Development of an Open Source Quadrupedal Robot Platform for Education: SmallKat

by

Keion Bisland

A Thesis

Submitted to the Faculty

of the

WORCESTER POLYTECHNIC INSTITUTE

In partial fulfillment of the requirements for the

Degree of Master of Science

in

Robotics Engineering

by

May 2020

APPROVED:

Professor Haichong Zhang Primary Thesis Advisor

Professor Nicholas Bertozzi Committee Member

Professor Loris Fichera Committee Member

Abstract

In the field of robotics, quadrupedal robotics is a rapidly growing segment. Despite the large number of robotic quadrupeds developed so far, there is currently no platform specifically developed for use in an educational setting. Currently available quadrupeds have several aspects that restrict them to use only in the research labs that developed them, preventing them from being available for use in undergraduate-level classes. This constraint limits the number of people able to gain experience with these highly complex platforms. To enable further development into the field of quadruped robotics, more engineers with in-depth experience with these platforms and the knowledge required to develop and operate them are needed. In this thesis we present the SmallKat platform which strives to fill this space and allow for further development into the fields of dynamic quadruped robotics without the fear of damaging an expensive robot. This thesis proposes a robot designed specifically for the purpose of teaching multiple robotics concepts including kinematics, control, dynamics, trajectory planning, and gait generation. Like many other quadrupedal robots, SmallKat uses 3-DoF legs allowing for coordinated motion in all 3 axes. The size, modularity, cost, and capabilities of the platform are what suit it to teach at a variety of levels. With the integrated sensing and safety features, this platform lends itself to the development of an undergraduate robotics course on quadruped robots, a sample of which is discussed in this thesis. Through the distribution of the SmallKat robot to more schools and universities, the robotics curriculum offered by these universities could be expanded further to offer courses at the undergraduate level in legged robotics.

Acknowledgements

This project could not have been completed without the work and support of both Xavier Little and Kevin Harrington who were both integral parts of the development process and the project as a whole from the initial robot. A special thanks must be extended to the great advisors of this project throughout its many iterations. Prof. Haichong Zhang, Prof. Nicholas Bertozzi, Prof. Loris Fichera, Prof. Michael Ciaraldi, Prof. Bradley Miller, Prof. Gregory Fischer all provided a great deal of support and insight to this project as a whole.

Contents

List of Figures

List of Tables

List of Equations

List of Abbreviations

DoF	Degrees of Freedom
IMU	Inertial Measurement Unit
MCU	Microcontroller
ISP	Independent Study Project
UDP	User Datagram Protocol

Chapter 1

Introduction

Quadrupedal legged robots are a class of mobile robotic systems that achieve locomotion by reproducing the walking gait of a quadruped (e.g., a dog). A number of these robots have been developed over the past ten years such as those seen in [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]. Most of these systems are used in research and/or for military and surveillance applications [11, 12, 13]. One of the major benefits of quadruped robots is that they can achieve locomotion on uneven terrains, therefore being particularly suitable for deployment in outdoor unstructured environments. While the growing interest in these platforms has driven up research and development, to date no robotic quadruped has been developed specifically for education. The work described in this thesis aims to fill this gap. We seek to develop a quadruped robotic system that is inexpensive enough to enable deployment in the context of an undergraduate robotics curriculum. The goals and how these goals are achieved from the mechanical, electrical, and software perspectives of this robot are achieved are explored in this manuscript.

Numerous universities offer courses in mobile robotics; these are facilitated by the numerous platforms available for purchase at a price feasible to use in a course setting such as the TurtleBot 3 Burger at a price point of ~$550. This makes it affordable to provide groups of students with their own robot to work with. Robots with this same concept have been developed internally to many universities to teach a variety of topics, such as the 3-DoF robotic arm developed at Worcester Polytechnic Institute for its RBE 3001 Unified Robotics III: manipulation course. The quadruped platforms developed in a research lab cost far more than would be feasible to provide each team within a class with their robot. Even quadruped robots targeted at being "low cost" [4] result in a platform costing several thousand dollars.

To overcome the limitations outlined above, this thesis introduces the SmallKat platform which was developed to meet a price point similar to the TurtleBot 3 Burger ($550) and is sufficient to teach concepts including kinematics, dynamics, controls, trajectory planning, and gait generation. The platform sensing and modularity in turn extends itself to topics such as artificial intelligence, human-robot interactions, bio-mimicry, and social robotics. The research throughout this thesis demonstrates the process of developing a complex robotic system, designed around education. In addition to a description of SmallKat, the description of the proposed quadrupedal robotic platform for an undergraduate-level course on quadrupedal robotics is discussed.

The structure of this thesis is as follows;

Chapter 2: A review of current quadruped platforms and background into the SmallKat project

Chapter 3: An explanation of the guidelines for teaching with this platform

Chapter 4: The process taken to successfully design the robot

Chapter 5: Testing done to ensure all parts of the robot were fully functional

Chapter 6: A summary of the final design and results from testing of the platform

Chapter 7: Conclusion of the paper and the recommended future or involving this platform

Appendix A: A GitHub of all files related to this project

Appendix B: The testing procedure to be followed to test the motherboard at large scale production

Chapter 2

Background and Literature Review

In this chapter, we review the existing scientific literature on quadruped robots. We then move to introduce the prior versions of the SmallKat platform.

2.1 Survey of Existing Quadrupedal Robotic Platforms

Big Dog by Boston dynamics seen in Fig. 2.1 is a 16-DoF robot. The Big Dog robot used a gasoline engine as a power supply for the robot and had several disadvantages, including the extensive maintenance required of the engine and the very large size and weight of the engine needed to supply enough power for such a large robot. Due to these disadvantages, the platform was discontinued at Boston Dynamics and was instead restarted as the Spot platform. Due to the discontinuation of the Big Dog project, no design information is publicly available for this robot.

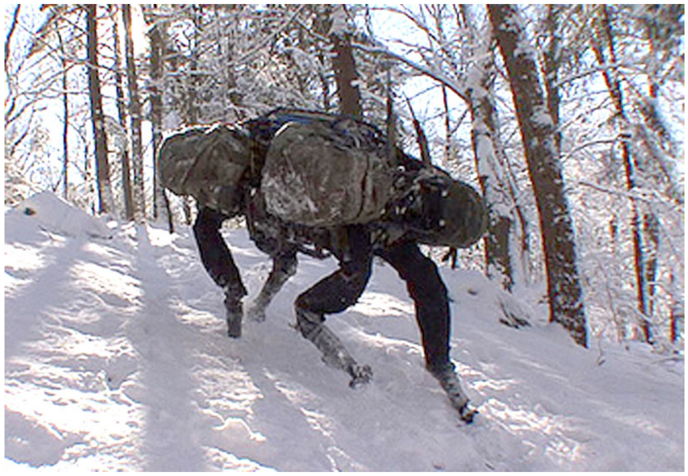

Figure 2.1: Boston Dynamics Big Dog [11]

The Little Dog Robot, developed by Boston Dynamics, seen in Fig. 2.2 was developed in a partnership between Boston Dynamics and DARPA to advance the state of the art of rough terrain locomotion algorithms [12]. The Little Dog robot is a 12-DoF platform with a large range of motion, up to 340° of motion in some joints. The Spot Mini robot developed by Boston Dynamics, seen in Fig. 2.3, is a 12-DoF

Figure 2.2: Boston Dynamics Little Dog [12]

quadruped using custom brushless motors and proprietary force sensors embedded into each joint of the robot. The designs of the Spot Mini robot are proprietary. The Spot Mini robot is available for purchase directly through Boston Dynamics.

Figure 2.3: Boston Dynamics Spot Mini [14]

The cheetah robot, seen in Fig. 2.4, is a 12-DoF quadruped robot developed in a partnership between Boston Dynamics and the Biomimetics lab at MIT (Massachusetts Institute of Technology). The primary intention of this robot was to achieve a very high top speed, resulting in a robot with a top speed experimentally determined to be ~28mph. The cheetah robot was discontinued as a partnership between Boston Dynamics and the Biomimetics lab at MIT, however the project continues to be developed by the Biomimetics lab at MIT.

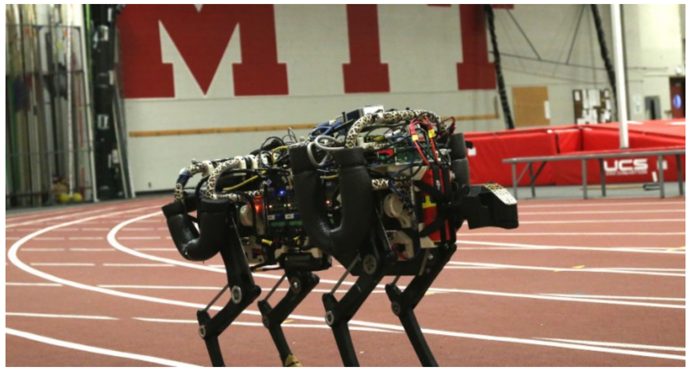

Figure 2.4: MITs Cheetah [15]

The Cheetah II robot is the second iteration of the Cheetah robot. The Cheetah II can be seen in Fig. 2.5. This platform aimed to developed further on the original cheetah robot aims to provide for a much more power efficient, lower speed running and trot gait, at a rate of ~13mph.

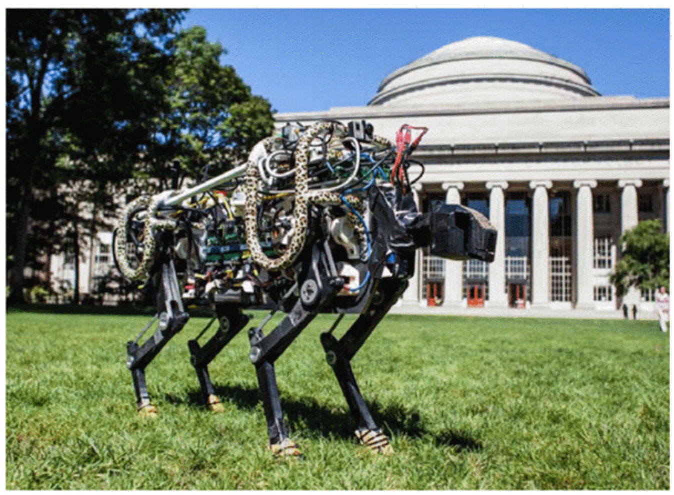

Figure 2.5: MITs Cheetah II [16]

The Cheetah 3 robot, seen in Fig. 2.6, is the third iteration of the cheetah platform. It is a 12-DoF platform with custom brushless motors. Similar to the Boston Dynamics Spot Mini, Cheetah 3 uses joint force sensors. As with the other Cheetah robot versions, the designs are proprietary and currently can only be accessed within the biomimetics lab at MIT. The Cheetah Mini robot developed by

Figure 2.6: MITs Cheetah 3 [2]

the Biomimetics lab at MIT, seen in Fig. 2.7, utilizes custom brushless motors and an overall proprietary design. The overall cost of the robot is ~$3600 [1] in parts not including production costs and assembly time.

The Cheetah Cub robot developed at the EPFL (the Swiss Federal Institute of Technology Lausanne), seen in Fig. 2.8 utilizes off the shelf servo motors to drive its compliant 3-DoF "pantograph" legs. This, combined with the contained size of the platform would in principle make this robot a good platform to satisfy many of the needs for teaching quadruped robotics. However, due to the proprietary design and the custom "pantograph" legs used in the robot, other schools and labs are not easily able to get access to the platform.

The Pneupard robot developed at Osaka University in Japan, seen in Fig. 2.9,

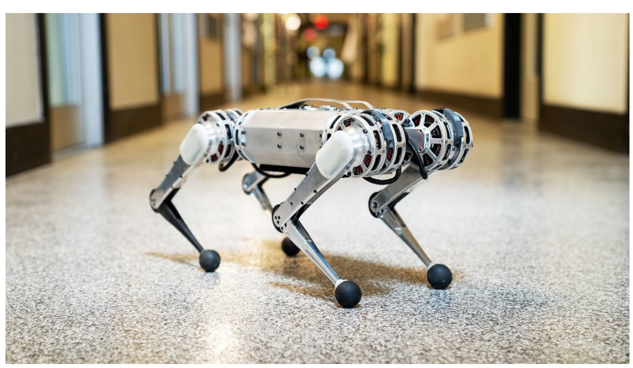

Figure 2.7: MITs Cheetah Mini [4]

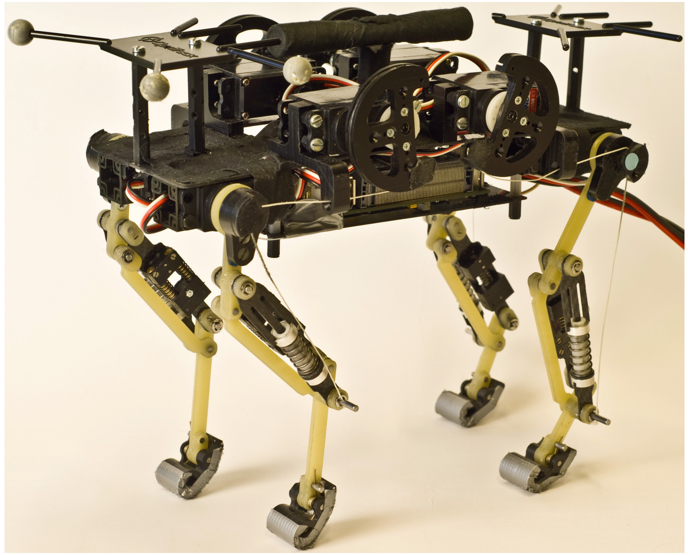

Figure 2.8: EPFLs Cheetah Cub [17]

integrates 3-DoF pneumatic powered artificial muscled to drive its legs. Due to the need for a constant supply of compressed air, the Pneupard robot can only be used in very constrained environments. The HyQ robot developed at the Italian Institute

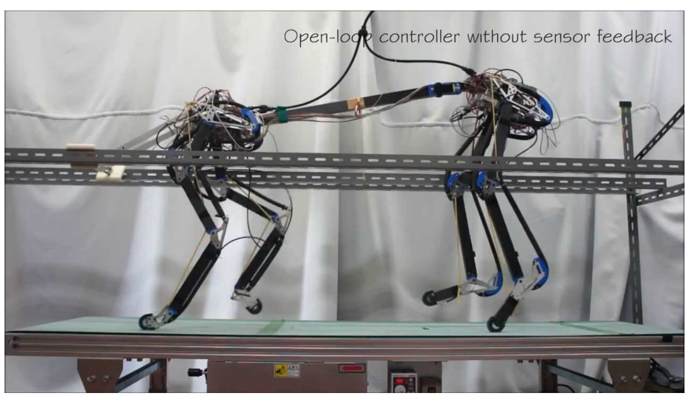

Figure 2.9: Osaka Universities Pneupard [18]

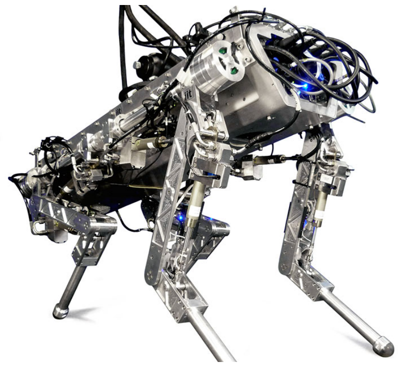

Figure 2.10: IITs HyQ [3]

of Technology, seen in Fig. 2.10, uses 4-DoF legs with 3 rotary joints, ending with a prismatic joint to assist with balance. The HyQ robot is very large at about 100cm long and high and about 50cm wide. This combined with its weight of 70kg results in a robot that should be operated under strict supervision for safety reasons. The ANYmal robot by ANYbotics, seen in Fig. 2.11, utilizes 3-DoF legs. The ANYmal is designed for a similar retail market as Spot Mini and is available for purchase through ANYbotics. The ANYmal's overall price results in a platform feasible to have a very limited number of purchases by an institution.

Figure 2.11: ANYbotics ANYmal [19]

The Aracna robot developed by the Creative Machines Lab at Cornell University is an 8-DoF quadruped robot designed using primarily 3D printed parts and off the shelf hardware [7]. This robot uses a peculiar configuration where each leg is driven by 2 motors instead of the standard 3, which limits its range of motion and its ability for dynamic motion. All designs and software for this platform have been made open source.

Figure 2.12: Aracna Quadruped Robot [7]

2.2 Existing Publications

Figure 2.13 shows the annual number of publications about quadruped robotics. This data was obtained through google scholars statistics as of Spring 2020. From this, it can be seen that the field, especially for research, is continually expanding, increasing by 175% over the span of 9 years. This demonstrates the growing interest annually in the topic and the related capabilities of these robots.

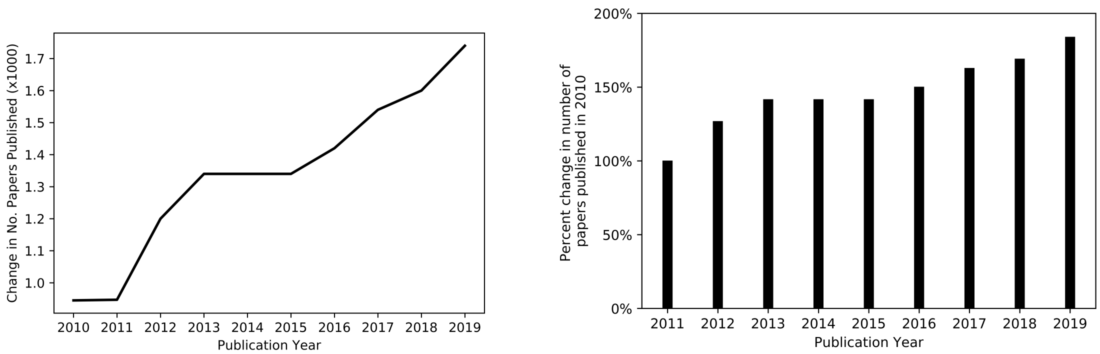

Figure 2.13: Number of Publications Each Year

2.3 Evolution of SmallKat

The SmallKat project has been an ongoing project for 3 years in which several major generational revisions have been made. Each generation explores a different area of feasibility with its advantages and disadvantages, all of which have culminated in the robot developed throughout this paper. Each major generation of the robot, its intended focus, and designs are covered in the following sections.

2.3.1 1^{st} Generation

The initial SmallKat robot was developed as an independent study project between two students over a 7-week term. This was done to explore the possibility of developing a quadrupedal robot capable of walking under its power for under $200. The resulting robot can be seen on the far left in Fig. 2.14, it utilized 3-DoF legs and a 2-DoF head and tail used to shift the center of mass during the walking gait. The final robot used 7g servo motors which proved to be very unreliable resulting in an overall poor performance. The final design of the robot used a Teensy 3.5 microcontroller for low-level motor and sensor control in combination with a Raspberry Pi Zero W for gait computation and development, communicating through USB-

HID. This combination also proved to have several issues due to the inconsistent USB-HID support in the Raspberry Pi architecture.

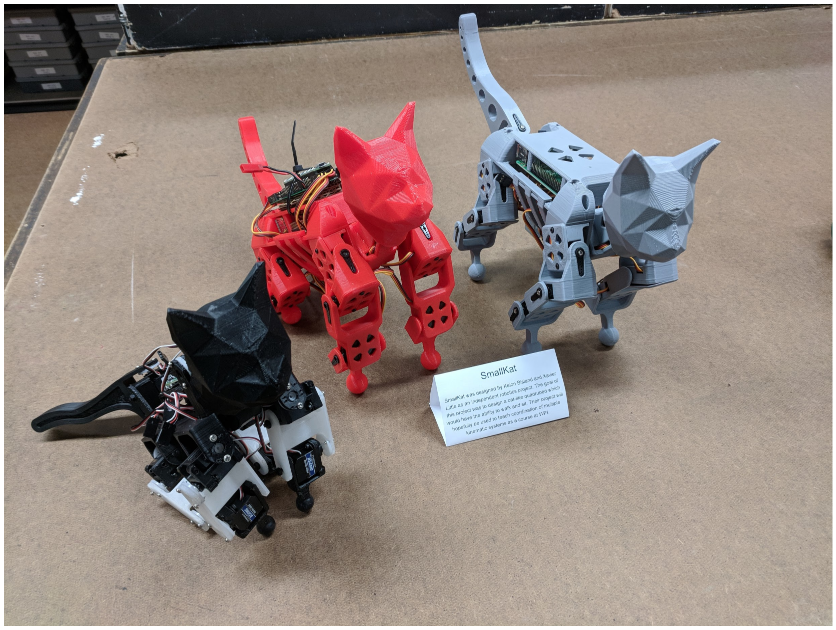

Figure 2.14: 1^{st} Generation: Jaguar (Far Left)

2.3.2 2^{nd} Generation

From the first iteration, several things were learned;

- Small servo motors are highly unreliable despite their high power to weight ratio
- longer Limbs allow for a more natural and more efficient walking gait

In developing the second revision of the SmallKat robot, Grace, the servo motors were upgraded to more standard-sized, 13g metal-geared servos, the Tower Hobby MG92B. These motors have a much higher torque rating that would allow us to make the limbs much longer without stalling the motors. The resulting revision can be seen in Fig. 2.15. This revision also came with several sensors and electrical upgrades including pressure-sensitive feet which were tested but never fully integrated into the final walking gait. An IMU was also integrated into this revision using the BNO055 9-DoF IMU, which gave a very accurate position and rotation of the body of the robot. This version did continue to utilize the Teensy 3.5 microcontroller for the early-stage development. This development was done connected to a standard computer over USB for gait processing. Once the platform was developed further the ESP32 microcontroller was used to have a reliable wireless communication to the robot in order to communicate sensor data to the gait computation and motor data to the robot.

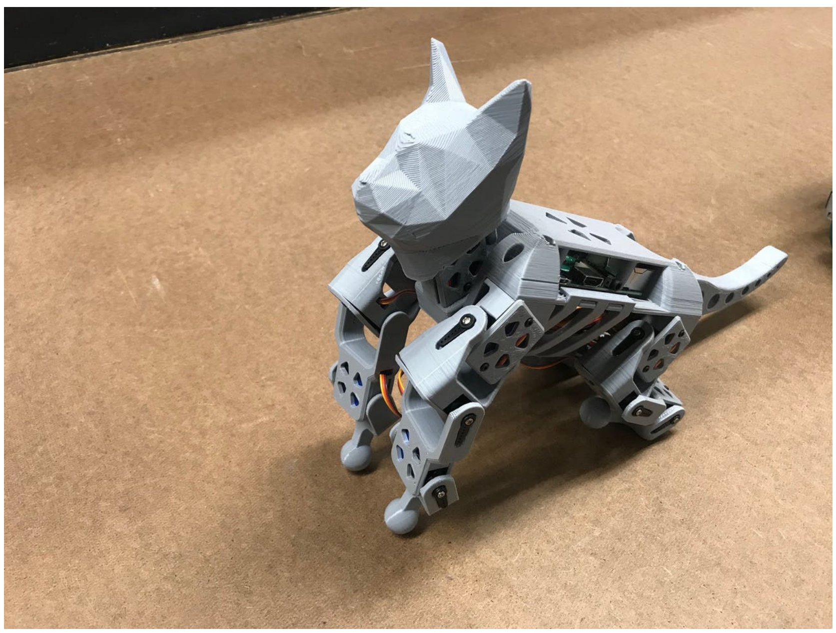

Figure 2.15: 2^{nd} Generation: Grace

2.3.3 3^{rd} Generation

The purpose of this robot's development was to explore the use of larger servos with more feedback as well as expanding the robot to have a 4^{th} redundant DoF and a distributed computing system to lower the overall computation and execution time for the system. Having motors with ~$32\frac{kg}{cm}$ of torque at each joint allowed the final robot, seen in Fig. 2.16, to be much larger than previous iterations. These motors were developed as part of the project and incorporated position, velocity, and force control and feedback which would allow for a more dynamic robot with the potential of adding in compliance control for traversing a more unstable environment.

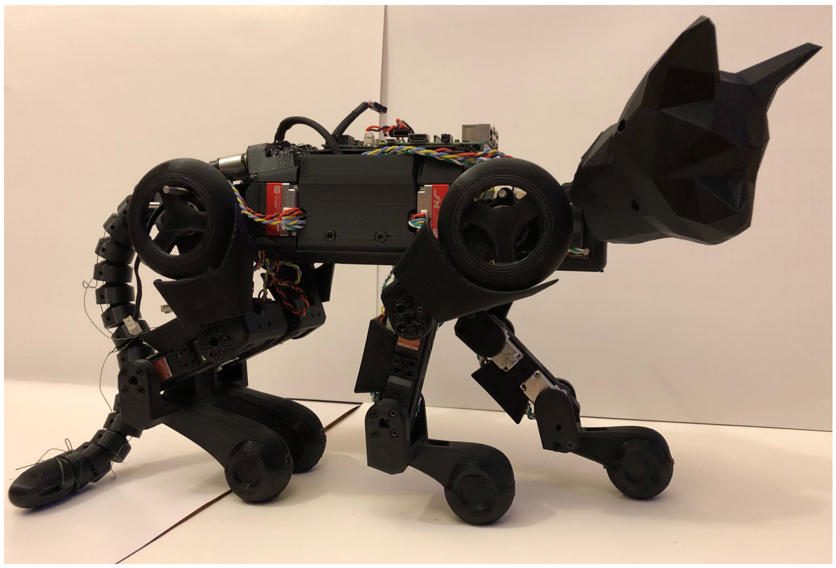

Figure 2.16: 3^{rd} Generation

Chapter 3

Platform Guidelines for Teaching

In this chapter, the project goals and topics needed to effectively teach quadruped dynamics are outlined. These requirements include both the topics that need to be taught but also the requirements of the platform itself for students to effectively learn and implement these topics. In addition the developed course material is outlined with how the implementation was done.

3.1 Project Goal

The goal of this study was to develop a platform capable of performing all aspects needed to effectively teach quadruped dynamics and supporting topics at a price point similar to that of the TurtleBot 3 Burger ($550). To accomplish this task a series of problem statements were devised:

- Can a quadruped robot be designed at a price point similar to that of the TurtleBot 3 Burger?

- Can a quadruped robot be designed to safely operate in a class setting?

- Can a quadruped robot be designed for quick assembly & mass production?

- Can a quadruped course and lab be developed for an undergraduate student level?

3.2 Requirements for Teaching

To design the platform to effectively satisfy our teaching objectives, these first had to be formally defined and subsequently considered in the development process. Sections 3.2.1 through 3.2.4 elaborate on the topics needed to effectively teach quadrupedal robotics.

3.2.1 Kinematics

In learning control and manipulation of a multi DoF system, Kinematics and Denavit–Hartenberg parameters are crucial. Through this series of equations and parameters, all following concepts can be taught and tested sufficiently. All following parameters and equations are in reference to the kinematic diagram seen in Fig. 3.1.

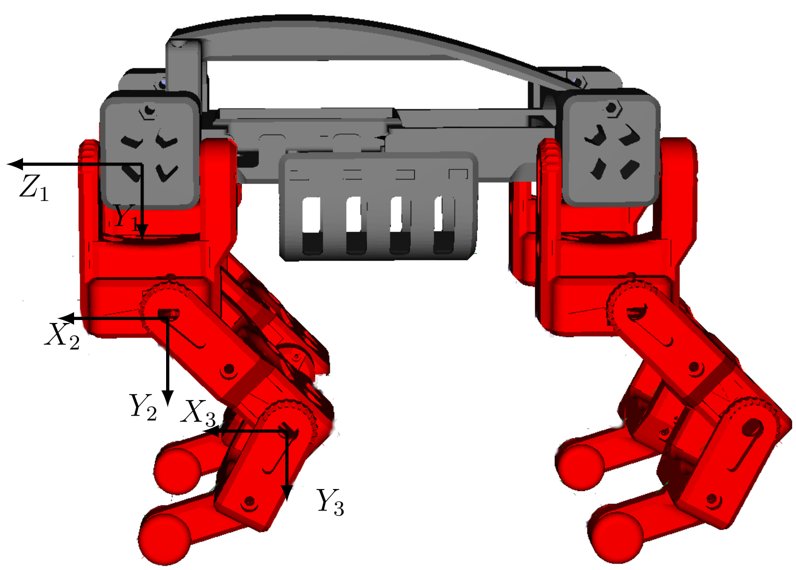

Figure 3.1: Kinematic Diagram of a Leg

D-H Parameters

D-H Parameters explain the configuration of each joint of the robot and its translation from the previous joint in a series of four Parameters. Through these parameters the forward kinematics of the system can be quickly and easily computed with very little modification in the case of a change to the system. This is further explained by J. Denavit and R. Hartenberg in [20]. As an example, the DH parameters of the SmallKat platform can be seen in Table 3.1

Link	a	α	D	θ
0	± 151.42	0	0	0
1	45.50	$-\pi/2$	0	θ_1
2	55.50	0	0	$\theta_2 - \pi/4$
3	60	0	0	$\theta_3 + \pi/2$

Table 3.1: D-H Parameters of a SmallKat Leg. All angles are in radians, all other values are in millimeters.

Forward Kinematics

Forward Kinematics are used to compute the end-effectors position in the X, Y, Z workspace given the current joint angles [21]. The forward kinematics of a single leg can be computed using the equations seen in Eq. 3.1.

θ_i- Joint i's angle of each joint

a_i- Joint i's a value from the DH-parameter table

$$
\begin{aligned}
\mathbf{X} &= a_3 sin(\theta_2 + \theta_3) + a_2 sin(\theta_2) \\
\mathbf{Y} &= sin(\theta_1)(a_1 + a_3 cos(\theta_2 + \theta_3) + a_2 cos(\theta_2)) \\
\mathbf{Z} &= -cos(\theta_1)(a_1 + a_3 cos(\theta_2 + \theta_3) + \theta_2 cos(\theta_2))
\end{aligned} \tag{3.1}
$$

Equation 3.1: Forward Kinematics of a SmallKat Leg

Inverse Kinematics

Inverse kinematics is used to calculate the joint angles required to reach a certain target goal in the workspace [21]. These joint angles of a single leg can be derived from a series on equations seen in Eq. 3.2.

x, y, z- Target coordinate in workspace

a_i- Joint i's a value from the DH-parameter table

$$
\begin{aligned}
\theta_{\mathbf{1}} &= tan_2^{-1}(y/z) \\
\theta_{\mathbf{2}} &= tan^{-1}(x/\sqrt{(}z^2 + y^2)) + cos^{-1}\frac{(x^2 + \sqrt{(}z^2 + y^2)^2 + a_2^2 - a_3^2}{2 * a_2 * \sqrt{(}x^2 + \sqrt{(}z^2 + y^2)^2))} \\
\theta_{\mathbf{3}} &= cos^{-1}(((x^2 + \sqrt{(}z^2 + y^2)^2) - (a_2^2 + a_3^2))/(2 * a_2 * a_3))
\end{aligned} \tag{3.2}
$$

Equation 3.2: Inverse Kinematics of a SmallKat Leg

3.2.2 Trajectory Generation

Following kinematics and preceding gait generation the topics involved in trajectory planning are of high importance. Path planning provides a geometric description of robot motion, but it does not specify any dynamic aspects of the motion [21].

3.2.3 Gait Generation

Gait generation is the formulation and selection of a sequence of coordinated leg and body motions that propel a legged robot along a desired path. This covers the trajectories that must be followed for each leg in a given sequence to achieve a specific body trajectory, moving the entire robot in a specified direction. Further information can be found in [22].

3.2.4 Controls and Dynamics

When developing a complex system controls and dynamics are crucial to have the robot accurately and safely perform tasks and actions commanded to it. The platform must be able to have both control systems and dynamic controllers implemented to account for and respond accordingly to external input from the environment.

3.3 Course Implementation

To test the robot's hardware abilities as well as the software developed for the platform, a course was developed with both course and lab materials. This course was implemented as a voluntary independent study project (ISP) with three students. Each week these students were given a series of videos and lecture slides that cover the topics for the week ranging from an introduction to kinematics through trajectory planning and gait generation. Along with this, each week the students were assigned tasks to complete as lab material, each student worked independently on the project with support and advice from the SmallKat team and the project advisors. The students followed the course schedule as seen in Table 3.2.

Week	Course Topic	Lab Task
1	Familiarize with robot and software	Assemble robot
2	Familiarize with scripting interface	Test robot with test software
3	How to create a walking cycle	Perform static motion, movement in Z
4	How to create a walking gait trajectory	Develop and test trajectory on one leg
5	Balancing based on contact area	Adjust the pose of the robot based on IMU input
6	Trajectory planning and threads	Integrate threading into the trajectory generation
7		Final Evaluation, walking and turning

Table 3.2: SmallKat ISP Course Schedule

All student concerns were recorded and addressed throughout the course. These ranged from assembly instruction modifications to integration and addition of more sensors for future advancement and development. At the end of the course the students were asked for feedback on the platform as a whole with an overall positive response with minor changes for improvements. Since then, all concerns that would not change the robots price point drastically have been integrated into a revision of the robot including a revised power management system and updated robot firmware and several mechanical and software advancements such as the calibration jig as well as a simpler means of launching and controlling the robot from the high-level simulation environment. A number of the student's concerns revolved around the desire for more advanced sensors, specifically encoders and force sensors. The integration of independent encoders for each limb was decided against due to the drastic increase in the overall cost, however, in the revision of the electronics following the course, current sensors were integrated to allow for a rough means of force sensing.

As a final project for the course, each student was tasked to develop a walking gait for the robot with the ability to walk straight and turn. Further credit was awarded for the development of a semi-dynamic gait which would use the head and tail to balance the robot in the case of an external input such as mildly uneven terrain. All students were able to accomplish the task of developing a successful walking gait with most able to integrate the head and tail for dynamic balancing. After completion, most of the students were interested in further developing on the system and extending this project into future work.

Chapter 4

Robot Design

This chapter goes into the process taken to develop the robot in its three major sections, the mechanical design, electronics, and the software for the robot.

4.1 Mechanical Design

4.1.1 Motor Selection

For motors, the 13g servo form factor was chosen as it would allow the overall robot to remain small in size and lightweight. The final list of motors included four motors from different suppliers and manufacturers, the torque ratings of which can be seen in Table 4.1. After testing with all of the considered motors ability to output their advertised torque and their ability to remain at its stall torque for extended periods, the final motor chosen was the Tower Hobby MG92B for its high size to torque ratio, being far higher than the other options at ~3.5 kg/cm as well as its availability from many distributors around the world.

Motor	Torque @ 6V (kg/cm)
MG90D	2.4
MG92B	3.5
MG90S	2.4
TGY-9018MG	2.5

Table 4.1: Torque values of different motors

4.1.2 Ease of Manufacturing

While designing the SmallKat robot a great deal of effort was spent to ensure parts could be easily manufactured. Both 3D printing and resin casting have different constraints in what can be produced successfully, efficiently, and quickly. 3D printing because it would allow the robot to be iterated quickly and make the platform available to the most people. However for mass production, 3D printing was chosen as the primary manufacturing method because would not be used due to the ~2.5 days of printing time for each robot. To resolve this, for production at a large scale, a series of molds for the small parts of the limbs would be created and parts would be cast using a polyurethane resin. Larger parts such as the head and tail would remain printed to keep their overall mass low. This required all parts that would be cast to be designed with both manufacturing styles requirements in mind ensuring there were are few areas with unreachable overhangs and that parts could be placed flat to reduce the amount of support material while printing.

4.1.3 Ease of Assembly

While ease of manufacturing encapsulates the majority of the design process, the means of assembling was also considered carefully. By analyzing the process taken to assemble an early prototype of the robot and adjusting the design, the assembly time was reduced from ~5hrs to ~1.5hrs. This was done by observing the aspects of the assembly of the previous generation and noting the areas of the assembly that would slow the process. The major hindrances of the process were finding which part was correct as each link was specific to the leg it was to be assembled onto, meaning each part was unique and had only one correct place, in addition to this, the number of small M2 screws slowed the assembly as they were hard to handle and manipulate into place. The final revision of the design utilized 2 lengths of M3 bolts and their corresponding nuts to assemble the whole robot. In addition to the use of a smaller variety of parts, reducing from 79 unique parts to 22 unique parts that are used in a variety of positions making the assembly far less time-intensive.

4.2 Electronics

The electrical system of the SmallKat platform incorporates several features including a main microcontroller, a high current power regulator, a 9-DoF inertial measurement unit as well as several safety features to ensure reliability and consistency of the platform. The system architecture of the motherboard developed can be seen in Fig. 4.1. This figure shows the major sub-modules of the motherboard and which modules can communicate and interact with one another. It can be seen the safety microcontroller can interact with the charging circuitry and the onboard power supplies and is then able to relay any necessary information back to the main microcontroller which interfaces with the IMU and the servo motors.

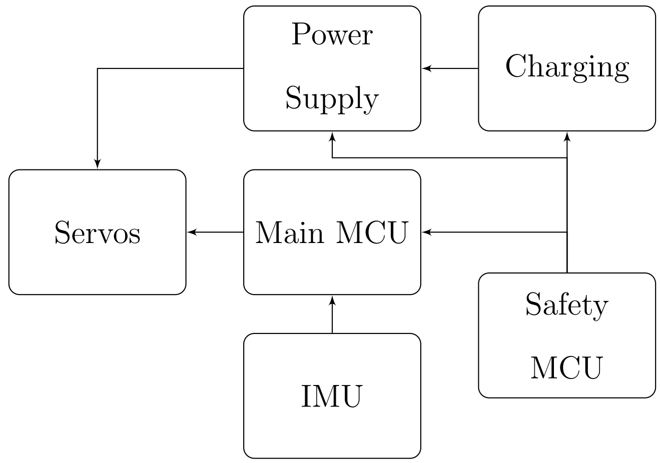

Figure 4.1: Overview of the Robots Electrical System

4.2.1 Microcontroller

When choosing the microcontroller there were a few requirements set forth, it would have to be able to run at least 16 independent PWM signals, communicate through either USB-HID or UDP and have hardware i^2c support. From this, a list of currently available microcontrollers development boards was constructed:

1. Teensy 3.5/3.6
2. Teensy 4.0
3. ESP32
4. STM32

for each of these available development boards, a custom expansion board was developed. This expansion board broke out all necessary servo channels to easily accessible pins, and integrated an IMU into the center of the PCB. From these microcontrollers it was narrowed to either the Teensy 3.5 or the ESP32, one option providing hardware USB-HID and the other UDP over WiFi. From this a final version of the motherboard was developed using the available boards with integrated power regulation. Both boards were then tested thoroughly, by assessing their ability to communicate with the desired software, their ease of programming and their ability to drive all necessary servo motors.

4.2.2 IMU

The IMU chosen was the BNO055 as it filters the data returned by the read gyroscope, accelerometer, and magnetometer data through an internal Kalman filter processed on the embedded microcontroller. This allows the system to have a very reliable source of feedback, providing a reference for the robot in the environment. The IMU is also able to provide a gravity vector and the Euler angles within each axis of rotation.

4.2.3 Power Regulation

- Due to the power requirement of the motors chosen, a consistent 6V power supply with up to 20A continuous current is required. This power requirement was done by referencing the advertised motor specifications with a max operating voltage of 6V, several MG92B motors were then tested at their stall torque to verify the ~1A current draw, this was then scaled up linearly for the 16 necessary motors in a stall scenario. 6V at 20A is a very uncommon power configuration with standard 2-cell lithium batteries supplying a nominal 7.4V and most bench power supplied only supplying a maximum of 10A. As lithium batteries scale in size proportionally to the stored power, to increase the run time of the robot, a 1.2mAh 16V battery was chosen due to its physical size and weight being appropriately sized match the dimensions and capabilities of the robot. To regulate this to the required 6V, a high current switch mode power supply was designed. With a peak efficiency of 90%, the onboard supply is highly efficient at converting high voltage, low current supplies down to 6V at the required 20A, this power supply effectively converts the 1.2Ah at 16V to a 10.8Ah at 6V battery.

During the development stages of the motherboard, off the shelf development boards and modules were used for the microcontroller, IMU, and power regulation. This drastically increased the overall cost of the electronics in the system. Once the system was proven and tested, the final revision of the motherboard was designed with all components integrated directly onto the motherboard, using no off the shelf modules. This resulted in a cost reduction of about 80%, reducing from $260 down to $50, a savings of $210 per robot. In addition to the cost reduction, The overall number of through-hole components was drastically reduced, making it marginally easier for mass production. The schematics and all related files can be found on the Operation SmallKat GitHub organization seen in Appendix A.

Testing Fixture

To allow for rapid testing and programming of the motherboard, a series of exposed test points were placed along the reverse side of the motherboard that allows it to be placed into a test fixture, comprised of a mirrored motherboard with spring-loaded pins. On this test fixture the motherboard can have a series of tests run to confirm functionality as expected. The test points test the USB programming circuit, the i^2c devices, the microcontroller functionality, and the 3.3V power supply. In addition to these, 16V is injected to simulate a battery being connected and the output voltage for the integrated switch-mode regulators is tested. The testing procedure can be seen in Appendix B

Safety

In designing the system, great caution had to be taken with the lithium-ion battery used to power the robot. To minimize the chance of a battery failure, a series of fuses were integrated into the motherboard to prevent short circuits or motor failures. In addition to this current sensors have been added to each motor to determine major stall states to prevent complete failure of the motor. controlling all of the safety and charging onboard is a secondary microcontroller, implemented as a safety controller. This microcontroller will monitor battery voltage, charge state, battery charging and battery balancing to ensure each cell is evenly charged for longevity and safety. This microcontroller will also maintain control over the onboard power regulators and be able to turn them on or off in a series of cases. This microcontroller will remain on and in low power mode while the robot is powered down to monitor the battery over time. To safely charge the battery pack, a charging system utilizing USB power delivery [23] was implemented along with a cell balancing and monitoring system.

4.3 Software

In choosing the programming language, development environment, and in turn some of the electronics and hardware, several things had to be considered. The prime focus came down to choosing the programming languages used to develop both at the high level on the main processing and the low level on the microcontroller. Each of these comes with a varying list of potential languages. So to determine which language should be chosen, a list of programming languages at both the high level and low level were devised. The primary list of languages for this high-level control development includes C/C++, Java, Python, Matlab and GO. For the low-level programming language C/C++ and micro python were the main choices. The control structure of the robot can be seen in Fig. 4.2. This figure shows the division of computation of both the high-level controller and the low-level program on the microcontroller as well as the communication between them. The high- level controller computes the system dynamics, the gait for the robot to follow, and its associated trajectories, these are then passed to the UDP handler which communicates the joint data to the robot. The robot then receives the necessary joint angles over WiFi, processes the UDP packet, and assigns the received joint angles to the servo motors. The microcontroller then reads all sensor data and reports it back to the high- level controller for processing in the next cycle.

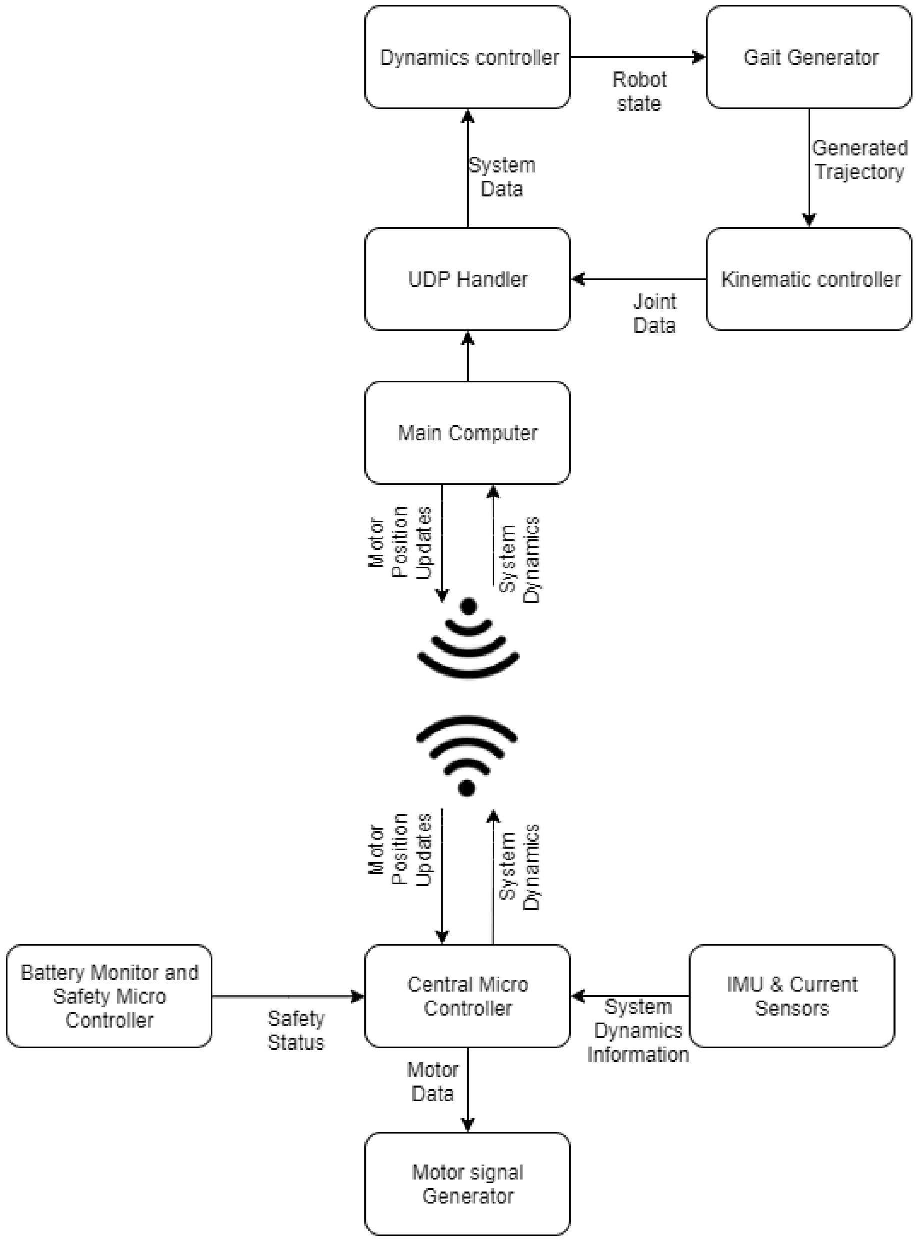

Figure 4.2: System Control Diagram

4.3.1 High Level Control

When choosing the language for the high-level programming environment, several considerations were made. The operation support off each language being of high importance as all computation and communication should ideally be system agnostic. From there the development speed and run time speed were analyzed for each language as well as how common the language is to aid in development and education. Given all of these criteria, Java was chosen, specifically groovy, a scripting language based in java. Groovy provides the user with good cross-platform operability as it runs natively in the java runtime environment. Due to this, it also takes the benefits of being extremely well supported as Java is currently the second most common programming language according to the Tiobe index [24].

4.3.2 Micro-Controller

When deciding the language to use for microcontroller development several factors such as environment setup, language functionality, and library support were considered. These criteria were analyzed for both micro-python and C/C++ in two environments, through the Arduino IDE or the Espressif integrated development framework (IDF). In end, C/C++ through the Arduino IDE was chosen due to the simplicity of the system, the integrated RTOS and availability of documentation and support, and the overall efficiency of the language in comparison to Micro Python. The Arduino environment was chosen over the integrated development framework due to the ease of setup and ease of sharing of the code; libraries can be easily updated and the code can be easily refreshed with little to no user input for sample and starter code.

4.3.3 Computational Environment

In the end, the Bowler Studio programming environment [25], developed by a team member of the project, was chosen for all high-level computation. Bowler Studio is a robot development application that combines scripting and device management with powerful control and processing features. This environment was chosen due to its active support, ease of development, and the integrated simulation environment using the Bullet physics engine [26]. This environment and its integrated simulation environment allow for the development of all components of the system without the need for testing on the real robot. Through the use of the simulator an effective and correct inverse kinematics, control scheme, and walking gait can be developed, tested, and then interfaced with the robot with no changes made. Using the integrated physics simulator, realistic external forces can be applied to test any dynamic engines implemented. The robot implemented in Bowler Studio can be seen in Fig. 4.3. This figure also shows the control environment allowing the user to simply manipulate with the simulated model as well as communicate with the physical robot.

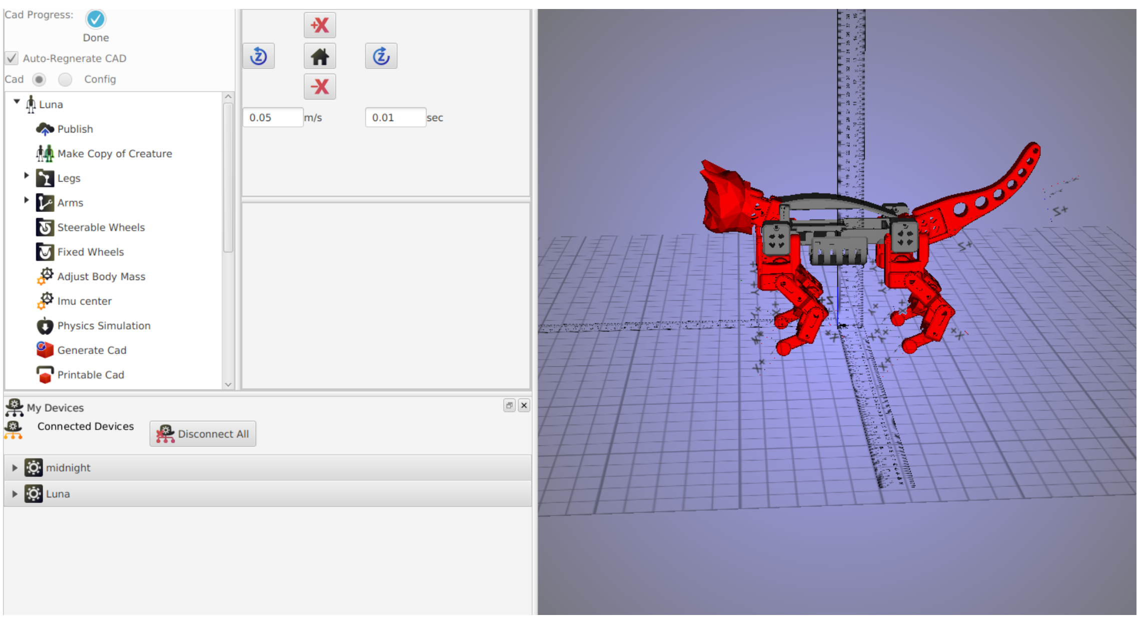

Figure 4.3: SmallKat implemented in Bowler Studio

Chapter 5

Platform Testing

This chapter goes into the testing that was performed to ensure reliability and measurable improvements to the system. These changes were made in three major divisions: software, mechanical, and electrical, each of which will be spoken about in the following sections.

5.1 Software Testing

To test the software changes, a known working platform was kept with no hardware changes made to ensure any software changes that were made at either the high level in Bowler Studio or at the low level on the microcontroller would not negatively affect the performance of the robot. These changes were then tested using the reference robot being a Grace model seen in Fig. 2.15 as it was a known working platform utilizing the same gait planner and kinematics engine with changes only being made to the robot specific DH-Parameters. After confirmed working, the changes were then tested using the new hardware to ensure it continued to function as expected.

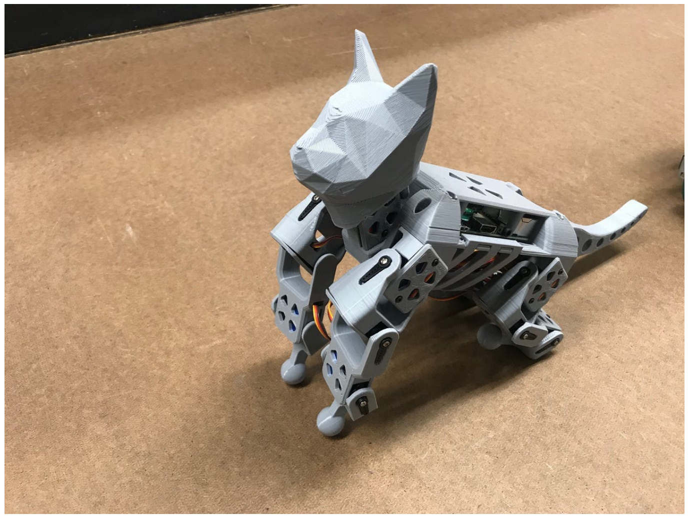

Figure 2.15: SmallKat Grace (repeated from page 14)

5.2 Electrical Testing

For the electrical system there are several validation steps to ensure full functionality and reliability. Primarily the motherboard was tested for full functionality, ensuring it was able to correctly provide the 16 servo PWM signals and control them independently, it was able to communicate with all onboard sensors and the main microcontroller was able to be programmed successfully with the integrated programming circuit. Secondly, the power supply integrated onto the board was tested for resilience, the power supply was set to provide 10A at 6V which would be much higher than the general use case however is a possible current the robot is expected to draw during certain motions; an electronic load was then used to draw full power from the board until failure, the supply was able to continuously provide 10A at 6V stably for over 2 hours at which time the test was stopped and the power supply concluded to be sufficiently resilient. Finally the new motherboard was tested in a physical robot, utilizing the Grace model seen in Fig. 2.15 to start as it was a known working model and then the Luna model seen in Fig. 6.1. These tests were completed successfully with the robot able to walk under its power and any issues found were fixed in subsequent motherboard revisions.

5.3 Hardware Testing

When designing the Luna model, the finalized version, seen in Fig. 6.1 several precursory tests were done to test the lifting strength of the motors, to verify the advertised motor strength. The motors were all able to match closely to the advertised torque ratings, leading us to choose the motor providing the most torque. A test leg was developed and assembled using the chosen servos and tested for its ability to support and estimated weight. Once this was done the dimensions of the leg were adjusted and the remainder of the robot was printed and assembled. After minor adjustments and tolerance updates a finalized model was completed.

Once a final model was completed, the legs were then printed in several materials including PLA, PETG, ABS, PC and PA66 nylon, the material and ease of printing properties can be seen in Table 5.1; each of these having a variety of advantages and disadvantages. Each of these legs was assembled and tested for link failure, this would occur due to heat generated by the motor causing the plastic to deform. In the end PETG was the chosen material as it comes with several temperature performance benefits as well as a low price and a high ease of printing. When using PETG there was no notable deformation after the testing. All other materials except PLA were also able to withstand the temperatures generated but come with an increased cost or difficulty in printing.

Material	Glass Transition Temperature (°C)	Cost ($/kg)	Ease of Printing
PLA	60	22	High
ABS	105	25	Low
PETG	88	25	High
PC	147	53	Low
Nylon	70	45	Medium

Table 5.1: Material Comparison

Chapter 6

Design Summary, Validation Results, and Discussion

This Chapter will summarize the resulting hardware developed in Chapter 4 and the results of the testing procedure done on the robot, further described in Section 6.4. The results analyze the accuracy capabilities of the developed robot and the feedback from the students who participated in the independent study project described in Section 3.3.

6.1 Final Robot

After all gained knowledge through the previous versions of the platform, seen in Section 2.3.1 through Section 2.3.3, and all of the teaching requirements and course guidelines in Section 3.2, a finalized version for the platform with the intention of use in education was developed, the process by which can be seen in Chapter 4. The resulting robot developed through the methods explained in Sections 4.1 through Section 4.3 can be seen in Fig. 6.1. The intended use for this platform would be to distribute a robot to each team of 3-4 students within a class to perform a series of lab assignments on developing a dynamic walking gait while learning and improving abilities in kinematics, trajectory planning, controls and dynamics; all of which are integral parts of robotics. To do this safely, the platform must conform to several aspects described further in Section 3.2.

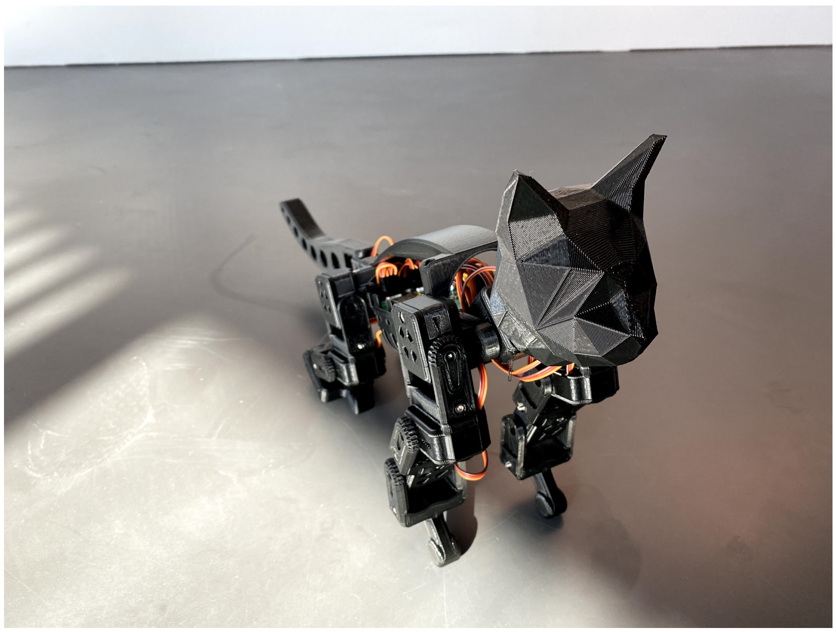

Figure 6.1: 4^{th} Generation: Luna

6.2 Summary of Mechanical Design

In the end robot was successfully designed utilizing four, 3-DoF legs integrating three Tower Hobby MG92B servo motors for actuation and 2-DoF head and tail that can be utilized for balancing and to shift the center of mass as needed. The use of 3-DoF legs was chosen for its overall simplicity, low cost both computationally and in parts and materials. This means that the platform becomes more physically and computationally available to more people, not needing special hardware to operate the robot. The resulting design integrates all necessary sensors, electronics, and battery pack. An exploded view of the final design can be seen in Fig. 6.2. The DH parameters of this robot can be seen in Table 3.1. The final design is made of 22 unique parts, comprising the 79 total parts that are used to build the robot. all of these parts are easily 3D printable and many retain the ability to be resin cast for large scale mass production. All designs have been made open source both as the finalized model and as modifiable source files. The final robot spans ~50cm in length making it appropriately sized to operate on a workbench with the students operating it in close vicinity.

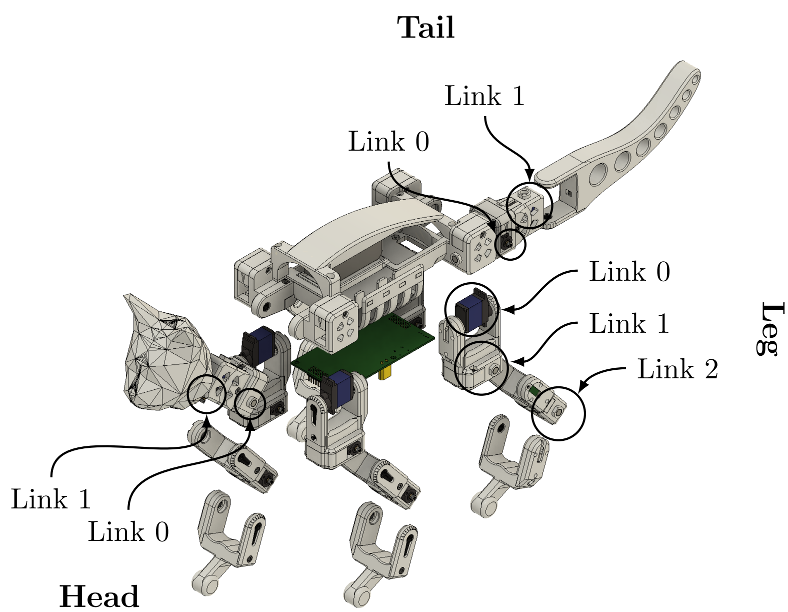

Figure 6.2: Exploded View of the Robot

Due to the minimal number of parts and the emphasis placed on ease of assembly and ease of servicing parts, the entire robot can be constructed in ~1 to 2 hours and any given motor can be replaced with the removal of 2 to 3 screws and can, therefore, be done quickly. Using the designed calibration jig seen in Fig. 6.3, in which an un-calibrated leg is inserted, the servos are set to the neutral home position and the servo horns are inserted, the total time taken to calibrate the robot has been decreased drastically, reducing from 5 to 6 hours to between 1 and 2 hours.

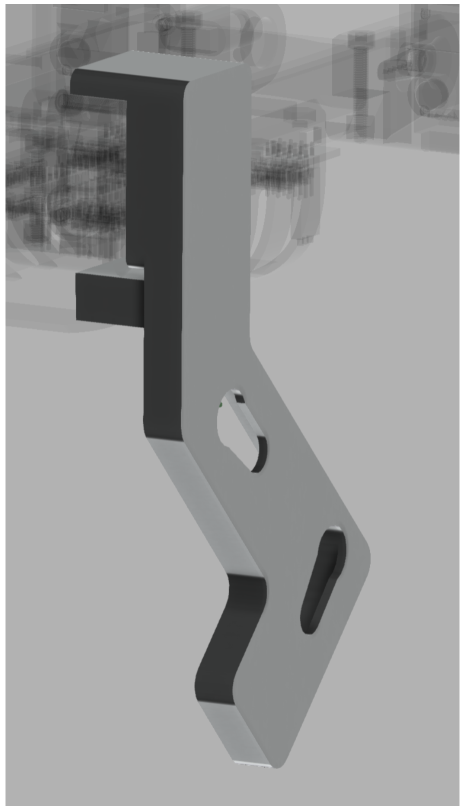

Figure 6.3: Calibration Jig

6.3 Summary of Electronics Design

The finalized electronics came with a major focus on functionality, speed, and safety. The finalized motherboard integrated 20 Servo channels, a 6V 20A power supply, integrated fuses for each bank of four motors, 16Bit current sensors for each motor as well as integrated battery charging and monitoring circuitry, USB C for power delivery and programming, two dedicated microcontrollers and a 9-DoF IMU. The resulting motherboard can be seen in Fig. 6.4.

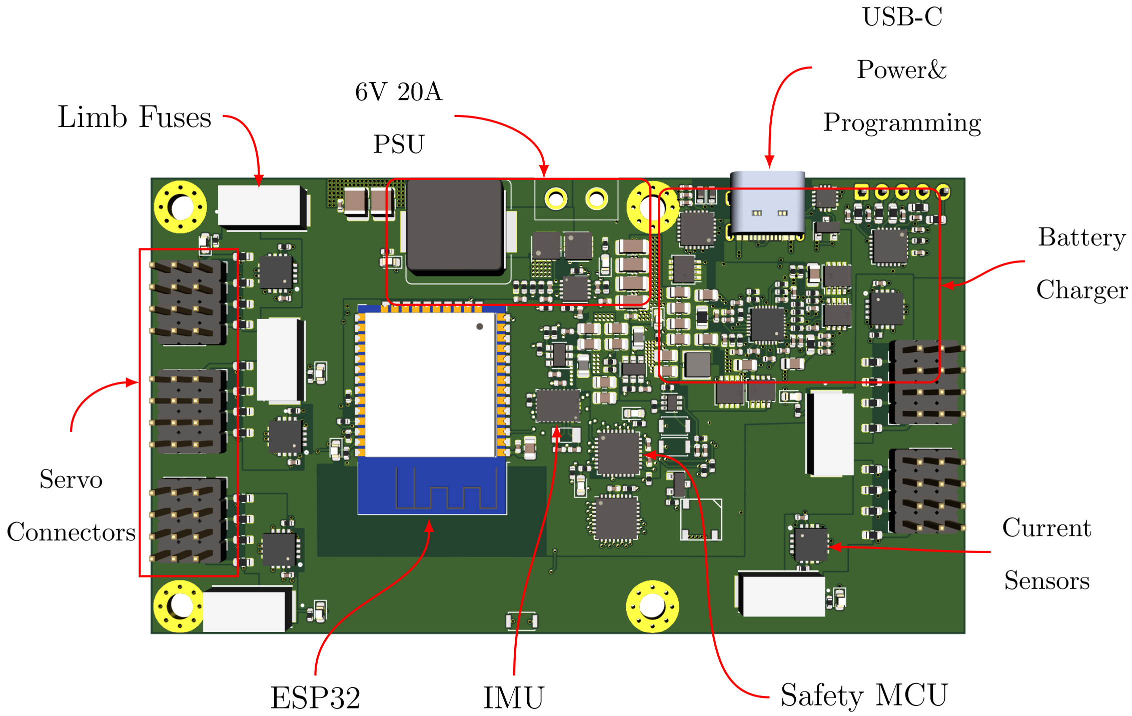

Figure 6.4: Resulting Motherboard

The main microcontroller used was the ESP32, chosen for its support in the Arduino environment, high processing power being a dual-core 240MHz processor and its integrated WiFi and Bluetooth capabilities. The secondary microcontroller chosen used to monitor each motors current, the batteries charging and discharging states, and governs all power supplies on the robot was the LPC824 due to its low current consumption, low cost, and the vast array of general I/O and peripherals. The chosen IMU was the BNO055 due to its high fidelity output due to its embedded microcontroller used to filter all read data through an internal Kalman filter from the accelerometer, gyroscope, and magnetometer. With this integrated Kalman filter the Euler angles of the system as well as the directional gravity vector can be obtained.

The overall power requirements for this motherboard resulted in a dedicated switch-mode power regulator capable of converting the 16V supplied by the battery to constant 6V with a continuous current draw of ~20A. This would only be reached if all servos on the robot were to stall simultaneously. To test this aspect of the robot, 10A was drawn continuously for 2 hours to test for failure, however, a failure point was not reached, meaning in normal operation there would be no issues seen supplying the necessary power to the robot. This switch mode power supply can be enabled and disabled directly from the safety microcontroller for long term storage and in the case of a detected error. While in operation, the current of each motor is monitored for stall or failure. In the case a motor fails the power to the system can be cut until the motor is repaired. In combination with this, if a failure is not detected in time, the integrated fuses are used to prevent the battery from being short-circuited which could result in a battery failure. The general usages and charging of the battery is closely monitored to ensure the battery is not overcharged or over-discharged as well as to ensure all battery cells are closely matched in voltage to ensure the longevity of the battery pack and reduce its chance of failure.

6.4 Motion and Accuracy

A series of tests were run to compare the robots ability to perform tasks required for the teaching objectives listed in Section 3.2. To prove the derived kinematic equations are correct and the robot, seen with its coordinate frame in Fig. 6.5, is able to perform the desired motions the controller computes, 14Bit absolute magnetic encoders were attached to each joint of one leg. The controller was then used to compute the inverse kinematics to reach the desired position and the required joint angles communicated to the robot. The encoders were then used to record the joint angle, communicate it back to the controller which would then computed the forward kinematics for the corresponding joint angles. The desired endeffector setpoint and corresponding robot position in X, Y and Z can be seen in Fig. 6.6. In this figure, the controller is used to compute a 25mm motion in each axis, returning to the starting position between each motion. The robot is able to accurately move between these points in a controlled motion with a mean error of 1.72 ± 2.26mm.

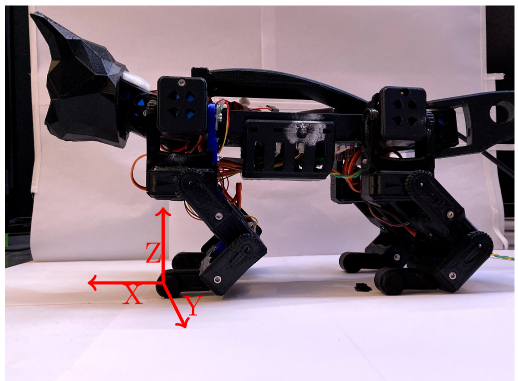

Figure 6.5: Robot with its Coordinate Frame

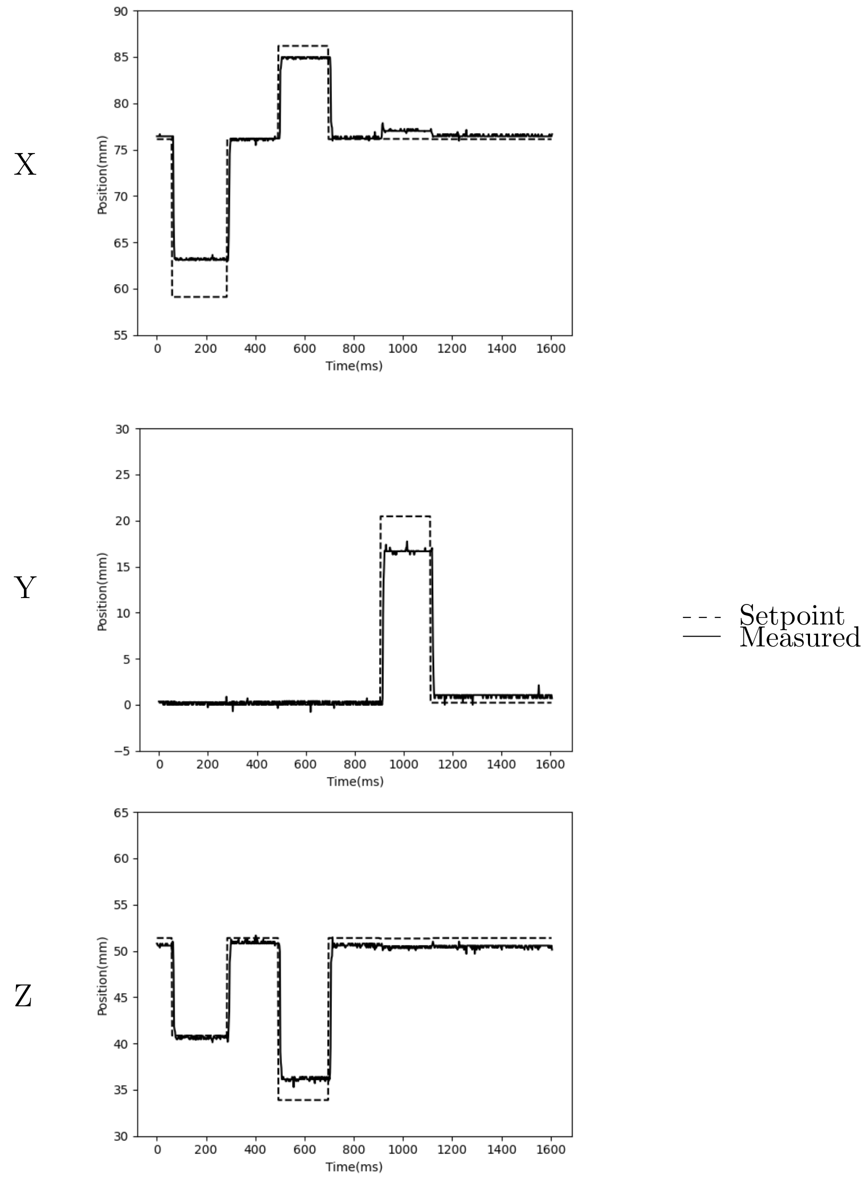

Figure 6.6: Plot Comparing Setpoint vs Endeffector- Point to Point

Following these tests, the robots ability to perform the developed gait was assessed. As before, 14Bit absolute magnetic encoders were attached to each limb and used to compute the forward kinematic of the achieved robot position. In Fig. 6.7 the computed motion in X, Y, and Z for the walking gait can be seen compared to the motion performed by the robot itself.

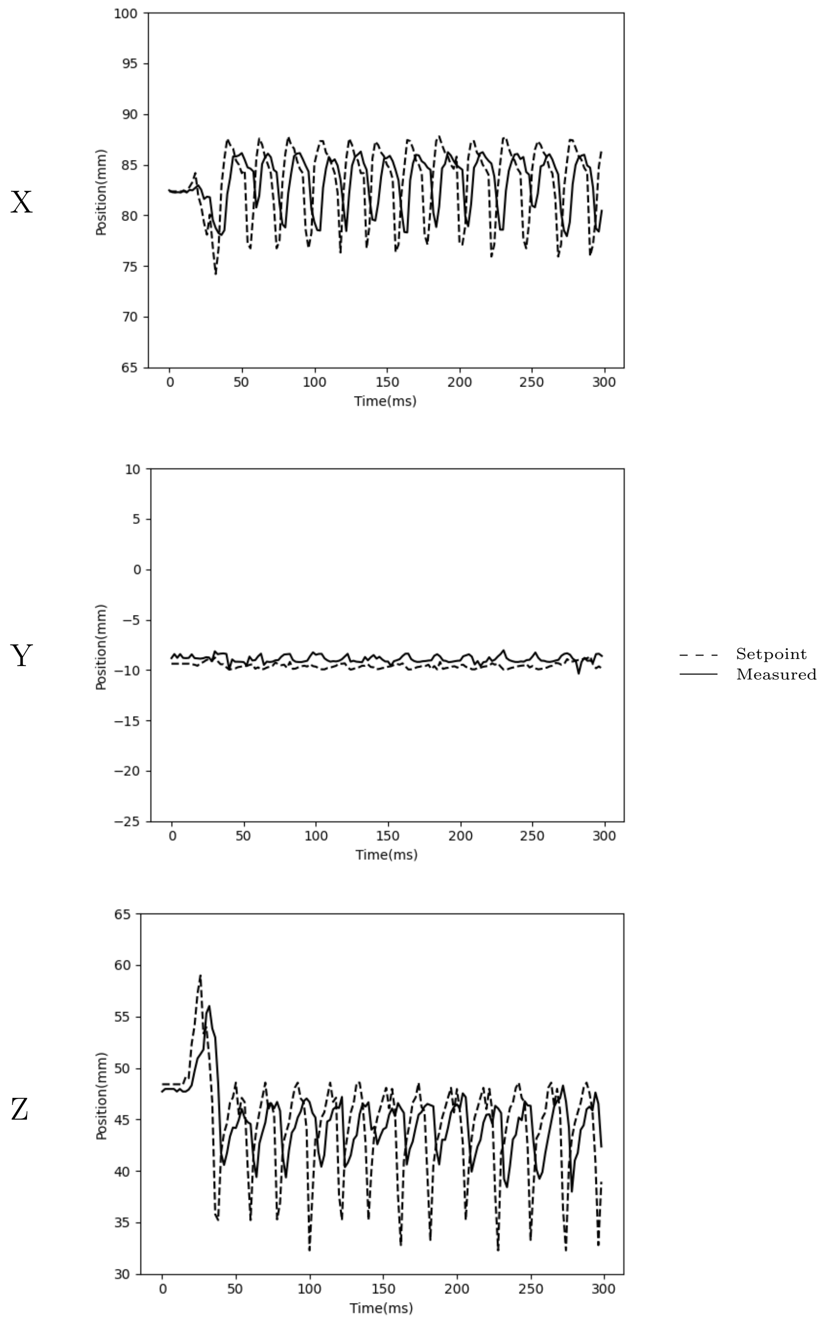

Figure 6.7: Plot Comparing Setpoint vs Endeffector- Generated Gait

Through this figure it can be seen that the root is able to accurately follow its desired gait with a mean error of 1.39±2.012mm. In the gait cycle there was a lower mean error than that observed in the positional accuracy tests, this is likely due to the increased time span of this tests in comparison to the positional accuracy test. A magnitude plot of the final gait motion and its corresponding motion on the robot can be seen Fig. 6.8.

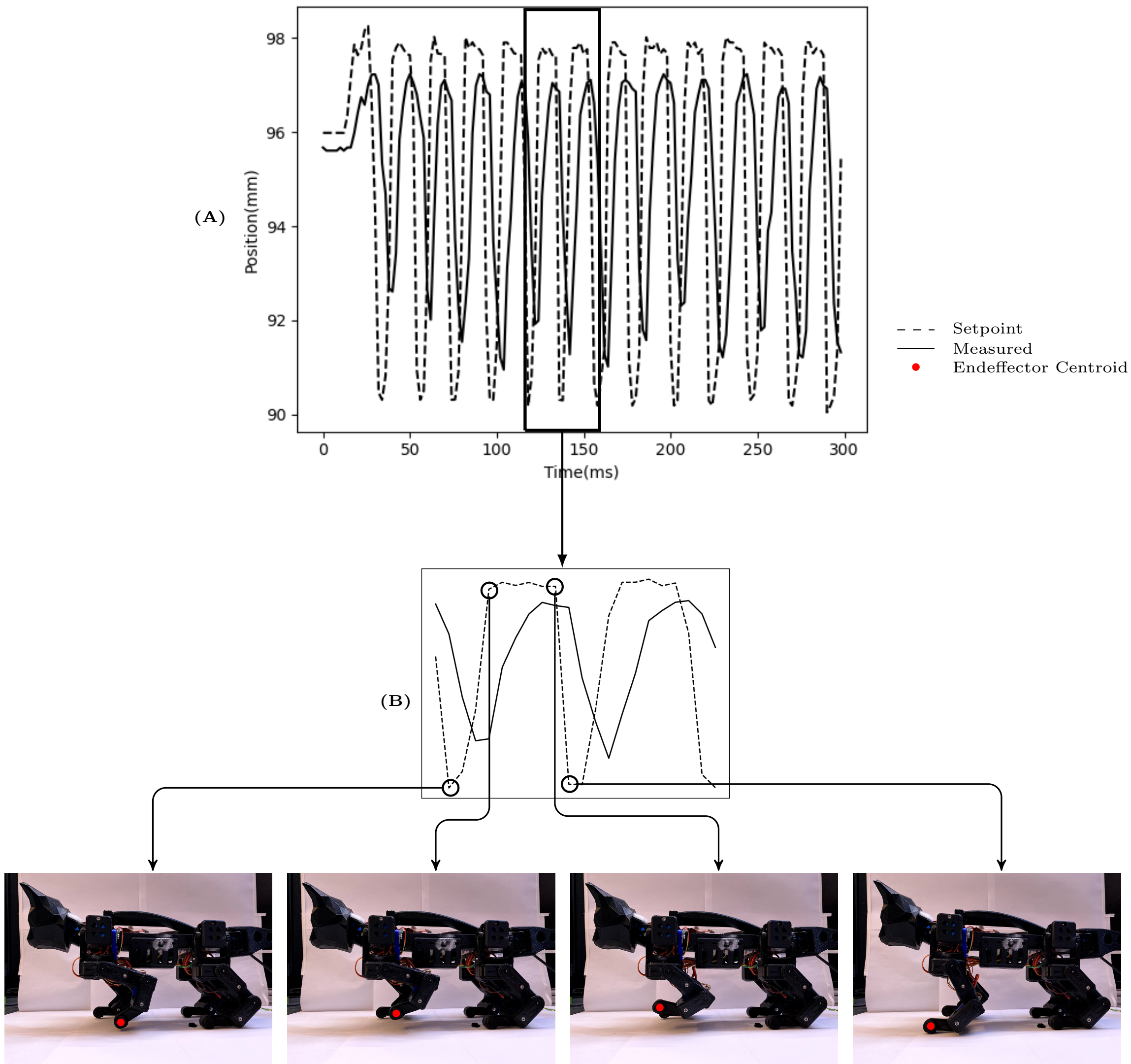

Figure 6.8: A: Gait Generation as Shown on Robot
B: Single Gait Cycle

In this figure the center of the end effector is seen highlighted in red, the set point sent to the robot is seen in the dotted line and the followed motion can be seen in the solid line. Due to the communication protocol used by the system being asynchronous, the joint angles were reported one step cycle after the setpoint is sent. This results in the delay between setpoint and motion seen in Fig. 6.8.B.

6.5 Course Implementation Results

The success of the ISP outlined in Section 3.3 was evaluated on two main aspects. The student's ability to complete the desired tasks and the student's overall opinion of the course and materials provided. The final result of each of the student's projects was evaluated and the students responded to a simple survey to provide their feedback on the course itself. In the end all 3 students were able to complete the final project, successfully developing a static walking gait with 2 students furthering the gait by integrating the head and tail of the robot to increase balance and make the system more dynamic in its walking. This speaks to both the ability of the students, the robot, and the course and lab materials developed. After completion the students were asked to fill out a brief survey based on that provided by WPI to get feedback on standard courses. This survey was comprised of 5 questions, these questions and their mean responses can be seen in Table 6.1. The responses are measured on a five-point Likert scale ranging from strongly disagree (1) to strongly agree (5). From the results provided by this survey it can be seen these three students with a variety of robotics experience had an overall very positive view of the course material that was developed. The students had an overall positive feedback of all questions other than the workload being an appropriate amount. This was primarily due to the added workload of completing the project and work single-handedly where the intended implementation of the course would have teams of 3 to 4 students working together to complete the work and project.

Survey Question	Mean response
Course material appropriately tailored to topics	4
Lab material appropriately tailored to robots abilities	4.6
Appropriate workload	3.6
Issues with the robot were addressed in a timely manner	4.6
Appropriate expectations for the final project	5

Table 6.1: ISP Feedback Survey Results (1 - Strongly Disagree, 5 - Strongly Agree)

6.6 Limitations

There are several limitations of this system, primarily due to the choice of motors used. The chosen hobby servos being the frugal option, have limited torque, slow speed, and a relatively slow update frequency of only ~330Hz. These limit the size of the platform being as large as possible at its current size only allowing for a minimal payload capability. In addition to this without further sensor integration such as encoders for each limb, there is a limit to the possible accuracy of the platform. The lack of joint feedback such as joint forces and joint position forces the assumption that all commanded motions are accurately achieved in the needed time frame by the robot. This is blind control of the end effector which may result in the robot stumbling if the target setpoint is not accurately achieved before another position is sent.

The use of an accurate joint force-sensing would allow for a much more dynamic system and gait to be developed. This would allow for a fully dynamic control model of the robot to be developed allowing for superior walking and performance in uneven terrain. In addition, an adaptation of a contact sensor to the end effector of each leg would allow for the robot to know where and ideally the direction in which contact has been made and inversely the force being applied to the limb.

The course developed came with a few limitations. Primarily, having tested with only one person teams, it is clear that the students believe that completing the assigned labs and the final project alone is too much work, however, expanding the teams to 3 to 4 students may prove to reduce the workload drastically. To refine the work several persons will need to be consulted including the students that took the class, numerous professors with experience coordinating such courses, and potentially more students willing to participate in an experimental version of the course. A further limitation to the course is the lack of much formal class content, the students were primarily given research papers with some guidance to understand and learn the topics. For a final developed course, refined lecture material would need to be devised.

With the current communication protocol operating over WiFi through the UDP protocol, packets may be lost or delayed. A higher fidelity wireless communication using either WiFi or Bluetooth may prove to be more reliable in cases that require all packets arrive and are executed in a fixed time frame. For mass production, resin casting materials have not yet been fully tested to ensure a material that is both strong enough and light enough to replace the 3D printed parts is easily available. This would not prevent the final product from being manufactured at a large scale, it would likely remain being 3D printed. Despite these limitations the platform and its supporting architecture are capable of performing all required tasks as well as perform all necessary subsections required in the teaching process.

Chapter 7

Conclusion and Future Work

This paper introduces the SmallKat robotic quadruped platform, the capabilities of the robot, and the design criteria utilized when creating the final robot. The resulting robot was capable of demonstrating several educational concepts including kinematics, trajectory planning, gait generation, controls, and dynamics. This robot will allow for institutions and research labs to develop and teach the general student body with a capable platform. This will assist in furthering the development of multi-pedal systems as a whole by providing more engineers with experience in the field.

The final platform is a combination of custom electronics, off the shelf servo motors, and custom-designed parts and software. This combination allows for a reliable yet affordable robot. With the integrated safety features and the teaching objectives, the system is ideally suited for teaching within an undergraduate course setting. With most large quadruped robots, the robot has enough force to break itself or injure a user operating it; SmallKat has enough torque to walk but has safety features integrated and utilizes motors with low enough torque that it is unable to damage itself or the end-user in any meaningful way. With this level of control the robot is still able to maintain a mean accuracy of below 3 mm error at the end effector in the XYZ workspace.

As can be seen in Section 6 the robot designed can successfully and accurately perform all tasks required in the defined teaching objectives in Section 3.2. The resulting robot and simulation developed can perform static and basic dynamic walking gaits. The overall robot was designed to meet a low-cost price point to make it feasible as a lab kit for colleges and universities to further expand their course offerings. The targeted price point for the completed system including charging, battery, and all related electronics similar to that of the TurtleBot 3 Burger being ~\$550. The final cost of the SmallKat robot developed is ~\$260 in parts and materials.

7.1 Future Work

For further development to be done on this platform, all of its downfalls must first be found. The course must be further developed to have a full-scale course of students operating in 2 to 3 person teams with their own dedicated robots. From this all of the problems experienced must be closely and carefully documented; this must be done to understand the issues experienced by novice users to find what can be improved. These problems must then be addressed and changes made to both hardware and software to address them. Further development could be performed into the addition of external sensors such as pressure-sensitive feet and adding encoders to each joint for position feedback. This could be combined with the current sensors attached to each motor to develop a dynamic model and gait to further improve the capabilities of the robot itself.

In the future, there are plans to implement a course at the graduate level in the robotics engineering department utilizing this robot to teach multipedal locomotion. This will be closely monitored by the team and support will be provided to the users to ensure as optimal of an experience as possible. All changes and work done will be updated and published to the public through GitHub, a link to which can be found in Appendix A.

Appendix A

All Files and Models

All files related to this project are hosted in the operation SmallKat repository on GitHub and can be accessed at: `https://github.com/operationSmallKat`.

Project	Contents
SmallKatLinkLoader	Example software for testing and basic walking
Luna	Development of the Luna platform including models
SmallKat-Motherboard	Finalized motherboard design and all related files
Graycat	Development of the Grace platform including models
SmallKat_V2	Release version of the Grace model

Table A.1: Projects on the OperationSmallKat Organization

Appendix B

Testing Procedure of the Motherboard

1. Plug in both USB cables for ESP32 programmers on jig
2. Plug in communication USB for onboard micro controller
3. Run Python test script (available in the GitHub organization in Appendix A)
   (a) Program controller board with test firmware
      i. Test for 3.3V regulation, i^2c communication and micro controller status
      ii. If passed program with final firmware
   (b) Program Motherboard with test firmware
   (c) Enable Battery power input
      i. Test for 3.3V regulation, 6V regulation and battery monitoring circuit
      ii. Test i^2c communication to the micro controller

iii. Test i^2c communication to the IMU

iv. Check micro controller status

v. If passed program with final firmware

4. Display a Pass/Fail status of the board, with reason of failure

Bibliography

[1] W. Bosworth, S. Kim, and N. Hogan, "The mit super mini cheetah: A small, low-cost quadrupedal robot for dynamic locomotion," in *2015 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*. IEEE, 2015, pp. 1–8.

[2] G. Bledt, M. J. Powell, B. Katz, J. Di Carlo, P. M. Wensing, and S. Kim, "Mit cheetah 3: Design and control of a robust, dynamic quadruped robot," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2018, pp. 2245–2252.

[3] C. Semini, N. Tsagarakis, E. Guglielmino, M. Focchi, F. Cannella, and D. Caldwell, "Design of hyq -a hydraulically and electrically actuated quadruped robot," *Proceedings of the Institution of Mechanical Engineers. Part I: Journal of Systems and Control Engineering*, vol. 225, pp. 831–849, 08 2011.

[4] B. Katz, J. D. Carlo, and S. Kim, "Mini cheetah: A platform for pushing the limits of dynamic quadruped control," in *2019 International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 6295–6301.

[5] C. Semini, "Hyq - design and development of a hydraulically actuated quadruped robot," 2010.

[6] D. Dholakiya, S. Bhattacharya, A. Gunalan, A. Singla, S. Bhatnagar, B. Amrutur, A. Ghosal, and S. Kolathaya, "Design, development and experimental realization of a quadrupedal research platform: Stoch," in *2019 5th International Conference on Control, Automation and Robotics (ICCAR)*, April 2019, pp. 229–234.

[7] S. Lohmann, J. Yosinski, E. Gold, J. Clune, J. Blum, and H. Lipson, "Aracna: An open-source quadruped platform for evolutionary robotics," 07 2012, pp. 387–392.

[8] A. Shkolnik, M. Levashov, S. Itani, and R. Tedrake, "Motion planning for bounding on rough terrain with the littledog robot," in *Proceedings of 2010 IEEE, International Conference on Robotics & Automation*, 2010, pp. 134–142.

[9] J.-W. Chung, I.-W. Park, and J.-H. Oh, "On the design and development of a quadruped robot platform," *Advanced Robotics*, vol. 24, pp. 277–298, 01 2010.

[10] Y. Geva and A. Shapiro, "A novel design of a quadruped robot for research purposes," 2014.

[11] M. Raibert, K. Blankespoor, G. Nelson, and R. Playter, "Bigdog, the rough-terrain quadruped robot," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 10 822–10 825, 2008.

[12] M. P. Murphy, A. Saunders, C. Moreira, A. A. Rizzi, and M. Raibert, "The littledog robot," *The International Journal of Robotics Research*, vol. 30, no. 2, pp. 145–149, 2011. [Online]. Available: https://doi.org/10.1177/0278364910387457

[13] N. Yadav, T. Raina, N. Gupta, and N. Shrivas, "Design and development of a structurally stable quadruped robot for surveillance," vol. 2148, 09 2019, p. 030028.

[14] "Spot." [Online]. Available: https://www.bostondynamics.com/spot

[15] S. Seok, A. Wang, Meng Yee Chuah, D. Otten, J. Lang, and S. Kim, "Design principles for highly efficient quadrupeds and implementation on the mit cheetah robot," in *2013 IEEE International Conference on Robotics and Automation*, 2013, pp. 3307–3312.

[16] H.-W. Park and S. Kim, "Quadrupedal galloping control for a wide range of speed via vertical impulse scaling," *Bioinspiration & Biomimetics*, vol. 10, no. 2, p. 025003, mar 2015. [Online]. Available: https://doi.org/10.1088%2F1748-3190%2F10%2F2%2F025003

[17] A. Spröwitz, A. Tuleu, M. Vespignani, M. Ajallooeian, E. Badri, and A. J. Ijspeert, "Towards dynamic trot gait locomotion: Design, control, and experiments with cheetah-cub, a compliant quadruped robot," *The International Journal of Robotics Research*, vol. 32, no. 8, pp. 932–950, 2013.

[18] S. Nakatsu, A. Rosendo, K. Narioka, and K. Hosoda, "Stable reflex-based walking of forelimbs of a bio-inspired quadruped robot-modeled cheetah," in *2013 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2013, pp. 1813–1818.

[19] M. Hutter, C. Gehring, D. Jud, A. Lauber, C. D. Bellicoso, V. Tsounis, J. Hwangbo, K. Bodie, P. Fankhauser, M. Bloesch *et al.*, "Anymal-a highly mobile and dynamic quadrupedal robot," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 38–44.

[20] J. Denavit and R. Hartenberg, *A Kinematic Notation for Lower-pair Mechanisms Based on Matrices.* ASME, 1955. [Online]. Available: https://books.google.com/books?id=nCDOoQEACAAJ

[21] M. W. Spong, *Robot Dynamics and Control*, 1st ed. USA: John Wiley & Sons, Inc., 1989.

[22] J. Clune, B. E. Beckmann, C. Ofria, and R. T. Pennock, "Evolving coordinated quadruped gaits with the hyperneat generative encoding," in *2009 IEEE Congress on Evolutionary Computation*, 2009, pp. 2764–2771.

[23] "Usb charger (usb power delivery)." [Online]. Available: https://www.usb.org/usb-charger-pd

[24] "Tiobe index." [Online]. Available: https://www.tiobe.com/tiobe-index/

[25] "Bowler studio." [Online]. Available: http://commonwealthrobotics.com/

[26] E. Coumans, "Bullet physics simulation," in *ACM SIGGRAPH 2015 Courses*, ser. SIGGRAPH '15. New York, NY, USA: ACM, 2015. [Online]. Available: http://doi.acm.org/10.1145/2776880.2792704