TURTLE Quadruped Project (QUAD)

TEXAS A&M UNIVERSITY Engineering

Project Lead: Ian Lansdowne

Members: Zachary Bucknor-Smartt, Kalen Jaroszewski, Aarathi Devakumar, Oscar Dominguez, Jonathon Foltyn, Micah Guttman, Shreyas Kannanganat, Jevine Kong, et al.

TEXAS A&M UNIVERSITY
ROBOTICS TEAM & LEADERSHIP EXPERIENCE

Problem Definition

Legged robots are more agile and adaptable than robots with other forms of locomotion. The purpose of this project is to design and manufacture a robot capable of balancing and maneuvering on four legs to better understand the capabilities of walking robots.

Approach

Robot dogs are not a new concept by any means. This project has drawn heavy inspiration from other quadrapedic robots such as Boston Dynamics' Spot, MIT's Mini Cheetah, James Bruton's openDog V3, as well as a prototype by the project lead. To create a working robot, the team started with robot dimensions similar to the Mini Cheetah, known as QUAD V1. Upon realizing the difficulty of testing on a large and expensive robot, a smaller version was created in order to serve as a software testbed. Most recently, V2 was completed taking the shortcomings from V1 and the framework of mini and improved upon them. A cycloidal drive was also designed and tested to be used in a future iteration.

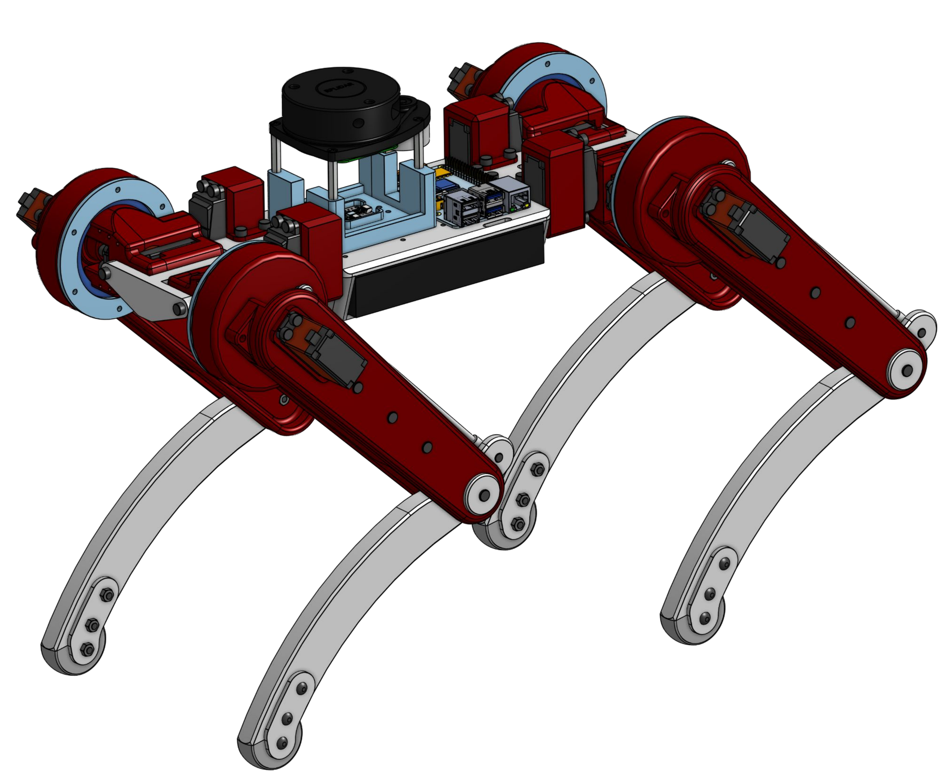

Figure 1. CAD model of QUAD V1.

Hardware

QUAD V1

Version 1 of the quadruped has 3 main joints on each leg: a knee pivot and 2 shoulder pivots. QUAD V1 uses a tie rod to actuate the lower leg. The tie rod was chosen to allow the lower leg actuator to reside in the upper leg lowering the inertia of the leg. Silicone molded feet were chosen to increase grip and prevent slipping on smooth surfaces. QUAD V1 uses a Raspberry Pi 4 as its processor and an offboard PWM driver to control all leg motors. Additionally, an IMU and LiDAR sensor were added to experiment with mapping, balancing on uneven terrains, and orientation in different environments.

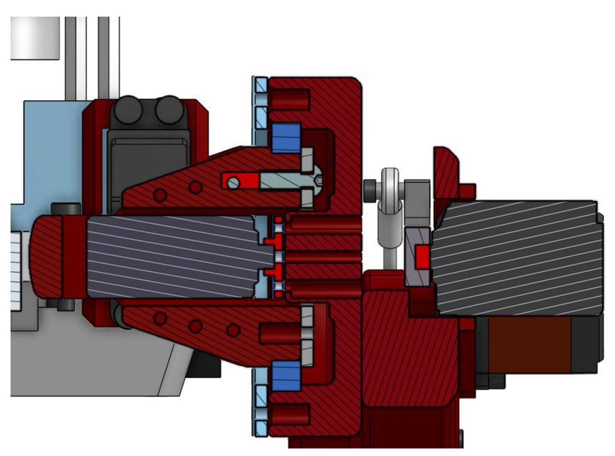

Figure 2. QUAD V1 Shoulder Cross Section.

Mini QUAD V1

Mini quad tackles the challenge of starting a miniature line of dogs in parallel to the standard QUAD series motivated by a need to have a dog that is lower maintenance than a standard QUAD as well as to serve as a software testbed.

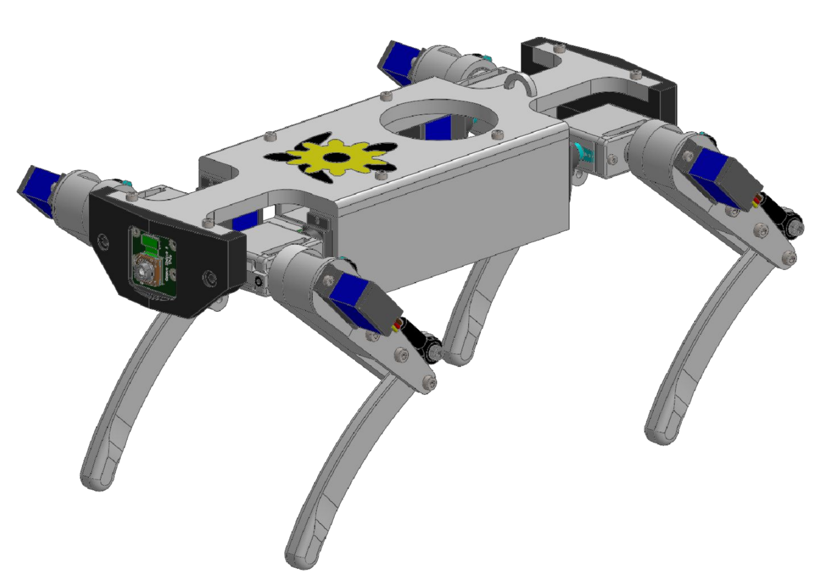

Figure 3. CAD model of Mini QUAD V1.

QUAD V2

Based on the results of the V1 and Mini QUAD iterations, several improvements are being considered for the QUAD V2. Ensuring the robot can support more than its own weight will be the main priority, so the maximum torque output will be improved by using a 19:1 cycloidal gear reduction and replacing servo motors with powerful brushless DC motors.

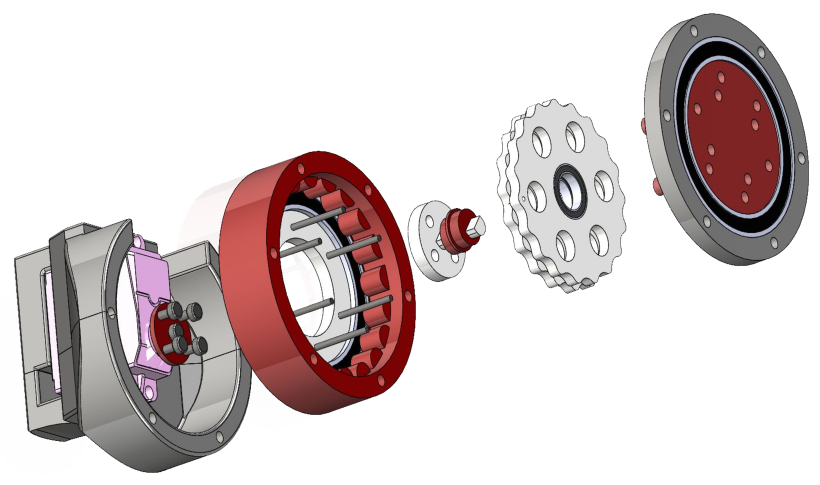

Figure 4. Exploded View of Cycloidal Actuator CAD model.

Software/Modelling

Inverse Kinematics

Software for all of the quadrupedal robots was standardized as a single package for compatibility across the different robots with varying parameters. Since both existing robots use position-controlled servos to actuate the legs, the angle of each joint was calculated to move the leg to a desired position, $\mathbf{p}$. The formulae for the joint angles α (shoulder), β (upper leg), and γ (lower legs) is shown in figure 5. The constants l_s, l_u, l_l, r_f are the lengths of the shoulder, upper leg, and lower leg and foot radius. Each angle is mapped to a pwm duty cycle through a calibration routine unique for each robot. All settings are saved in a JSON configuration file that is read on start. The trajectory of the foot used for walking is created by interpolating between a resting, lifted, and forward stepping state to emulate walking. Each leg is shifted a quarter-period out of phase for multiple legs to touch the ground at once.

Teleoperated Control

To command the robots remotely, a driver station software was built. The software was made in Python, and reads joystick angles and sends them over UDP to the robot. The custom driver station software allows for added functionality beyond driving the robot.

V2 Motor Software

The motor planned to be used for version 2 is the MJBOTS Moteus brushless motor. These were calibrated and programmed with Python through interfacing with the current software environment. Currently, work is being done to make sure that the motor works with the Python API.

Future Goals

The key future plans for QUAD revolve around the V2 leg modeling. The highest priority goal is to complete a prototype leg and test it. In order to test the leg, a jumping stand will also be made along with the leg in order to learn how much torque and force the motors will need to make the leg stand on its own as well as perform a jump.

The next generation of TURTLE's Quadruped robot will have sensors on every joint to allow for development of a feedback-driven control loop. With a control loop, the robot can be more responsive to its environment, which allows it to conquer rougher terrain and interact safely with humans. With greater agility and strength, QUAD V2 will act as a more practical testbed and be able to perform more meaningful tasks.

$$\mathbf{p} = x\hat{\mathbf{i}} + y\hat{\mathbf{j}} + (z + r_f)\hat{\mathbf{k}}$$

$$\alpha = \arctan2(z, y) - \arccos\left(\frac{l_s}{\sqrt{y^2+z^2}}\right)$$

$$\beta = \arctan2(\sqrt{y^2 + z^2 - l_s^2}, x) - \arccos\left(\frac{l_u^2+x^2+z^2-l_l^2}{2l_u\sqrt{x^2+y^2+z^2-l_s^2}}\right)$$

$$\gamma = \arccos\left(\frac{l_s^2+l_u^2+l_l^2-x^2-y^2-z^2}{2l_u l_l}\right)$$

Figure 5. Joint Angle Equations.