cool uniformly which prevents warping. Large ABS parts may warp and peel up from the print bed as the upper plastic layers cool faster and constrict more than the bottom layers closer to the heated build plate. By comparison, PLA has a

lower melting point and thus can be printed at lower temperatures but can create issues with parts in application that need to hold up to higher temperatures. It does not significantly shrink after cooling and has a much lower chance of warping. Parts printed in PLA can be more brittle than ABS, which generally flexes when stressed. Whether ABS or PLA is used, painter's tape is often applied to the build plate to enhance the adhesion of the first print layer. Other print parameters depend on characteristics of the part being printed. If



(a) (b)

Figure 2.5: The part brought into Cura in (a) has empty cavities that requires support to print properly. The sliced output in (b) shows an added raft underneath the part and support in the empty cavities.

the part has overhanging features or empty cavities, such as that in Figure 2.5a, support material must be printed below this geometry. Figure 2.5b shows the sliced output with the support present along with a raft underneath the part to improve adhesion to the build plate and helps to reduce any irregularities of the build plate surface. These assistive elements are removed after printing.

Parts can be rotated to reduce the amount of necessary support and minimize print time. Although parts can be printed at any orientation, keeping flat faces horizontal or vertical

Table 2.1: Key material properties for NinjaTek filaments

	NinjaFlex	**Cheetah**
Shore Hardness	85A	95A
Elongation	660%	580%
Abrasion Resistance	20% >ABS 68% >PLA	40% >ABS 76% >PLA
Impact Strength	NA	84% >ABS

will produce the smoothest results. The expected torsion and shear forces on the part also inform the optimal print orientation. A part is most susceptible to shearing between vertical layers with low adhesion area and breaking at thin geometry.

Even though ABS and PLA are the most common materials used there are a variety of other materials offered. For this design two other flexible polyurethane materials from NinjaTek (Ninjaflex and Cheetah) were utilized. Ninjaflex is made from a specially formulated thermoplastic polyurethane (TPU) material[12]. The key material properties for these two filaments are shown in Table 2.1

The manufactures suggested applications are: seals, gaskets, plugs, leveling feet, and protective applications. It has also been used for pneumatic application such as a gripper for soft robotics. Cheetah is a more rigid version of Ninjaflex with greater strength and durability[13]. The manufactures suggested applications are: seals, plugs, hinges, sleeves and snap-fit parts. For this design Cheetah was used for the flex spline of the Harmonic drive and Ninjaflex for sole of the foot, which will be discussed later on in Section 3.4.1.

2.6 Inter-Integrated Circuit (I^2C)

Inter-Integrated Circuit (I^2C) is a synchronous serial communication interface, meaning both the sender and receiver access the data according to the same clock. I^2C allows for multiple slave chips to communicate with one or more master chips. This is accomplished by each slave chip having it's own unique address. These address are sometime fixed or can be set with external pull-up or pull-down resistors depending on the manufacture specifications. There are only two lines required for communication which can support up to 1008 slave devices. The two lines are Serial Data Line (SDA) and Serial Clock Line (SCL). Unlike other forms of serial communication the driver outputs are "open collector", which means that a pull up resistor must be used on the bus lines as shown in Figure 2.6. Data rates for I^2C are either $100kHz$ or $400kHz$.

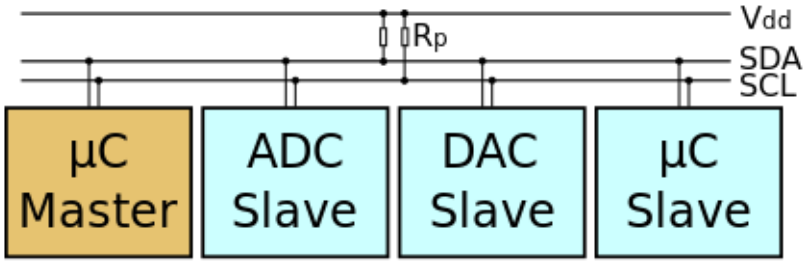


Figure 2.6: Block diagram of I^2C[3]

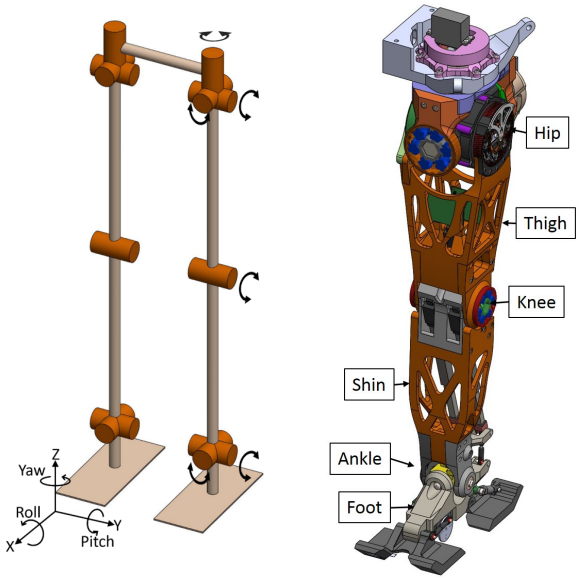
Chapter 3

MECHANICAL DESIGN

This chapter first outlines some general design specification in the following section. The following section are broken up into each joint with multiple designs being discussed for the ankle and knee.

3.1 Design Specifications

The general design specification that were followed in creation of this 3D printable biped leg are as follows: First to follow the kinematic scheme given in Figure 3.1a as this would appear to be a common scheme in large scale bipedal robots [6][14]. This kinematic scheme consist of six DoF in each leg with multiple DoF joints having intersecting axis of rotation. To accomplish the latter, 3D printable Harmonic drives should be utilized because of the advantages previously stated. To make it easily accessible the body will be 3D printed, which in turn will also reduce cost. Off the shelf components will be used to also aid in reducing cost of this design.



(a) (b)

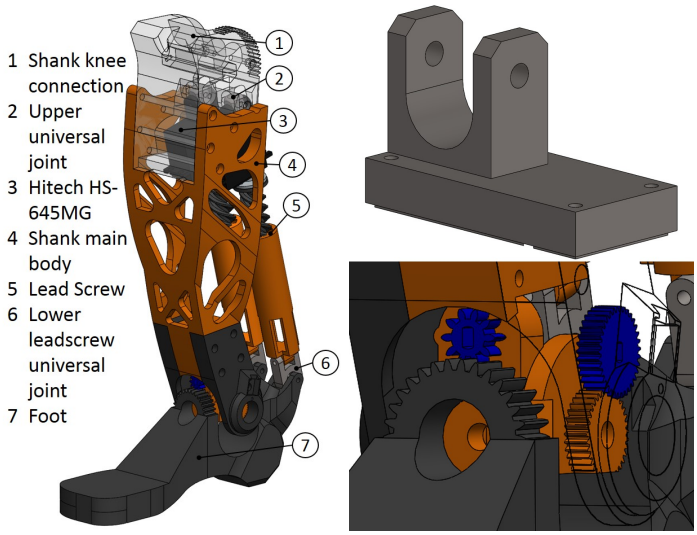
Figure 3.1: (a) Basic kinematic scheme for robot with 3 DoF Hip, 1 DoF Knee, 2 DoF Ankle, and rectangular foot (b) Final CAD model of 3d printed leg

3.2 Ankle

3.2.1 Design 1

In this design shown in Figure 3.2a, a 3D printed *lead screw* ⑤ with a four start thread and a pitch of 1.25 inches, was attached to a *Hitech HS-645MG* servo ③. These servos were modified for continuous rotation and a new bottom plate for the housing was printed for ease of mounting as shown in Figure 3.2b Top. Because of the continuous rotation modification, the potentiometer needed to be relocated to the ankle also shown in Figure 3.2b Bottom, in order to provide some feedback. The gear ratios for each DoF was chosen to provide maximum resolution from the potentiometer for the range of motion.

The linear actuator in this design wasn't properly braced and caused the actuators to twist slightly along the length. This combined with limited resolution of 3D printers caused



(a) (b)

Figure 3.2: (a) First design of ankle (b) Top: Hitech HS-645MG replacement bottom plate Bottom: Ankle close up of potentiometer mounting

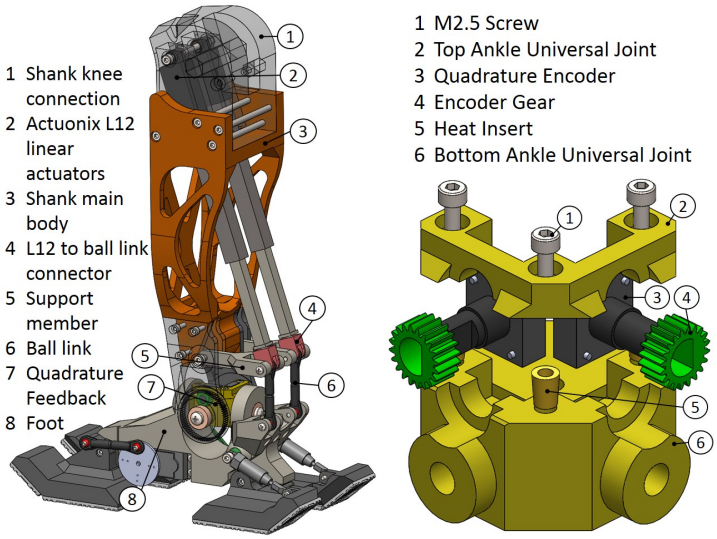
$\approx 10^{\circ}$ of play in the roll axis and $\approx 2^{\circ}$ of play in the pitch axes when the leg was powered off. While moving the roll angle would suddenly change drastically from a single degree to 5° for the same step motion. This was especially apparent while under load.

3.2.2 Design 2

The ankle design uses two linear actuators to provide 2 DoF rotational motion. When the actuators move in the same direction the ankle will rotate along the pitch axis. For the ankle to rotate around the roll axis, the actuators have to move in opposing directions. The first design was based off of LOLA[6] where as the second iteration was based off the ape robot presented in the works of [15][16][17].

The large size of the linear actuators used in the first design made it difficult to properly brace without increasing the size of the robot. So these were replaced with *Actuonix L12 linear actuators* ②. These provide customizable stroke length, gear ratio, and control

option all in a small size, making them an excellent replacement for the 3D printed version. The following option were chosen, 100 mm stroke length, 100:1 gear ratio, and RC servo input with potentiometer feedback.



(a) (b)

Figure 3.3: (a) Second ankle design (b) Exploded view of ankle universal joint

This updated design, as shown in Figure 3.3a, includes the addition of *two ball links* ⑥ and a *support member* ⑤ to create two four-bar linkage system similar to that of the ape robot [16]. It was decided not to 3D print the ball links because of the difficulty in printing spherical objects, especially at such a small size ($\approx 5mm$ dia.). It should be stated that given a higher resolution printer, the ball links could be printed. In order to connect the ball links to the linear actuators a custom *connector* was printed ④. This four-bar linkage system allows for the linear actuators to stay parallel with each other, simplifying the mounting for motors and reducing the collision area for moving parts. The ankle play was reduced to $\approx 5^{\circ}$ for the roll axis, while removing play in the pitch axis completely.

This ankle design also features *quadrature feedback* ⑦ built into the universal joint as shown in the exploded view of Figure 3.3b. Quadrature encoding can produce higher resolution with a faster sampling rate when compared to potentiometer style encoding. However, as quadrature feedback is not absolute, the position feedback of the linear actuator will be used to calibrate the quadrature at power on. The gearing for the encoders was changed to an internal spur gear with both position sensors being mounted to the ankle universal joint. This resulted in a more compact design that allowed simplification of the wiring harness, while the gearing increased the resolution of the encoders.

3.3 Knee

3.3.1 Design 1

A close up view of this first design is shown in Figure 3.4a. Similar to the ankle, a 3D printed *lead screw* (③ & ④) with a four start thread and a pitch of 1.25*inches* was attached to a *Hitech HS-755HB* servo ①. The continuous rotation modification was also done to this servo, and the potentiometer relocated to read the potion of the Knee through the use of a *spur gear* ⑤. This potentiometer reading was fed back into the servo as feedback to allow for the position control of the knee with existing internal servo circuitry. Two *gear bearing* ⑥ are used to brace each side of the knee. This helped reduce the amount of friction while adding support. A gear bearing was chosen because of it's ease of printing compared to conventional bearings. The gear bearing was an open source design that was customized based on the outer diameter, number of planet gears and number of teeth for planets and sun gears [18].

Due to limited resolution of 3D printers, the lead screw had too much backlash, causing

$\approx 10^{\circ}$ of play for the Knee at power off.

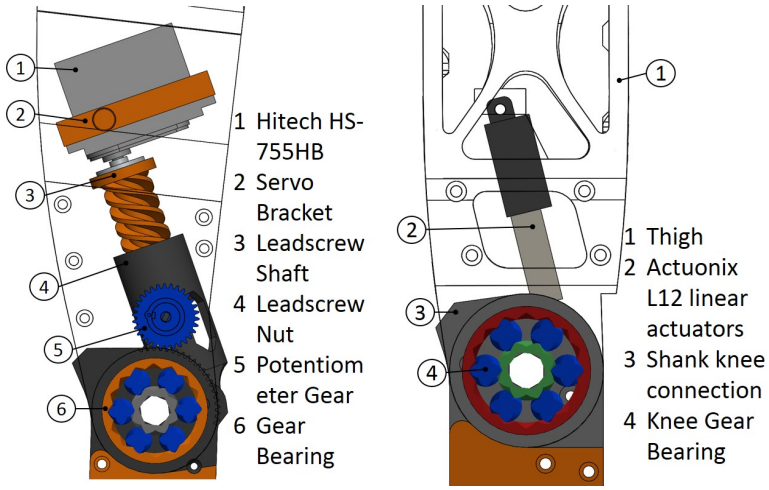


Figure 3.4: Close up view of first knee design (a) and second knee design (b)

3.3.2 Design 2

The difference between the potentiometer spur gear backlash and the lead screw caused osculation due to the servo constantly correcting itself, which was especially apparent while under load. On the other hand, the gear bearing provided the stability and proved to be the correct design choice.

The knee is a one DoF joint where the first design used a hobby servo with a 3D printed lead screw and the second design used an off the shelve linear actuator.

A close up view of the final knee design can be shown in Figure 3.4b. As with the ankle, an *Actuonix L12 linear actuator* ② replaced the 3D printed lead screw in the previous design. This reduced play in the Knee to $\approx 5^{\circ}$, half the amount seen previously. The smaller size of the actuator reduced overall thigh mass by approximately 13 grams and allowed room in the thigh for housing electronics. The *gear bearings* ④ from the previous

design were kept because the high stability and low friction.

3.4 Hip

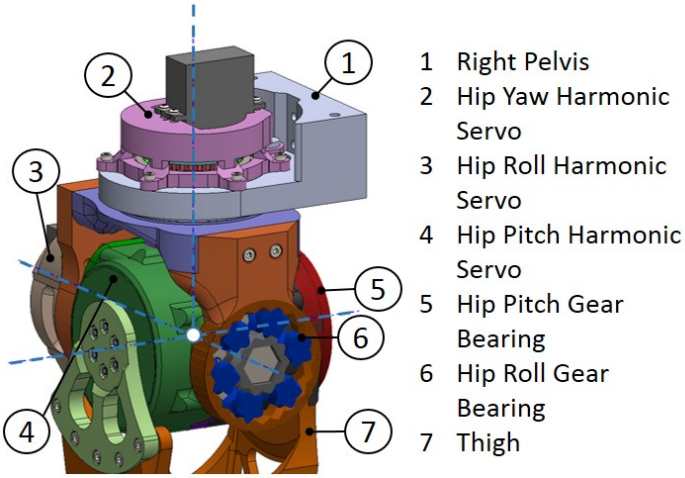


Figure 3.5: Three DoF Hip with Harmonic drive actuated joints

The hip has 3 DoF and is configured to mimic a ball joint as shown in Figure 3.5. This is accomplished by having the axis of rotations (blue dashed lines in Figure 3.5) intersect with each other. It is comprised of two smaller *harmonic drives* ② & ③ for Yaw and Roll rotations, a larger *harmonic drive* ④ for Pitch rotation, two *gear bearings* ⑤ & ⑥ and two thrust bearings as shown in Figure 3.4.

The smaller harmonic drive is an adaptation of the open source design provided in [19]. It is powered by a *Power HD-1501* servo ⑦ with it's *potentiometer* ⑩ connected via *spur gear* ⑨ to the *output shaft* ①. This allows for the servo to still be position controlled through the standard interface. The output shaft has a hexagon extrusion to act as a key for driving the thrust bearing. *Heat inserts* ⑧ where placed in the output shaft for easier attachment of the *flex spline* ⑤. The *housing* ② has two variation for Yaw and Roll axis,

Figure 3.6a and 3.6b respectively. The *wave generator* ⑥ is comprised of two bearings connected to servo horn via a 3D printed part. A *bearing* ③ comprised of 3D printed races and 3*mm* Delrin balls was used to help maintain concentric alignment between the output and housing.

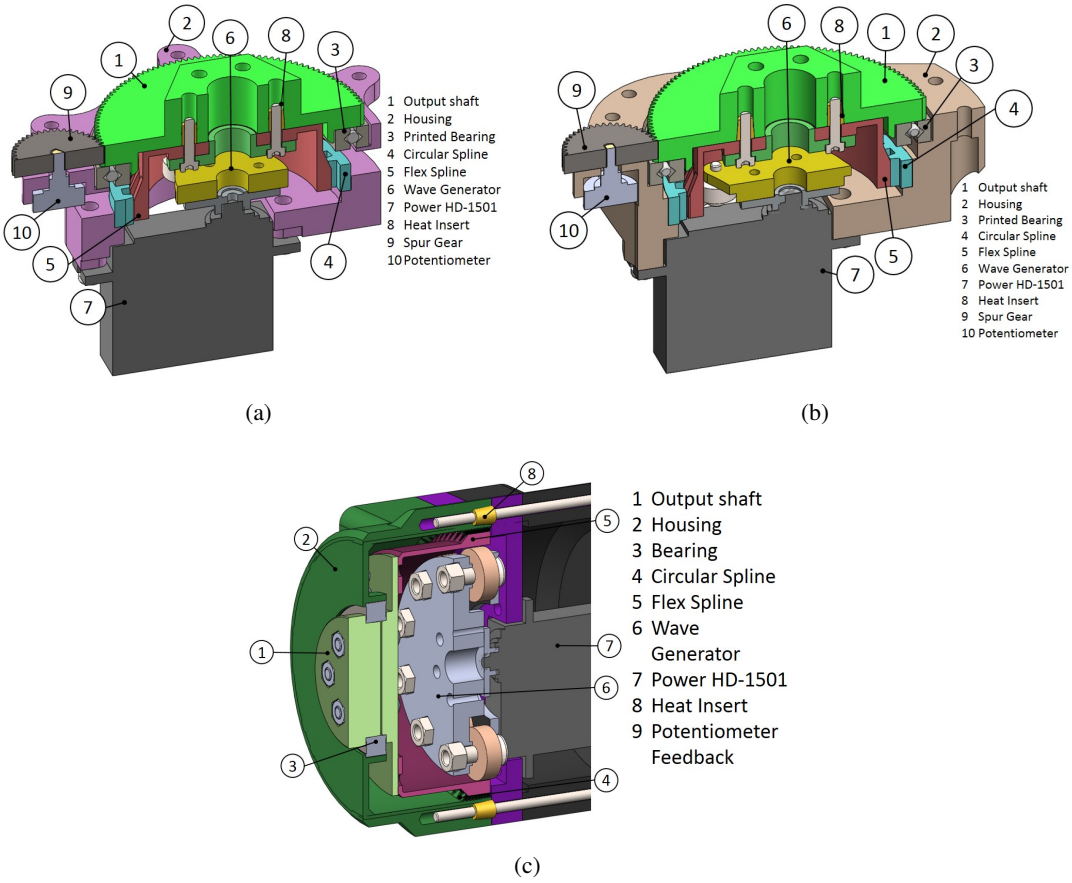


Figure 3.6: Sectional view of Harmonic drives for hip roll (a), yaw (b) and pitch (c) joint

The larger harmonic drive (Figure 3.6c) is an adaptation of the open source design provided in [20]. This design was modified to run off the same type of servo ⑦ used in the smaller harmonic drive, original design was intended for a stepper motor. The *output shaft* ① and *housing* ② were also modified so that it could be mounted in the leg. The

wave generator ⑥ is an elliptical disk with bearing mounted around the outside. This helps maintain the shape of the *flex spline* ⑤ in the Harmonic drive.

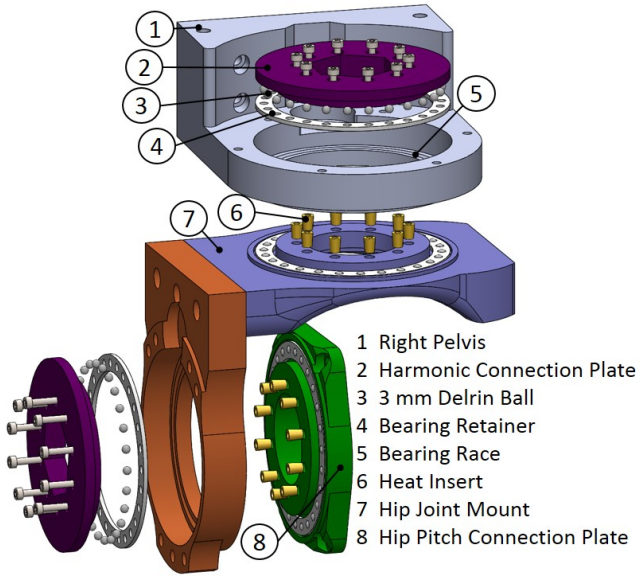


Figure 3.7: Thrust bearing for the Harmonic drives for the hips roll and yaw axes

As stated before the hip has two thrust bearing for the Yaw and Roll axes, which was done to reduce any unwanted forces or moments that would cause damage to the harmonic drives Figure 3.4. A Harmonic *connection plate* ② was developed to transfer motion between the Harmonic drive and joint. This sits on top of either the *pelvis* ① or *hip joint mount* ⑦ with 3*mm Delrin balls* ③ with a laser cut *bearing retainer* ④ sandwiched between them. The connection plate is bolted to either the *hip joint mount* ⑦ or the *hip pitch connection plate* ⑧.

3.4.1 Foot

This foot design, as shown in Figure 3.8, is comprised of two DoF joints: a passive heal and an active toe. The heal has two *shocks absorbers* ⑦ for a hobby RC car and are intended

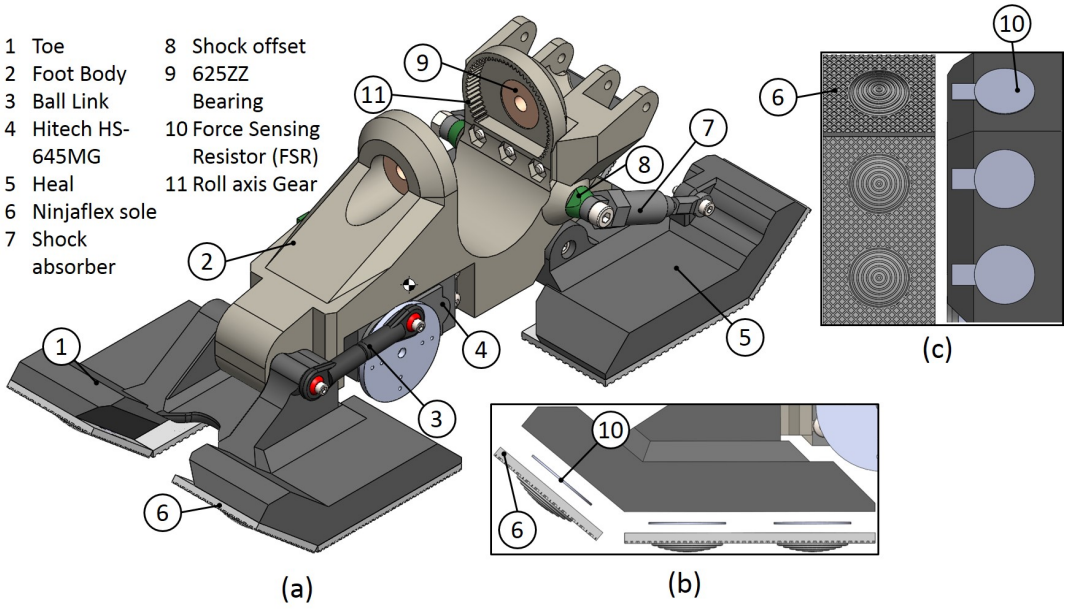


Figure 3.8: (a) Foot Assembly (b) Exploded view of Toe (c) Bottom view of the Toe with Ninjaflex sole (Left) and FSR (Right)

to reduce the forces seen during the impact phase of the walking gait when the heal touches the ground. The toe is connected via a *ball link* ③ to a *Hitech HS-645MG* servo ④. The primary reason for an active toe is because it has been shown to increase walking speed[21]. Both the heal and tow are equipped with an array of six *Force Sensing Resistors (FSR)* ⑩ distributed along the bottom. The material Ninjaflex was used to 3D print a sole for the foot and placed on top of the FSR adding cushioning and increase friction. The FSR array was implemented to determine the Zero Moment Point (ZMP)[1] position for the stance leg during the SSP of a walking gate. This is similar to the work presented in [23].

[1]ZMP is a point on the ground where the net moment vector of inertial & gravitation forces of the entire body has zero components in horizontal planes [22]

Chapter 4

Electrical System

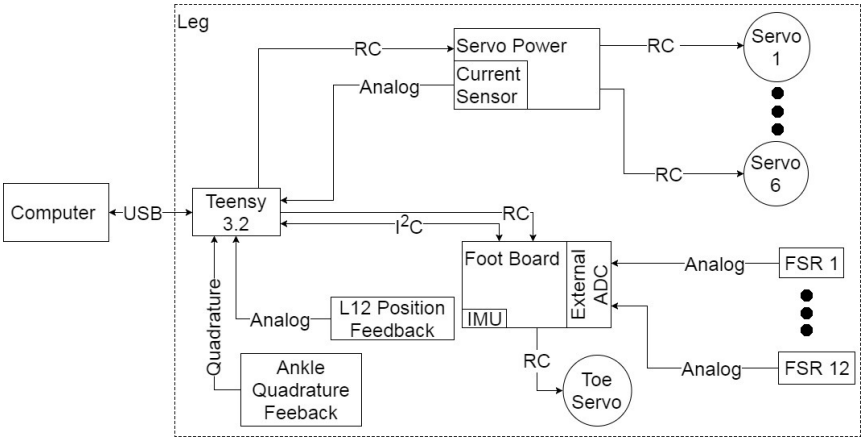


Figure 4.1: Overall block diagram of the Electrical system

The electrical system for this robotic leg was designed in accordance with the requirements of bipedal locomotion. Figure 4.1 shows the block diagram for the electrical system, in which connecting lines represent information flow and state either communication protocol or sensor. The electrical system is set up so that each leg is controlled through it's own Teensy 3.2 microcontroller, and will use USB communication to talk with a computer that will dictate desired joint position and process sensor information.

The Teensy 3.2 is a 32 bit ARM Cortex-M4 based development platform supported by the Arduino IDE, and is built around the Freescale MK20DX256VLH7 processor. A central board to host the Teensy was designed, there were also two daughter boards designed: one

for the sensing the current draw of the servos and the other for foot sensors and toe servo.

4.1 Teensy

The Teensy 3.2 was programmed using the Teensyduino add-on for the Arduino integrated development environment (IDE). Arduino was developed to abstract hardware from higher level functionality to allow for programing multiple types of microcontroller with the same code bases. This abstraction was developed with C++ objects and is defined as libraries. Arduino supports both C/C++ style programing and is commonly used in research for it's ease of use and abundance of libraries. It should be noted that even though Arduino provides a certain amount ease with the libraries it doesn't mean that they can be used blindly. One of the major problems that occur is multiple libraries using the same hardware. For example the library IntervalTimer uses interrupts to call a function at a precise timing interval and the Teensy can only run up to four of these objects simultaneously. Several other libraries uses IntervalTimer for this functionality (*i.e.*tone function, FreqCount, ShiftPWM, and NewPing just to name a few). So if someone was to program using four of these libraries and also a separate timed interrupt then there would be erratic behavior observed from the Teensy. So it is necessary to make sure that for the microcontroller being used there is enough hardware available for what the program is trying to execute.

4.1.1 Servo Library

Arduino has a Servo library which allows for controlling the servo with either degree or the pulse width of the on time, along with the ability to set the limits for the max and min pulse width values. The library uses a timers and interrupts to control the RC signal, the

Programmable Delay Block (PDB) is used as the timer for the Teensy 3.2. Because the library uses the PDB there is no interference with normal PWM functionality while also reduces any issues of multiple libraries using the same hardware.

After the max and min pulse width values were determined for each servo another library was created for controlling the L12 linear actuators. This library uses some of the functions available in Servo while adding the ability to write a distance and set the stroke and closed length for the linear actuators. This was done because Servo library can only write the angle or pulse width as mentioned and with the linear actuators, it is the distance that is being changed not an angle. The equation for calculating the pulse width in microseconds (PW) from a distance in millimeters (D) is given by Equation 4.1

$$PW = \frac{PW_{max} - PW_{min}}{L_S} \times (D - L_C) + PW_{min} \tag{4.1}$$

where

PW_{min} = minimum pulse width in microseconds

PW_{max} = maximum pulse width in microseconds

L_C = closed length of actuator in millimeters

L_S = stroke length of actuator in millimeters

For the three hip servo motors it was decided to calculate the pulse width from the angle outside of the Servo library. This was done for two reasons, first the Servo library only allows an angle between 0° and 180° which differs from the kinematic scheme. Second the Servo library assumes that the center angle value is the middle value of the pulse width

range, with the mounting of the potentiometer feedback for the harmonic drives there is no guarantee that this is the case. With this in mind a hip library was created to keep track of angles and determine the pulse width based on Equation 4.2

$$PW = \frac{PW_{range}}{\theta_{range}} \times \theta + PW_{center} \tag{4.2}$$

where

PW_{range} = range of the pulse width in microseconds

θ_{range} = angle rage of the servo in degrees

PW_{center} = center pulse width value in microseconds

θ = angle of harmonic drive in degrees

The slope of this linear equation is determined by the servos pulse wide and angle ranges before being modified and connected to the harmonic drives, because of how the servo interprets pulse width to the potentiometers reading. For the hip pitch harmonic drive the slope is negative because of the potentiometer's orientation in relation to the output shaft is flipped compared to the unmodified servo. Were as for the hip yaw and roll harmonic drive's slope needs to be multiplied by the gear ration that connects the output shaft of the harmonic drive to the shaft of the potentiometer.

4.1.2 Quadrature Decoding

The Teensy has two hardware quadrature decoders with some extra functionality that can be used to keep track of angular position. As can be seen from the block diagram in Figure 4.2 there are two extra functionalities to the Quadrature decoder, a filter and polarity selection

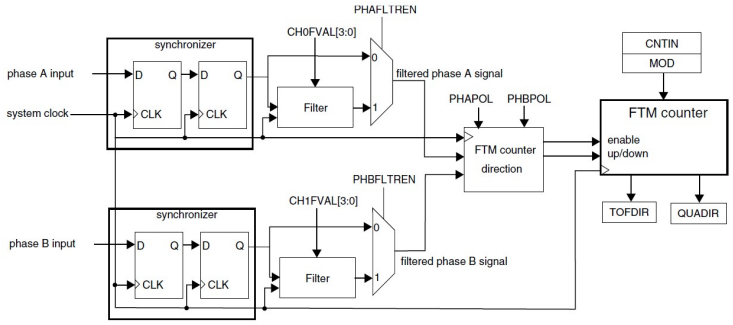
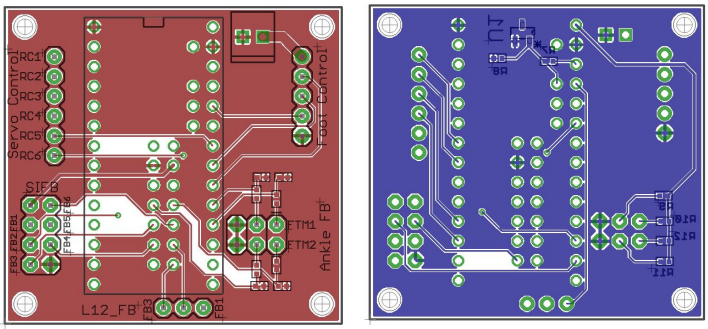


Figure 4.2: Block Diagram of Teensy 3.2 hardware Quadrature decoder

for each channel. The filter can aid in reducing error from noise on the lines, where the polarity selection can aid in making sure positive rotation is in the correct direction.

4.2 Microcontroller Carrier Board



(a) Top View (b) BottomView

Figure 4.3: Top (a) and Bottom (b) view of the carrier board

This central board was created to break out the Teensy pin's to the appropriate connectors, see Figure B.3 in Appendix B for schematic. The Teensy 3.2 is equipped with 21 pins that are multiplexed into two ADCs. These ADCs come with a lot of functionality including but not limited to external referencing, hardware averaging, Programmable Gain Amplifier (PGA) up to x64 gain, selectable speed and resolution. The microcontroller board was set

up with an external voltage reference of $2.048V$ which with a twelve bit conversion gives a $0.5mV$ step voltage for the ADC.

4.3 Current Sensing Board

	Stall Current@ 6V (mA)	Shunt Resistor (Ω)
Power HD-1501MG	2500	$0.04 \pm 1\%$
L12 Linear Actuators	460	$0.2 \pm 1\%$

Table 4.1: Student Database

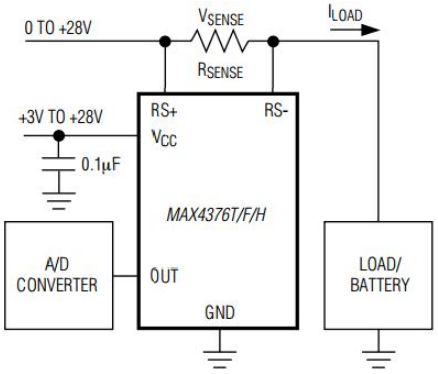


Figure 4.4: 2-D scatterplot of the Student Database

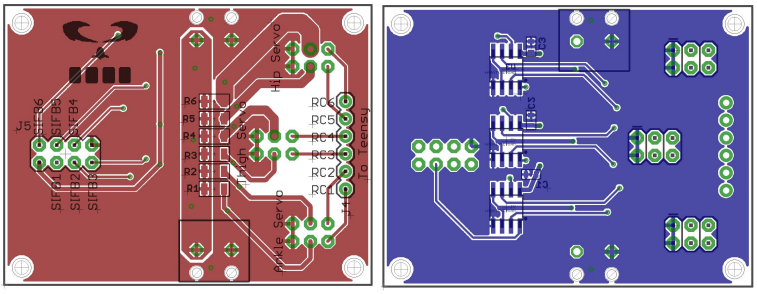
The current sensing board is equipped with three MAX4377 dual current sensing Integrated Circuit (IC) to determine motor torque. A schematic has been provided in Figure 4.4 in Appendix B. This board was only design to read six of the servo motors current, so at this time the toe servo motor's current is not being monitored. As shown in Figure 4.4 [1], the MAX4377 uses high side current sensing where a shunt resister (R_{SENSE}) is placed in series between the power supply and load. The output voltage(V_{OUT}) is determined by multiplying the IC's gain (G) by the shunt resistor and the load current (I_{LOAD}), shown in equation 4.3. This was then solved for the load current and is given in equation 4.4.

$$V_{OUT} = G * R_{SENSE} * I_{LOAD} \tag{4.3}$$

[1] The figure shows the MAX4376 which is the single current sensing version of this family of ICs

$$I_{LOAD} = \frac{V_{OUT}}{G * R_{SENSE}} \tag{4.4}$$

Ohm's Law was used for determining the shunt resister value by taking the full scale sense voltage ($V_{SENSE} = 100mV$) and the servo motor's stall current, this is given in Table 4.1. With the gain of the current sensor to be 20, the full scale output voltage is $2V$. Output of the MAX4377 is feed into the Teensy's internal ADC.



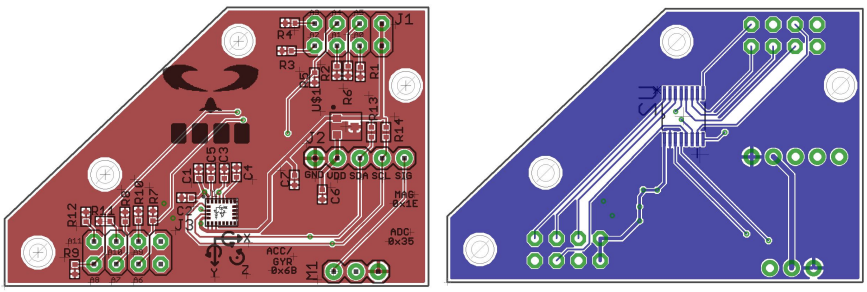
(a) Top View (b) BottomView

Figure 4.5: Top (a) and Bottom (b) view of the current sense board

4.4 Foot Board

The foot board is equipped with the LSM9DS1 Inertial Measurement Unit (IMU), and the MAX11611 external Analog to Digital Converter (ADC). A schematic has been provided in Figure B.1 in Appendix B. Both the IMU and MAX11611 communicate over Inter-Integrated Circuit (I^2C) to the Teensy and are powered by the on board voltage regulator. The IMU is comprised of a gyroscope, accelerometer, and magnetometer. It is used to help determine orientation and acceleration of foot during the swing action. The ADC is used to measure forces from the network of twelve Force Sensing Resistors (FSR). It was chosen to use an external ADC for two reasons, it would reduce the number of pins required by the Teensy and would eliminate any noise that would have been picked up from the length of

wire that would be needed to travel from the foot to the thigh. The FSR network is used to determine the Center of Pressure (CoP) of the foot for the stance leg during SSP. The layout of the FSR can be as shown in Figure 3.8 (c). The sensor data for each of these sensors will alternate between left and right foot during the walking cycle based upon which role that leg is preforming, i.e. swing or stance leg. Due to the high number of FSRs used, an external ADC was added to the foot board.



(a) Top View (b) BottomView

Figure 4.6: Top (a) and Bottom (b) view of the foot board

In this project the first step was to program individual functionality in order to simplify debugging of both hardware and software. The foot board functionality was first tested and uses preexisting libraries for the IMU and external ADC. For interfacing with the IMU, SparkFunLSM9DS1 library was used and provided functionality necessary. While the external ADC were controlled using the library created in the project [**?**]. After functionality was confirmed, attention was moved to controlling the servos.

Chapter 5

Forward & Inverse Kinematics

In this chapter the overall leg kinematics are discussed first in detail. In this particular design linear motion is translated to joint rotational motion using linear actuators in the knee and ankle. The kinematic relationship of the joint angle and linear actuator's length for the knee and the ankle's roll and pitch angles to the two linear actuator's lengths will be discussed.

5.1 Leg

Figure 5.1 show the kinematic description of the bipedal leg design.

Where:

l_i = length of link i in millimeters

$\theta_1, \theta_2, \theta_3$ = hip yaw, roll and pitch angle respectively in degrees

θ_4 = knee angle in degree

θ_5, θ_6 = ankle pitch and roll angle respectively in degrees

(X_i, Y_i, Z_i) = local reference frame no. i [1]

[1] The axis Y_i isn't labeled in Figure 5.1 to reduce the amount of clutter in the diagram

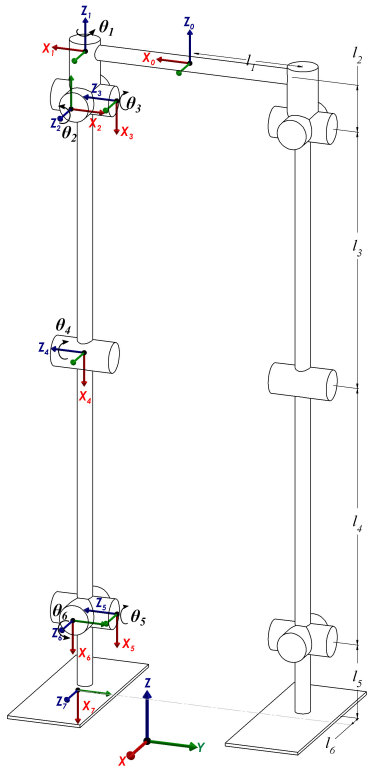


Figure 5.1: Kinematic description of robot leg

DH parameter	Joint					
	1	**2**	**3**	**4**	**5**	**6**
θ_i	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6
d_i	$-l_2$	0	0	0	0	l_6
a_i	l_1	0	0	l_3	l_4	l_5
α_i	0	$\pi/2$	$\pi/2$	0	0	$-\pi/2$

Table 5.1: DH Table parameters

The local frames (X_i, Y_i, Z_i) are assigned to each joint according to the DenavitHartenberg (DH) convention[11]. Table 5.1 shows the DH parameters where θ_i is the angle between the X_{i-1} and X_i axes as measured about the Z_{i-1} axis; d_i is the distance from the X_{i-1} to the X_i axis as measured along the Z_i axis; a_i is the distance from the Z_{i-1} to Z_i axis measured along the X_{i-1} axis; and α_i is the angle between the Z_{i-1} and Z_i axes measured about the X_{i-1} axis. The angles are assumed positive, counterclockwise about the rotation axis.

A general transformation from one frame to another ($^{i-1}\mathrm{T}_i$) can then be determined by

multiplying rotation and translation matrices together as shown in Equation 5.1.

$$^{i-1}\mathrm{T}_i = A_i = Rot(z,\theta_i) \times Trans(0,0,d_i) \times Trans(a_i,0,0) \times Rot(x,\alpha_i)$$

$$A_i = \begin{bmatrix} C\theta_i & -S\theta_i \cdot C\alpha_i & S\theta_i \cdot S\alpha_i & a_i \cdot C\theta_i \\ S\theta_i & C\theta_i \cdot C\alpha_i & -C\theta_i \cdot S\alpha_i & a_i \cdot S\theta_i \\ 0 & S\alpha_1 & C\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.1}$$

where:

$$C\theta_i \equiv \cos(\theta_i), \quad C\alpha_i \equiv \cos(\alpha_i), \quad S\theta_i \equiv \sin(\theta_i), \quad S\alpha_i \equiv \sin(\alpha_i)$$

To obtain a transformation for the entire leg ($^0\mathrm{T}_6$) the individual transformation need to be post multiplied as shown in Equation 5.2.

$$^0\mathrm{T}_6 = \prod_{i=1}^{7} A_i = A_1A_2A_3A_4A_5A_6A_7 = \begin{bmatrix} \bar{n} & \bar{o} & \bar{a} & \bar{p} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.2}$$

where A_i^{i1} is a general link transformation matrix, relating the ith coordinate frame to the $(i1)$th coordinate frame, and $\begin{bmatrix} \bar{n} & \bar{o} & \bar{a} & \bar{p} \end{bmatrix}$ represents the normal vector, the sliding vector, the approach vector, and the position vector of the hand, respectively[24].

In oder to simplify the process of finding a closed form solution for the inverse kinematics this paper will only look at planar motion along a anatomical plane. Figure 5.2 shows the inverse kinematics of the Sagittal plane. Note that Δx and Δy are the differential step positions, while (X_1, Z_1), (X_5, Z_5) and (X_7, Z_7) denote the position for the waist, ankle and foot, respectively.

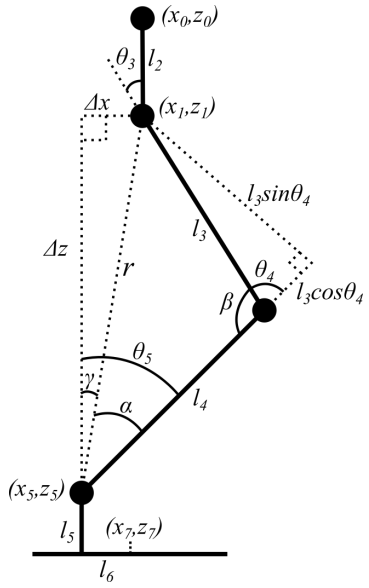


Figure 5.2: Right leg inverse kinematics in the Sagittal plane

The approach that this paper takes to determine the inverse kinematics of the right leg in the Sagittal plane consists of finding the joint angle for the knee (θ_4), given the global position of the hip and ankle. Which assumes that the hip and ankle trajectories in the Sagittal plane are known.

By applying the law of cosine, which relates the length of a triangle sides to one of it's angles [25], to the triangle bounded by l_3 and l_4 in Figure 5.2 yields

$$r^2 = l_3^2 + l_4^2 - 2l_3 l_4 \cos(\pi - \theta_4) \Rightarrow r^2 = l_3^2 + l_4^2 + 2l_3 l_4 \cos(\theta_4)$$
$$\cos(\theta_4) = \frac{r^2 - l_3^2 - l_4^2}{2l_3 l_4} \tag{5.3}$$

Equation 5.3 can be used to obtain $\sin(\theta_4)$ from Pythagorean theorem

$$\sin(\theta_4) = \sqrt{1 - \cos^2(\theta_4)}$$
$$\sin(\theta_4) = \sqrt{1 - \frac{r^2 - l_3^2 - l_4^2}{2l_3 l_4}} \tag{5.4}$$

Equation 5.3 & 5.4 can now be used with the trigonometric identity atan2 to produce a solution for θ_4

$$\theta_4 = \text{atan2}\Big(\sin(\theta_4), \cos(\theta_4)\Big)$$
$$\theta_4 = \text{atan2}\left(\sqrt{1-\left(\frac{r^2-l_3^2-l_4^2}{2l_3l_4}\right)^2}, \frac{r^2-l_3^2-l_4^2}{2l_3l_4}\right) \tag{5.5}$$

where:

$$r^2 = (x_5-x_1)^2 + (z_5-z_1)^2 \tag{5.6}$$

The angle for the ankle (θ_5) can be obtained by combining the angles α & γ from Figure 5.2

$$\gamma = \text{atan2}\big((x_5-x_1),(z_5-z_1)\big), \quad \alpha = \text{atan2}\big(l_3\sin(\theta_4), l_4 + l_3\cos(\theta_4)\big)$$
$$\theta_5 = \gamma + \alpha = \text{atan2}\big((x_5-x_1),(z_5-z_1)\big) + \text{atan2}\big(l_3\sin(\theta_4), l_4 + l_3\cos(\theta_4)\big) \tag{5.7}$$

With the ankle and knee angles both determined the next step would be to determine the hip angle. By constraining the hip to maintain a vertical position it is possible to use geometry to solve for the hip angle which yields

$$\left.\begin{aligned}\beta &= \pi - \theta_4 \\ \theta_3 + \theta_5 + \beta &= \pi\end{aligned}\right\} \theta_3 + \theta_5 + \pi - \theta_4 = \pi \rightarrow \theta_3 = \theta_4 - \theta_5 \tag{5.8}$$

Now that inverse kinematics have been solved it is nessacy to come up with a trajectory for the foot during the swing phase, which was done by utilizing a polynomial trajectory algorithm as present in [26].

5.2 Knee

Because a linear actuator is used in controlling the knee it is necessary to determine the mathematical relationship between the knee angle and linear actuator position. Figure 5.3 show the kinematic description of the knee along with variables to be used in this discussion of forward and inverse kinematics.

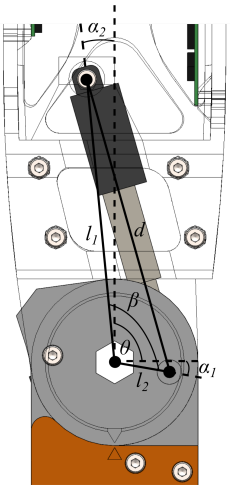


Figure 5.3: kinematic description of Knee

The law of cosines is used to relate the joint angle (θ) and linear actuator position (d), which yields

$$d^2 = l_1^2 + l_2^2 - 2l_1 l_2 \cos\beta \tag{5.9}$$

Now this only describes the angle β

$$\beta = \theta + \alpha_1 + \alpha_2 \tag{5.10}$$

The inverse kinematics can be solved by substituting equation 5.10 into equation 5.9

which yields

$$d = \pm \sqrt{l_1^2 + l_2^2 - 2l_1 l_2 \cos(\theta + \alpha_1 + \alpha_2)} \tag{5.11}$$

Because the distance cannot be negative this result is ignored. By solving for θ the forward kinematics can be determine and shown here

$$\theta = \cos^{-1}\left(\frac{-d^2 + l_1^2 + l_2^2}{2l_1 l_2}\right) - \alpha_1 - \alpha_2 \tag{5.12}$$

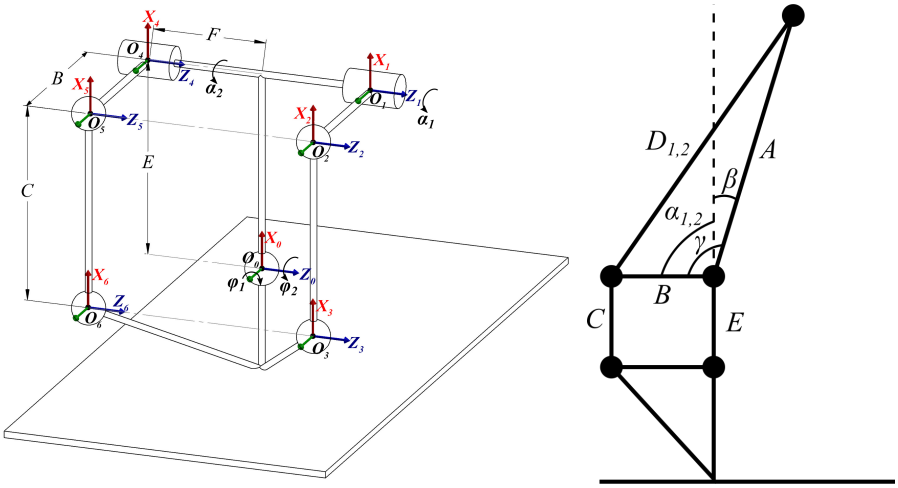
Both Equation 5.11 & 5.12 don't explicitly solve for θ_4 but rather solves for the angle θ presented in Figure 5.3. However θ and θ can be related by a phase shift as show in the equation below.

$$\theta = \frac{\pi}{2} - \theta_4 \tag{5.13}$$

5.3 Ankle

Complexity of the ankle poses some problems due to coupling between the joint angles and position of the linear actuators. One author [27] proposes a leg design using parallel linkages with two servo motors, in which he presents a method of solving the ankle angles to each servo angle. By utilizing this method to decouple the ankle joints angles to an angle linked to each linear actuator the inverse kinematics can be solved.

Figure 5.4a shows the kinematic description of the Ankle, in which ϕ_1 and ϕ_2 represent the ankle's roll and pitch angles respectively and α_1 and α_2 represent the decoupled angles. The rotation matrix R_0^3 which transforms the base reference frame O_0 to O_3 reference frame can be calculated based on the foot rotation about z_0 with angle α_2 and about y_0 with angle



(a) (b)

Figure 5.4: Kinematic description of Ankle shown in 3 dimension (a) and in the Sagittal plane (b)

α_1. The successive rotations give the rotation matrix:

$$R_0^3 = \begin{bmatrix} \cos(\phi_1)\cos(\phi_2) & -\cos(\phi_1)\sin(\phi_2) & \sin(\phi_1) \\ \sin(\phi_2) & \cos(\phi_2) & 0 \\ -\sin(\phi_1)\cos(\phi_2) & \sin(\phi_1)\sin(\phi_2) & \cos(\phi_1) \end{bmatrix} \tag{5.14}$$

The displacement vector $\overline{O_0O_3}$ can be expressed as

$$\overline{O_0O_3} = R_0^3 \cdot \begin{bmatrix} 0 \\ C \\ F \end{bmatrix} \tag{5.15}$$

The O_2 frame in relation to O_1 frame can be given by the following rotation matrix

$$R_1^2 = \begin{bmatrix} \cos(\alpha_1) & -\sin(\alpha_1) & 0 \\ \sin(\alpha_1) & \cos(\alpha_1) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5.16}$$

Therefore the displacement vector $\overline{O_0O_2}$ can be expressed by

$$\overline{O_0O_2} = \begin{bmatrix} E \\ 0 \\ F \end{bmatrix} + R_1^2 \cdot \begin{bmatrix} 0 \\ B \\ 0 \end{bmatrix} \tag{5.17}$$

Now the normal distance between frame O_2 and O_3 is the same as the length of bar C:

$$\left\| \overline{O_0O_2} - \overline{O_0O_3} \right\| = C \tag{5.18}$$

The above equation can be simplified to give a function that relates the roll and pitch to the either α_1 or α_2 as shown below.

$$\begin{aligned} f_1(\alpha_1, \phi_1, \phi_2) = 0 = {} & 2 \cdot B^2 - 2 \cdot F^2 \cdot \cos(\phi_1) - C^2 + E^2 + 2 \cdot F^2 - 2 \cdot E \cdot F \cdot \sin(\phi_1) \\ & - 2 \cdot B^2 \cdot \cos(\alpha_1) \cdot \cos(\phi_2) - 2 \cdot B \cdot E \cdot \sin(\alpha_1) - 2 \cdot B \cdot F \cdot \sin(\phi_1) \cdot \sin(\phi_2) \\ & - 2 \cdot B^2 \cdot \cos(\phi_1) \cdot \sin(\alpha_1) \cdot \sin(\phi_2) + 2 \cdot B \cdot E \cdot \cos(\phi_1) \cdot \sin(\phi_2) \\ & + 2 \cdot B \cdot F \cdot \sin(\alpha_1) \cdot \sin(\phi_1) \end{aligned} \tag{5.19}$$

For the opposite side the same reasoning can be applied and yields the equation

$$\begin{aligned} f_2(\alpha_2, \phi_1, \phi_2) = 0 = {} & 2 \cdot B^2 - 2 \cdot F^2 \cdot \cos(\phi_1) - C^2 + E^2 + 2 \cdot F^2 + 2 \cdot E \cdot F \cdot \sin(\phi_1) \\ & - 2 \cdot B^2 \cdot \cos(\alpha_2) \cdot \cos(\phi_2) - 2 \cdot B \cdot E \cdot \sin(\alpha_2) + 2 \cdot B \cdot F \cdot \sin(\phi_1) \cdot \sin(\phi_2) \\ & - 2 \cdot B^2 \cdot \cos(\phi_1) \cdot \sin(\alpha_2) \cdot \sin(\phi_2) + 2 \cdot B \cdot E \cdot \cos(\phi_1) \cdot \sin(\phi_2) \\ & - 2 \cdot B \cdot F \cdot \sin(\alpha_2) \cdot \sin(\phi_1) \end{aligned} \tag{5.20}$$

To determine the inverse kinematics, Equation 5.19 must be solved for α_1. Continuing

with the method presented here [27] consider the following

$$a_1 \cdot \sin(\alpha_1) + a_2 \cdot \cos(\alpha_1) + a3 = 0 \tag{5.21}$$

where:

$$a_1 = 2 \cdot B \cdot F \cdot \sin(\phi_1) - 2 \cdot B \cdot E - 2 \cdot B^2 \cdot \cos(\phi_1) \cdot \sin(\phi_2)$$

$$a_2 = -2 \cdot B^2 \cdot \cos(\phi_2)$$

$$a_3 = 2 \cdot B^2 - 2 \cdot F^2 \cdot \cos(\phi_1) - C^2 + E^2 + 2 \cdot F^2 - 2 \cdot E \cdot F \cdot \sin(\phi_1) - 2 \cdot B \cdot F \cdot \sin(\phi_1) \cdot \sin(\phi_2) + 2 \cdot B \cdot E \cdot \cos(\phi_1) \cdot \sin(\phi_2)$$

with the following change of variables:

$$\sin(\alpha_1) = \frac{2t}{1+t^2} \cos(\alpha_1) = \frac{1-t^2}{1+t^2}$$

where

$$t = \tan(\frac{\alpha_1}{2})$$

Then Equation 5.21 can be rewritten as:

$$(a_3 - a_2) \cdot t^2 + 2 \cdot a_1 \cdot t + (a_3 + a_2) = 0 \tag{5.22}$$

Which is a simple quadratic equation, solving for t and substituting back give our final result for α_1

$$\alpha_1 = 2 \cdot \tan^{-1} \left(\frac{-a_1 \pm \sqrt{a_1^2 + a_2^2 - a_3^2}}{a_3 - a_2} \right) \tag{5.23}$$

Only one of these solutions for the above equation is physically possible. With the same

reasoning α_2 can be determine and is given by

$$\alpha_2 = 2 \cdot \tan^{-1}\left(\frac{-b_1 \pm \sqrt{b_1^2 + b_2^2 - b_3^2}}{b_3 - b_2}\right) \tag{5.24}$$

where:

$$b_1 = -2 \cdot B \cdot E - 2 \cdot B \cdot F \cdot \sin(\phi_1) - 2 \cdot B^2 \cdot \cos(\phi_1) \cdot \sin(\phi_2)$$

$$b_2 = -2 \cdot B^2 \cdot \cos(\phi_2)$$

$$b_3 = 2 \cdot B^2 - 2 \cdot F^2 \cdot \cos(\phi_1) - C^2 + E^2 + 2 \cdot F^2 + 2 \cdot E \cdot F \cdot \sin(\phi_1) + 2 \cdot B \cdot F \cdot \sin(\phi_1) \cdot \sin(\phi_2) + 2 \cdot B \cdot E \cdot \cos(\phi_1) \cdot \sin(\phi_2)$$

Now that the ankle's roll and pitch angles have been decoupled we can determine the linear actuator's length. Figure 5.4b shows the kinematic description of the ankle in the Sagittal plane, because the joint angles have been decoupled the following steps can be directly applied to both sides(hence $D_{1,2}$ and $\alpha_1 1,2$). By using the law of cosines the follow equation can be deduced from Figure 5.4b

$$D_{1,2}^2 = A^2 + B^2 - 2 \cdot A \cdot B \cos(\gamma) \tag{5.25}$$

Given that γ is the summation of $\alpha_{1,2}$ and the constant β, then the above equation can be rewritten and solved for our acutator length ($D_{1,2}$). Which is given by

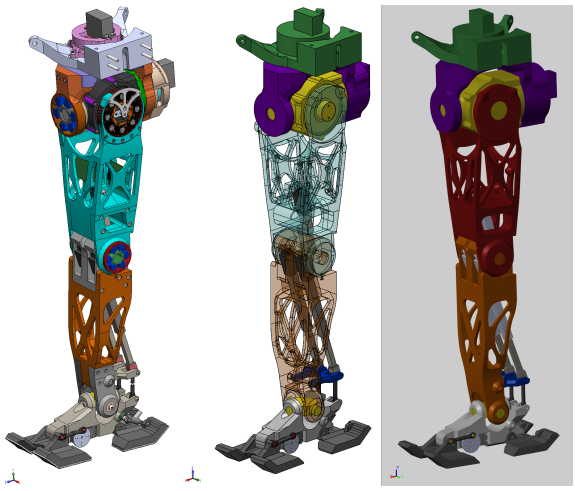
$$D_{1,2} = \sqrt{A^2 + B^2 - 2 \cdot A \cdot B \cos(\beta + \alpha_{1,2})} \tag{5.26}$$

Chapter 6

Simulation & Modeling

Simulations are often used by researches to test various walking algorithms in bipedal robotics. Simulation can also be use to validate a design before finished construction, however these are normally based off of some assumptions. Either a simplified model assuming to be close enough to actual behavior of the mechanism or an approximation of some constant associated with the system. This is done with good reason, it's improbably to consider every variable in a complex equation (such as those seen in bipedal locomotion) or to expect an approximation of a constant to come out to be exactly the right value (Like estimating parameters of a 3D printed design). For that reason it is proposed to consider a design strategy where both the model and the design can be updated as needed. As discussed earlier 3D printing allows for designs to be quickly developed and tested, the following chapter will discuss the ability to do this in simulation as well.

With many physical modeling simulation software it is often impractical to do the dynamic analyses on the detailed design. For this reason the bipedal leg design shown in Figure 6.1a was simplified into the mock-up design shown in Figure 6.1b. In which the mounting hardware, gear bearings and most of the servo motors where removed from the



(a) (b) (c)

Figure 6.1: (a) Detailed Solidworks Design, (b) Mock-up Solidworks Design, (c) Mock-up Simulink Mechanics Explorer

detailed design. Along with this split link members were recombined into a solid component, for example the shin was made to be one peace instead of four. After the mock-up design was completed it was exported into Simulink, seen in Figure 6.2 using Simscape Multibody Link add-on for Solidworks. The exporter uses Simscape Multibody library elements to create the simulation model to see dynamic behavior of a system. The exporter also generates STL files so that the system can be viewed in motion as seen in Figure 6.1c.

In order for the simulation to consider the dynamics of a system, link masses must be known. It is difficult to estimate the mass of a 3D printed part considering the amount of variables that can be adjusted.(i.e. shell thickness, infill, material type, etc...) Adding to that, having to take into account the mass for all the mechanical hardware and electrical components which can be found from data sheets but there is batch variation to consider as well. It was simpler to weigh each link once construct and add this mass property in the mock-up Solidworks model. Solidworks can compute all the moments of inertia for a

known mass along with computing a bodies Center of Mass (CoM). These values are also exported into the Simulink model. This allows for real world dependencies to be adjusted in the simulation model to give more accurate results.

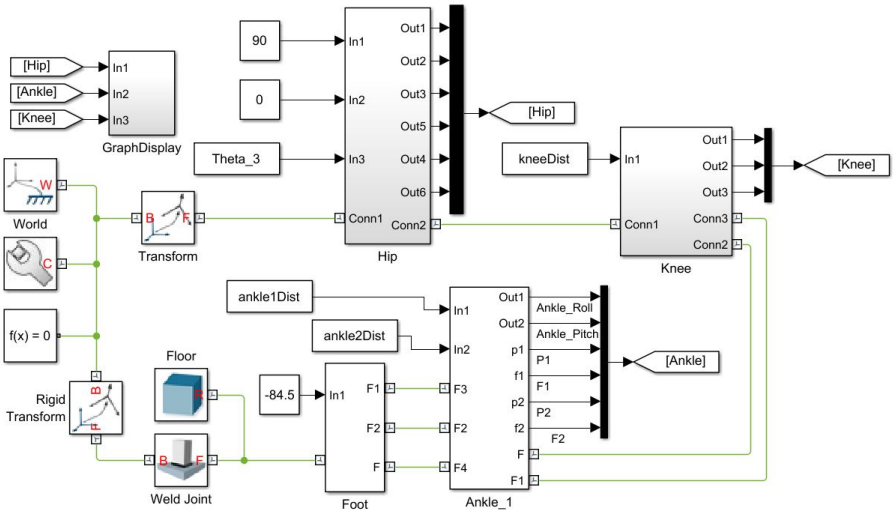


Figure 6.2: Simulink Block Diagram for leg design

As can be seen in Figure 6.2, Inputs where provided for the linear actuator distances along with the joint angles for the hip and toe. As discussed in Section 5.1 the inverse kinematics were solved for the Sagittal plane and as such the toe joint along with the hip roll and yaw joints were kept at a fixed angle. A Matlab script was written to solve for the inverse kinematics for each joint along with the corresponding linear actuator lengths based upon the foot trajectory during the swing leg or the hip location for the stance leg.

The foot subsystem block from Figure 6.2 is shown in Figure 6.3 as an example of how the subsystems look, the rest of the subsystem blocks can be seen in Appendix A. The kinematic loop for the toe linkage system can be seen in the figure below, as the collection of Revolute Joints. There is also two other subsystem block that are worth mentioning,

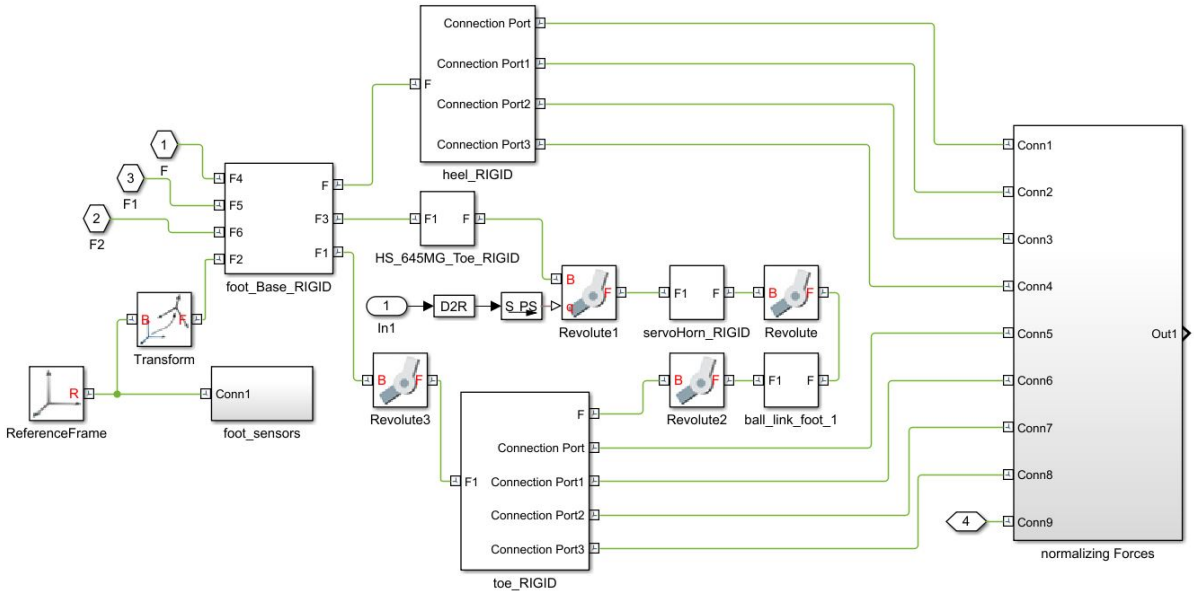


Figure 6.3: Simulink Block Diagram for Foot Block

the *foot_sensor* and *normalizing Forces* blocks. These two blocks are used as a metric of comparison to the real world bipedal leg design.

The *foot_sensor* block was created as an attempt to mimic an Inertia Measurement Unit (IMU) by utilizing a bushing joint element and taking the derivatives of the translational and rotation elements as shown in Figure 6.4. There is also a six Degree of Freedom (DOF) joint which relates the Cartesian position of the foot with respect to the global frame in order to see how well the simulation follows the foot trajectory.

The *normalizing Forces* block from Figure 6.2 is used to measure the Zero Moment Point (ZMP) by utilizing the Simscape Multibody Contact Forces Library[28]. This library simulates the contact force between two solid bodies in either 2D or 3D space. The content of the subsystem block can be seen in Appendix A with the rest of the subsystem blocks.

In bipedal robotics there has been a drive for more anthropomorphic design. So for

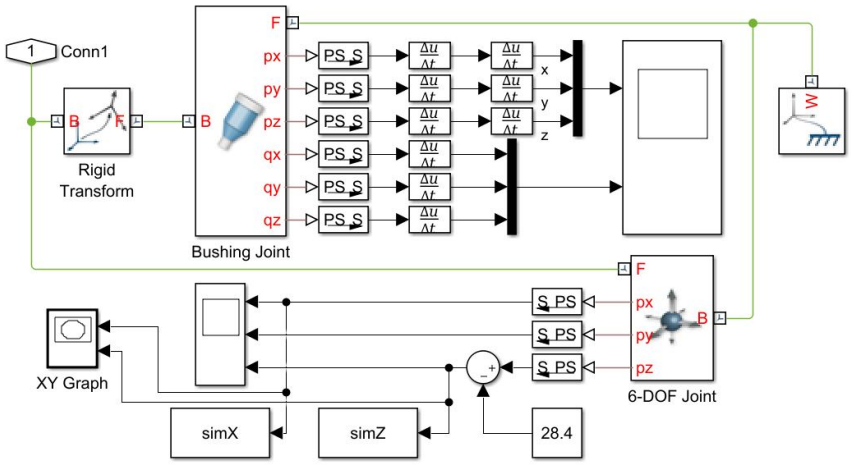


Figure 6.4: Simulink Block Diagram for foot_sensors Block

a metric of comparison anthropometric data[29] was taken along with Body Mass Index (BMI) were used to determine link mass and length properties. A Matlab GUI was created so that this information could be seen based upon the bipedal robots expected height, shown in Figure 6.5

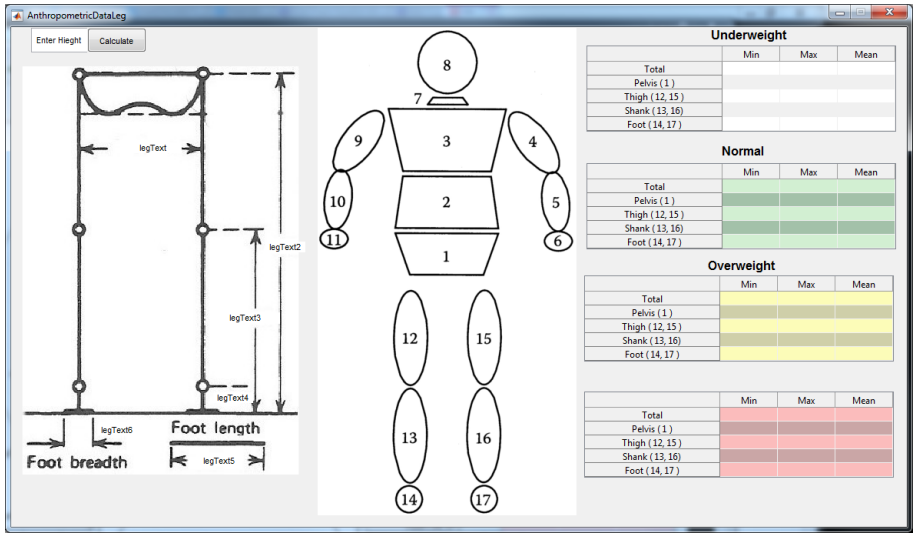


Figure 6.5: Anthropometric Leg Data GUI

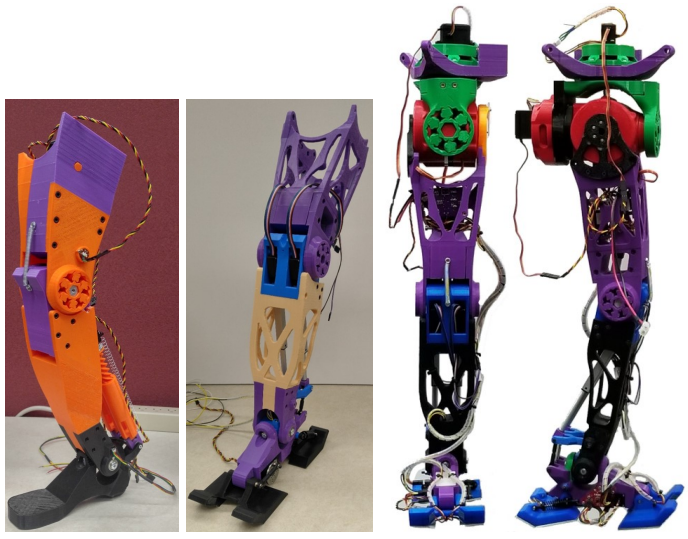
Chapter 7

Results

The first (Figure 7.1a) and second (Figure 7.1b) iteration of the lower leg design. Table 7.1 shows the amount of play in each joint for the two iterations. As can be seen the second iteration design provided more stability than the first.

Table 7.1: Amount of play in joints for design

	Ankle		Knee
	Roll	Pitch	
Design 1	10°	2°	10°
Design 2	5°	0°	5°



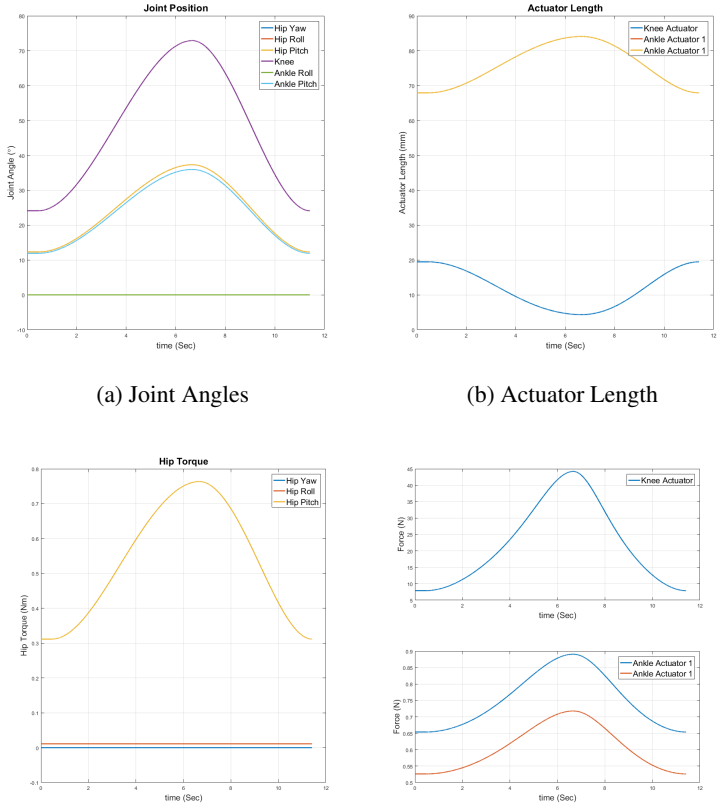
(a) (b) (c)

Figure 7.1: (a) first, (b) second, and (c) complete designs of the right leg of a biped robot

Figure 7.1c shows the completed leg assembly. Preliminary testing showed that the

hip roll drive train did not have the required torque need for a walking gate. This was determined by having the foot swing outward away from the body via the hip roll joint. This caused the motor to stall before reaching it's desired angle. Although the hip yaw motor was able to move the leg, it is unlikely that it would be able to swing an opposing leg as would be required in a walking gate.

The knee motor was initially tested by placing the leg in a static stance position. It was determined that the knee was unable to hold the full weight of the upper leg once it past approximately 30°.



(a) Joint Angles (b) Actuator Length

(c) Joint Torques (d) Actuator Force

Figure 7.2: Simulation Results

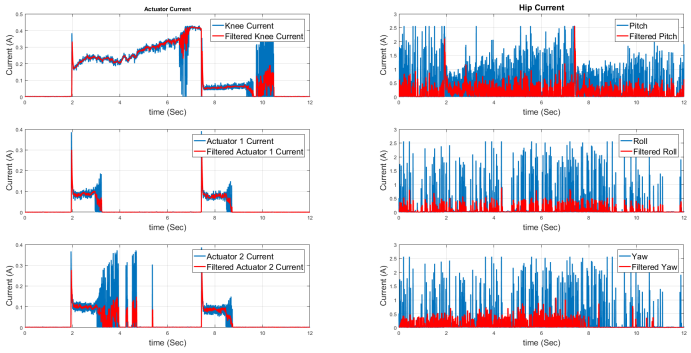
To test the simulation model the foot and hip locations were specified. Using the inverse kinematics shown in Section 5.1, the joint angles were then determined. From there a time was specified for the motion to complete in and a joint trajectory was calculated this can be seen in Figure 7.2a. After the trajectory was calculated the linear actuators length for the knee and ankle was determined and is shown in Figure 7.2b.

Because the Solidworks model had the masses of each part the joint torques and linear actuator forces was determined through the physics engine in Simscape. The torques and forces are shown in Figure 7.2c and Figure 7.2d respectively.

As with simulation, the foot and hip positions were specified for the hardware test. After the inverse kinematics were solved the joint angles were then sent to the microcontroller. As can be seen in Figure 7.3c, the joint angles from the ankle encoders are slightly off when compared to simulation. This could be due to the filter implemented on the microcontroller carrier board.

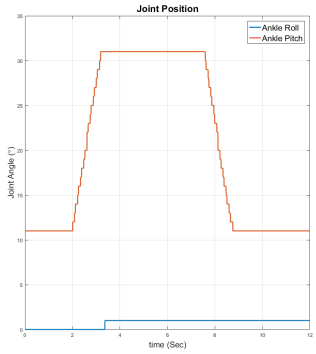
Figure 7.3a shows the current readings for the knee (top) and ankle (bottom two). As stated above the linear actuator for the knee did not have enough force to handle the joint torques required. This can be seen from Figure 7.3a top, were the current climbs up to it's stall value before the knee changes direction of motion. The bottom two plots in Figure 7.3a present the ankle's linear actuators. Now a direct comparison between the simulation force and hardware current readings for the ankle actuators isn't possible, it can be seen that in both simulation and hardware there is a difference in values between actuator 1 and 2. Meaning that one of the linear actuators is providing more force.

As can be seen from Figure 7.3b, there is a lot of noise for the current readings of the

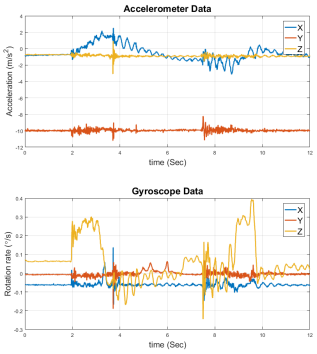


(a) Current reading for actuators

(b) Current reading for servos



(c) Anklke joint angles



(d) IMU data

Figure 7.3: Simulation Results

hip. A windowed filter was used on all of the current reading. While this cleaned up the linear actuator signals, the servo current readings still have a decent amount of noise. This is most likely due to the control circuitry not being properly tuned for the harmonic drives. The pitch motor does have a small amount of current draw at the beginning of motion(2sec) due to the large gear ratio of the harmonic drives.

Leg Section	**Actual Robot (mm)**	**Theory (mm)**
A	150	218
B	556	606
C	315.9	326
D	65.9	44.6
E	104	62.9
F	228.8	174

Table 7.2: Link lengths based on height

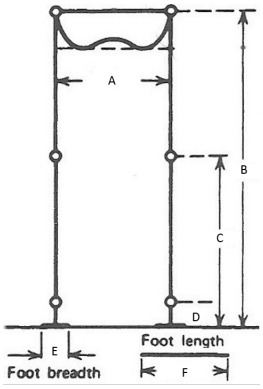


Figure 7.4: Link lengths based on height

Table 7.3: My caption

	Actual	Underweight	Normal	Overweit	Obese
Pelvis(1)	1383.44	2478.1	3534.3	4468.6	5281.1
Thigh(12, 15)	908.6	2092	2983.6	3772.4	4458.3
Shank(13, 16)	458.323	854.7	1219	1541.3	1821.5
Foot (14, 17)	290.493	255	363.7	459.9	543.5

Chapter 8

Conclusions & Future Work

This paper presents a design and implementation of a 3D printed seven DoF biped leg inspired by large scale humanoid robots. It has been shown that by utilizing 3D printing technology it is possible to have a complex design that are easily reproducible. Such designs include the 3-DoF hip comprised of harmonic drives driven by standard RC servos, the use of linear actuators in the knee and the coupled 2-DoF joint system of the ankle.

There are some down falls to the current design, as was explained the torque produced by the hip yaw and roll joints along with the knee is just simple not enough to produce any kind of walking gate. The knee's lack of torque can be improved by changing where it is mount with respects to either the shin or thigh. With the L12 linear actuators being so customizable this change in mechanical design would have little to no effect on the electrical system. The hip pitch joint does provide adequate torque but at a significant coast to speed, which also limits the ability to preform a successful walking gate. The hip over all would need some consideration before the design would be capable of a walking gate. Change the yaw and roll harmonic gears to that of the pitch harmonic gear would increase torque output and switching out the RC servos would handle the speed issue of the hip pitch joint. The RC servos would either be replaced with smart servos or a self

designed servo with BLDC motors or even stepper motors.

Even though this design has proven the concepts presented above there is still improvements that can be done. The first would be to replace the RC servo with either a smart servo or self designed servo system. It would be interesting to see the possibility of 3D printing other complex drive trains such as series elastic actuators.

Bibliography

[1] "Shop replicator 2x experimental 3d printer — makerbot," https://store.makerbot.com/printers/replicator2x/, (Accessed on 07/26/2017).

[2] "Mp select mini 3d printer v2, white - monoprice.com," https://www.monoprice.com/product?p_id=15365, (Accessed on 07/26/2017).

[3] "Ic - wikipedia," https://en.wikipedia.org/wiki/I%C2%B2C, (Accessed on 07/26/2017).

[4] E. Ackerman and E. Guizzo. (2016, Feb.) The next generation of boston dynamics' atlas robot is quiet, robust, and tether free. Automaton Robotics Humanoid Robotics. IEEE SPECTRUM. [Online]. Available: http://spectrum.ieee.org/automaton/robotics/humanoids/next-generation-of-boston-dynamics-atlas-robot

[5] S. Lohmeier, *Design and Realization of a Humanoid Robot for Fast and Autonomous Bipedal Locomotion*, 2010.

[6] Y. Ogura, H. Aikawa, K. Shimomura, H. Kondo, A. Morishima, H. O. Lim, and A. Takanishi, "Development of a new humanoid robot WABIAN-2," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2006, no. May, pp. 76–81, 2006.

[7] I. Ha, Y. Tamura, H. Asama, J. Han, and D. W. Hong, "Development of open humanoid platform DARwIn-OP," *SICE Annual Conference 2011*, pp. 2178–2181, 2011.

[8] C. Hernandez-Santos, E. Rodriguez-Leal, R. Soto, and J. Gordillo, "Kinematics and Dynamics of a New 16 DOF Humanoid Biped Robot with Active Toe Joint," *International Journal of Advanced Robotic Systems*, p. 1, 2012.

[9] D. Gouaillier, V. Hugel, P. Blazevic, C. Kilner, J. Monceaux, P. Lafourcade, B. Marnier, J. Serre, and B. Maisonnier, "Mechatronic design of NAO humanoid," *2009 IEEE International Conference on Robotics and Automation*, pp. 769–774, 2009.

[10] M. Lapeyre, S. N'Gyuen, A. Le Falher, and P.-Y. Oudeyer, "Rapid morphological exploration with the Poppy humanoid platform," *Humanoids 2014*, 2014. [Online]. Available: http://hal.inria.fr/hal-00861110

[11] C. Chevallereau, *Bipedal robots : modeling, design and walking synthesis*. London Hoboken, NJ: ISTE John Wiley & Sons, 2009.

[12] "Ninjaflex flexible 3d printing filament — ninjatek," https://ninjatek.com/products/filaments/ninjaflex/, (Accessed on 07/26/2017).

[13] "Cheetah flexible 3d printing filament — ninjatek," https://ninjatek.com/products/filaments/cheetah/, (Accessed on 07/26/2017).

[14] H. Ulbrich, T. Buschmann, and S. Lohmeier, "Development of the Humanoid Robot LOLA," *Applied Mechanics and Materials*, vol. 5-6, pp. 529–540, 2006.

[15] D. Kuehn, F. Grimminger, F. Beinersdorf, F. Bernhard, A. Burchardt, M. Schilling, M. Simnofske, T. Stark, M. Zenzes, and F. Kirchner, "Additional DOFs and sensors for bio-inspired locomotion: Towards active spine, ankle joints, and feet for a quadruped robot," *2011 IEEE International Conference on Robotics and Biomimetics, ROBIO 2011*, pp. 2780–2786, 2011.

[16] K. Fondahl, D. Kuehn, F. Beinersdorf, F. Bernhard, F. Grimminger, M. Schilling, T. Stark, and F. Kirchner, "An adaptive sensor foot for a bipedal and quadrupedal robot," *Proceedings of the IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics*, pp. 270–275, 2012.

[17] D. Kuehn, F. Bernhard, A. Burchardt, M. Schilling, T. Stark, M. Zenzes, and F. Kirchner, "Distributed computation in a quadrupedal robotic system," *International Journal of Advanced Robotic Systems*, vol. 11, no. 1, 2014.

[18] "Gear bearing by emmett - thingiverse," https://www.thingiverse.com/thing:53451, (Accessed on 07/26/2017).

[19] "Harmonic drive by jdow - thingiverse," https://www.thingiverse.com/thing:20177, (Accessed on 07/26/2017).

[20] "Harmonic drive by bartdring - thingiverse," https://www.thingiverse.com/thing:1966551, (Accessed on 07/26/2017).

[21] R. Sellaouti, O. Stasse, S. Kajita, K. Yokoi, and A. Kheddar, "Faster and smoother walking of humanoid hrp-2 with passive toe joints," pp. 4909–4914, 2006.

[22] H. F. N. Al-Shuka, F. Allmendinger, B. Corves, and W.-H. Zhu, "Modeling, stability and walking pattern generators of biped robots: a review," *Robotica*, no. December 2013, pp. 1–28, 2013. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-84889068004{&}partnerID=tZOtx3y1

[23] T. Y. K. S. H. Inoue,, "High-speed pressure sensor grid for humanoid robot foot," *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, 2005.

[24] R. Mittal and I. Nagrath, *Robotics and control*. Tata McGraw-Hill, 2003.

[25] "Law of cosines - wikipedia," https://en.wikipedia.org/wiki/Law_of_cosines, (Accessed on 07/26/2017).

[26] M. Ramirez, E. Cuevas, D. Zaldivar, M. Pérez-Cisneros, and M. Ramírez-Ortegón, "Polynomial trajectory algorithm for a biped robot," *International Journal of Robotics and Automation*, vol. 25, no. 4, pp. 294–303, 2010.

[27] M. L. A.T., "Kinematical modeling and optimal design of a biped robot joint parallel linkage," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 28, pp. 505–511, 2006.

[28] S. Miller, "Simscape multibody contact forces library - file exchange - matlab central," https://www.mathworks.com/matlabcentral/fileexchange/47417-simscape-multibody-contact-forces-library, (Accessed on 07/29/2017).

[29] T. N. Mass, B. Segments, and S. Human, "Anthropometric Data Table 2 : Standing and Sitting Dimensions in meters," *Biomedical Engineering*, pp. 1–3, 2012.

Appendix A

Simulink Block Model

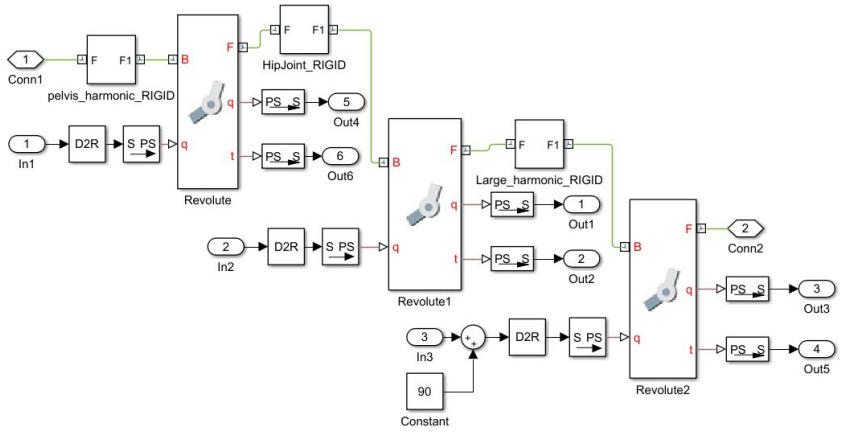


Figure A.1: Simulink Block Diagram for Hip Block

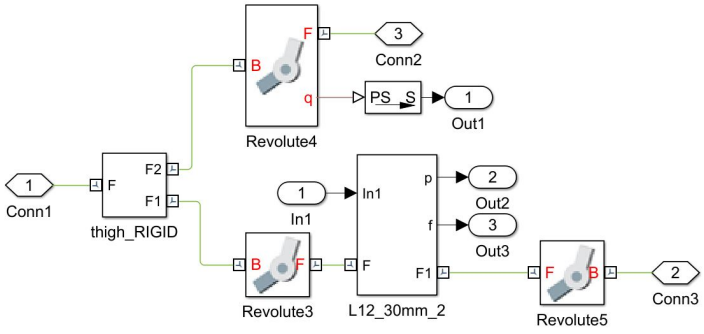


Figure A.2: Simulink Block Diagram for Knee Block

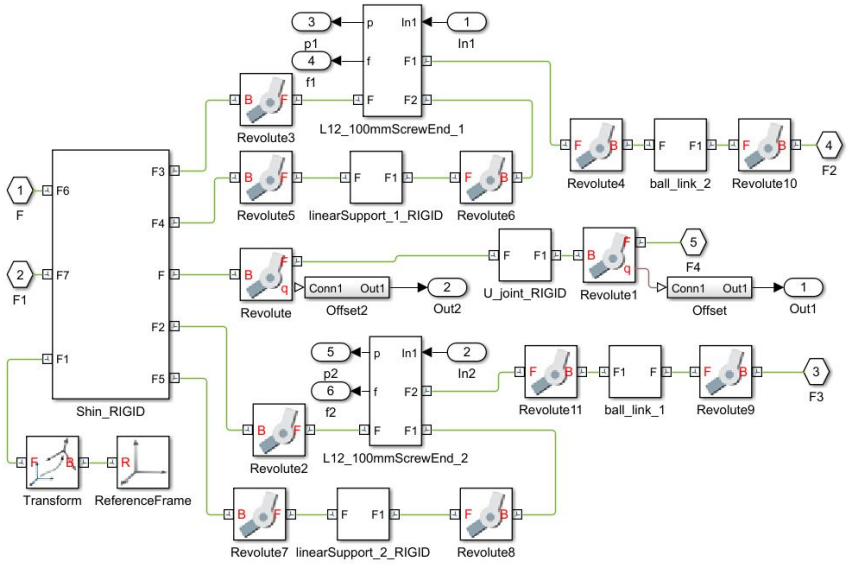


Figure A.3: Simulink Block Diagram for Ankle Block

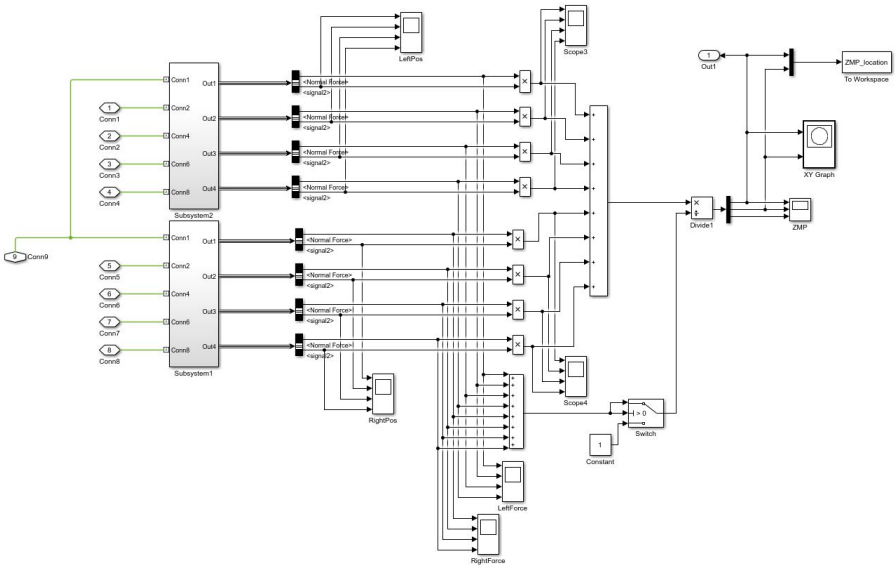


Figure A.4: Simulink Block Diagram for Ankle Block

Appendix B

PCB Schematics

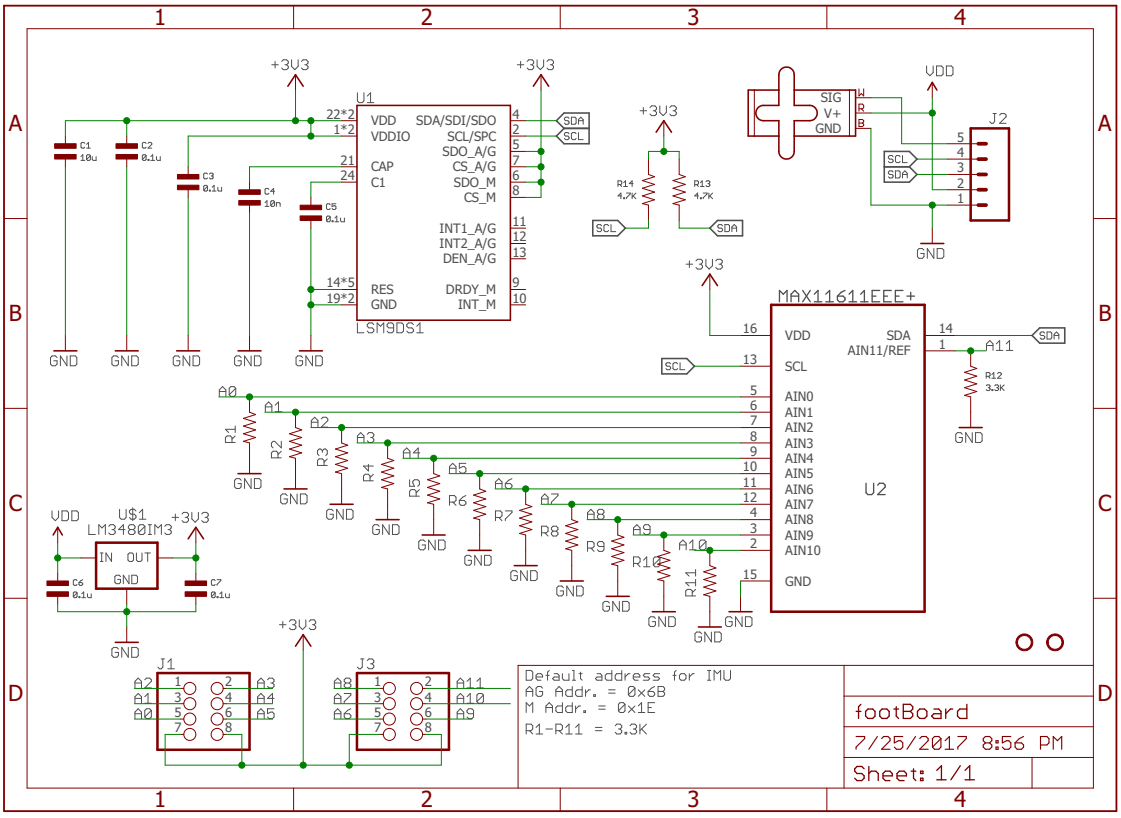


Figure B.1: Schematic for the foot PCB, created in Eagle.

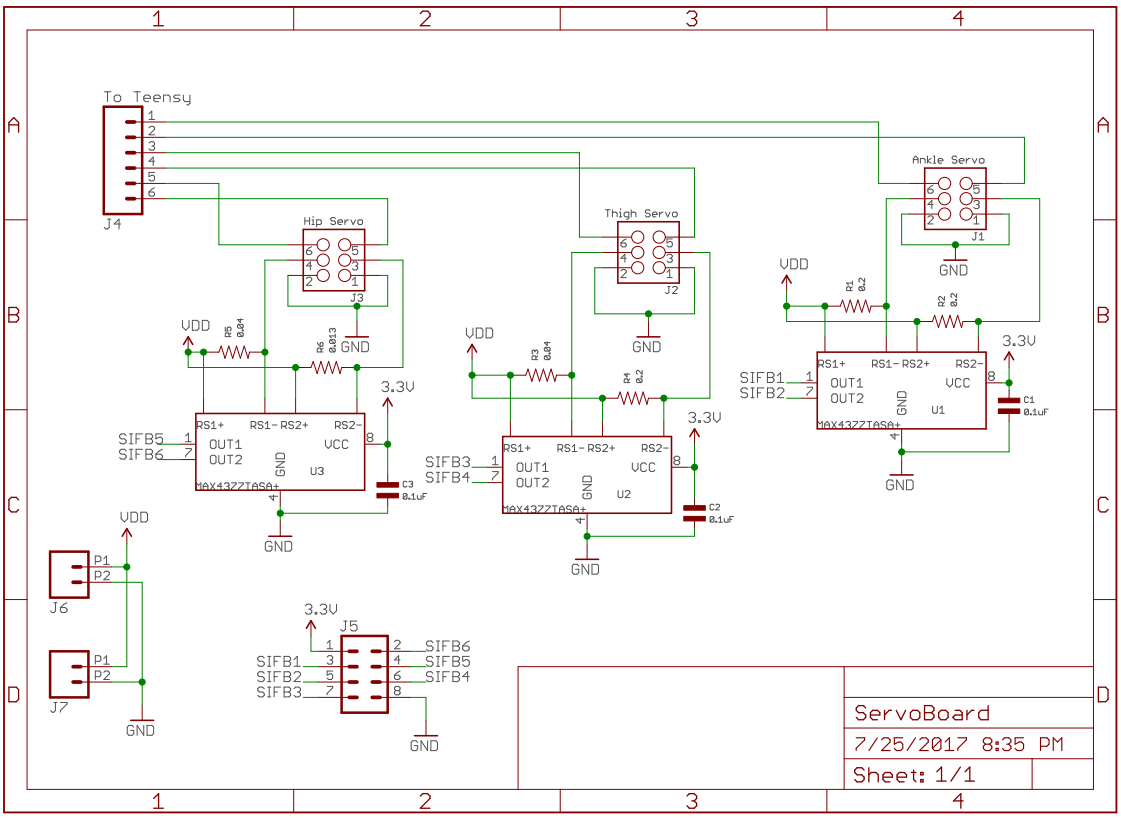


Figure B.2: Schematic for the Power PCB, created in Eagle.

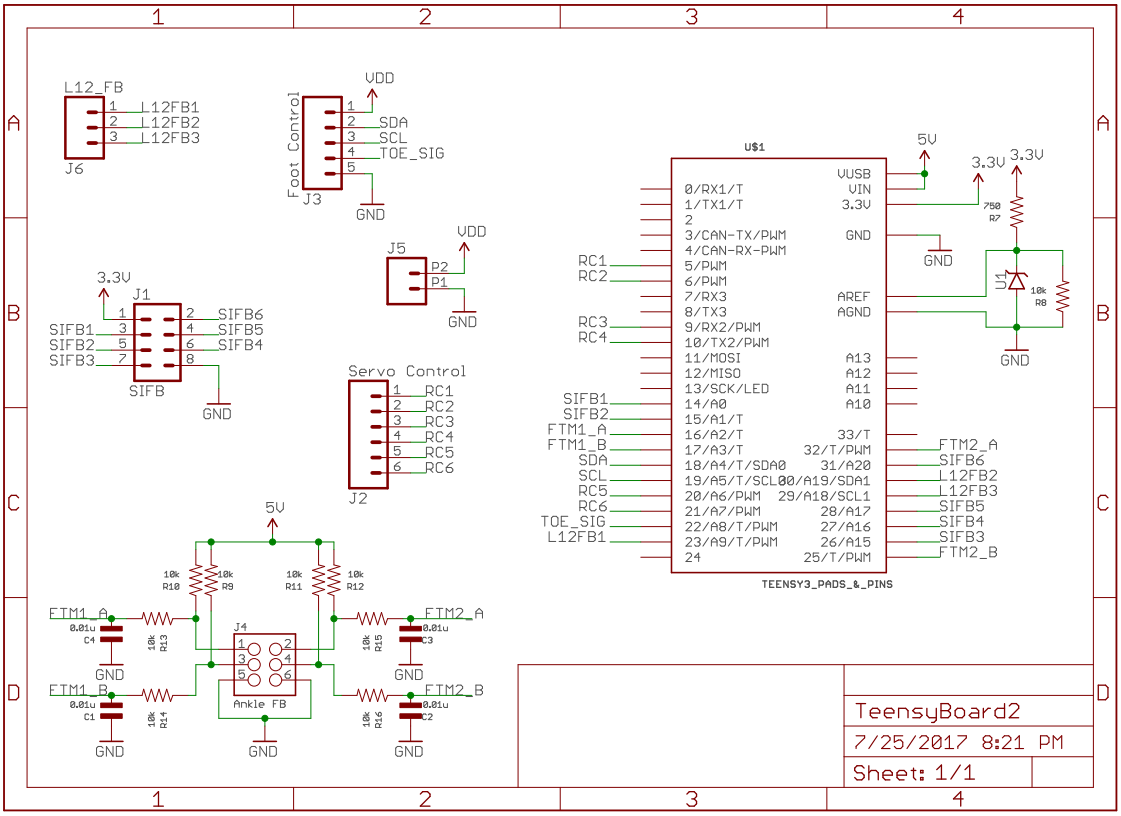


Figure B.3: Schematic for the Teensy interface PCB, created in Eagle.