

Article

Design and Control of a Open-Source, Low Cost, 3D Printed Dynamic Quadruped Robot

Joonyoung Kim , Taewoong Kang , Dongwoon Song  and Seung-Joon Yi *

Department of Electrical Engineering, Pusan National University, Busan 46241, Korea;
kjkjy98@pusan.ac.kr (J.K.); touy1@pusan.ac.kr (T.K.); vehddns@pusan.ac.kr (D.S.)

* Correspondence: seungjoon.yi@pusan.ac.kr; Tel.: +82-51-510-7917

Abstract: In this paper, we present a new open source dynamic quadruped robot, PADWQ (pronounced pa-dook), which features 12 torque controlled quasi direct drive joints with high control bandwidth, as well as onboard depth sensor and GPU-equipped computer that allows for a highly dynamic locomotion over uncertain terrains. In contrast to other dynamic quadruped robots based on custom actuator and machined metal structural parts, the PADWQ is entirely built from off the shelf components and standard 3D printed plastic structural parts, which allows for a rapid distribution and duplication without the need for advanced machining process. To make sure that the plastic structural parts can withstand the stress of dynamic locomotion, we performed finite element analysis (FEA) on leg structural parts as well as a continuous walking test using the physical robot, both of which the robot has passed successfully. We hope this work to help a wide range of researchers and engineers that need an affordable, highly capable and easily customizable quadruped robot.

Keywords: dynamic quadruped robot; 3D printing; open source hardware



Citation: Kim, J.; Kang, T.; Song, D.; Yi, S.-J. Design and Control of a Open-Source, Low Cost, 3D Printed Dynamic Quadruped Robot. *Appl. Sci.* **2021**, *11*, 3762. <https://doi.org/10.3390/app11093762>

Academic Editor: Richard (Chunhui) Yang

Received: 27 March 2021

Accepted: 20 April 2021

Published: 22 April 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

As we can see from people and animals, legged locomotion has the potential to handle rough terrain better than any other locomotion types. However, legged locomotion requires complex leg mechanism, high power density actuators and advanced control methods, which have prevented the widespread use of legged robots for a long time. Introduction of the Big Dog quadruped [1] has proved that a quadruped robot with a true rough terrain traversability is indeed possible, but the high power density required for the robot could be only realized by hydraulic actuators powered by a two-stroke gasoline engine, which are not preferable for smaller robots and indoor applications. Although there have been approaches to scale down the hydraulically actuated quadruped [2], hydraulic actuation has not been widely adopted for legged robots yet due to its complexity, weight, cost and maintenance issues.

Electric motors, although simpler, lighter and more convenient to use compared to the hydraulic actuators, have lower power density and torque and is not suitable for directly driving joints of legged robots in general. For bipedal robots powered by electric motors, the conventional actuation method uses high-speed, low-torque motors paired with high ratio gear reducers such as the Harmonic drive to generate the high output torque, which are controlled by a high-gain PD control method to minimize the positional tracking error. This actuation method provides high torque density and low positional tracking error, both of which are crucial for bipedal robots to achieve static stability with small footprint, but has limited backdrivability, high inertia and low maximum joint velocity due to the high reduction ratio and not ideal for walking over uneven terrain without a precise model of the surface where compliance, backdrivability and torque controllability of the joints are needed. Some approaches, such as in [3], uses force torque sensors to provide high bandwidth close loop torque control and active electrical compliance, but such setup is still limited by the high inertia and low transparency due to the high ratio gear reducer.

One popular method to provide backdrivability and compliance for the joint is the series elastic actuators (SEAs), where an elastic element is connected in series with the energy source to provide the system a passive compliance and impact robustness. Additional torque and deflection sensors can be used for a close-loop control of the output torque and position. ANYbotics ANYmal [4] uses modular SEA actuators for its joints, and has showed the capability to operate over a wide range of unstructured environments. However SEA has several disadvantages as well: they are relatively expensive, heavy and bulky due to the complex design, has fixed stiffness determined by the elastic element and limited torque control bandwidth.

A more direct method, called the quasi-direct drive (QDD) [5], uses a high-torque motor with a large stator radius, paired with gear reducer with relative low gear ratio, typically planetary gear or a multi bar linkage [6] without an elastic component. Such a system is highly backdrivable due to the low gear ratio, and has a high torque control bandwidth that can be used to replicate a user-designed nonlinear stiffness and damping profiles, which is beyond the capability of SEAs. QDD has been successfully used for several medium sized quadrupeds such as MIT Cheetah series [7] and ALPHRED [8], showing excellent maneuverability comparable to hydraulic powered quadrupeds, and terrain-blind walking over uneven terrain and even stairs using proprioceptive feedback alone.

Recently, the big commercial success of quadrotors have made high performance brushless motors, micro motor controllers and advanced Lithium batteries commercially available as affordable off-the-shelf components. Leveraging these recent advances, an modular low-cost QDD actuator based on off-the-shelf parts have been suggested in [9], and several small sized dynamic quadrupeds using modular QDD actuators have been suggested, which includes Unitree A1 [10], Jueying Mini [11] and many others. Another recent advance is the widespread adoption of affordable 3D printers, which have allowed design, manufacture and open sharing of complex 3D parts without using costly CNC machining or casting process. With help of the 3D printing technology, several open-source robotic platforms have been suggested, which aim to allow researchers test and develop software algorithms on common hardware platforms that can be built using commercially available components and easy to produce parts. However, to our knowledge, most of those open source robotic platforms, which includes Charlotte [12], RealAnt [13] and Lilibot [14], are based on position controlled servo motors that greatly limits their size, performance and the usage of advanced control algorithms. Two notable exceptions, the Stanford Doggo [15] and Solo [16], use custom QDD actuators that allows for advanced control methods, but they are lightweight, low DOF quadrupeds mainly designed for locomotion research, which largely limits their real-world application.

In this work, we present an open-source dynamic quadruped robot platform that is entirely built from commercial, off-the-shelf components and standard 3D printed plastic structural parts, to ensure the robot can be easily duplicated and distributed without relying on advanced machining processes. All the structural components of the robot are designed to be manufactured by a standard, low cost plastic 3D printing process, and their strength are validated by both the finite element analysis and actual dynamic locomotion testing with the assembled robot. All the actuators, onboard computers and communication adapter modules are commercial, off-the-shelf products that can be directly purchasable from open markets. We used a simple, position control-based locomotion controller with a high-bandwidth low level torque controller to test the basic capability of the robot, and have found that the robot is capable of dynamic trot-walking at the velocity of 1.0 m per second, with an expected continuous runtime of one hour using internal batteries.

The remainder of the paper proceeds as follows. Section 2 introduces the hardware design of the robot in detail, including the actuator selection, actuator test results, leg and torso design and the FEA result of the leg parts. Section 3 presents the hierarchical motion controller we used for the initial testing of the robot, which can generate several different gait patterns for the robot. Section 4 presents the test results of the suggested motion controller acquired in a physics realistic simulation, and Section 5 shows the experimental

results using the physical robot. Finally, we conclude with a discussion of potential future directions arising from this work.

2. Hardware Design

The overall design and actuation approach for the PADWQ is largely based on recent QDD quadrupeds, especially the MIT Mini Cheetah [9]. Each 3 DOF leg consists of three serial rotational joints, which are controlled by modular actuators placed together at the hip joint to minimize the leg inertia, and a spherical foot that makes a point contact to the ground. Front and rear leg assemblies are joined by the torso, which has rounded edges to prevent the self collision with the limbs. As shown in Figure 1, PADWQ has leg link length of 0.22 m for the upper and lower links, and is 0.419 m long between the front and rear leg pitch joints, and 0.332 m wide, and 0.332 m high, and weighs 12.7 kg with two batteries. Compared to the size and weight of the Mini Cheetah, it is roughly 10 percent larger in every dimension and 5 percent more dense.

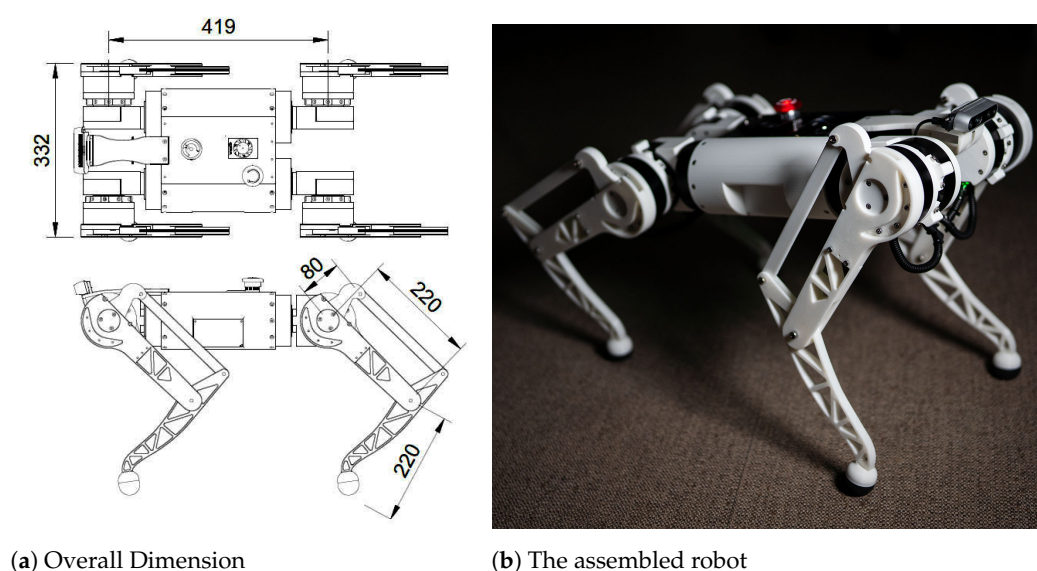







Figure 1. The PADWQ robot.

2.1. Modular Actuators

Since the introduction of the MIT Mini Cheetah and its modular actuator, several commercial actuators with similar design principle have been released to the market. They all share relatively large diameter stator for high torque output, single stage planetary reduction gear at the center cavity, and integrated encoder and driver board. We summarize the commercially available QDD actuators we considered for the robot and their properties in Table 1. They all share several similarities to the MIT Cheetah actuator, including roughly the same dimension, daisy-chainable bus interface with 1 M bps baudrate, and position and torque controllability using built-in motor controller and encoder. Among the four commercial QDD actuator brands, which includes Steadywin, T-motor, Gyems and Unitree, we chose the Gyems RMD-X8 series actuators as they were widely available in large quantities, relatively well documented and supported, and most affordable at the time of writing this paper. RMD motor shares the same 6:1 gear reduction ratio and CAN bus connectivity with the MIT mini cheetah actuator, and has additional functionalities inherited from their previous actuators, which includes hardware DIP switch that can be used to set motor ID and CAN bus termination resistor manually, and a dedicated UART port for motor diagnosis and setup. In terms of peak torque and weight, the RMD-X8 actuator has 23 percent higher peak torque (21 Nm), yet 29 percent heavier, and slightly thicker and heavier RMD-X8 Pro actuator has 105 percent higher peak torque (35 Nm) while 56 percent heavier. We think the most of the weight penalty comes from the fully

enclosed outer shell of RMD actuators, which on the hand allows the actuators to have IP54 ingress protection rating against dust and water splash, which should greatly help the robot operating outdoors. Overall, both of the Gyems actuators compare very favorably to the Mini Cheetah actuator on paper, especially considering their retail prices are only slightly higher than the bill of material (BOM) of the Mini Cheetah actuator.

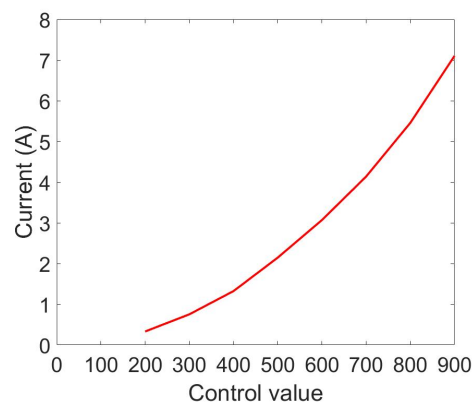
Table 1. Comparison of modular quasi-direct drive actuators.

					
	MIT Mini Cheetah Actuator	Steadywin GIM8008	T-motor AK80-9	Gyems RMD-X8	Unitree A1 Actuator
Weight (g)	480	600	485	620	605
Gear ratio	6:1	6:1	9:1	6:1	unknown
Nomial Torque (Nm)	6.9	7.6	9	9	unknown
Peak Torque (Nm)	17	15	18	21	33.5
Joint Velocity (rad/s)	40	34	25.6	25.1	21
Interface	CAN	CAN	CAN	CAN/RS485	RS485
Market Price (USD)	\$300 (BOM)	\$359	\$579	\$359	\$999

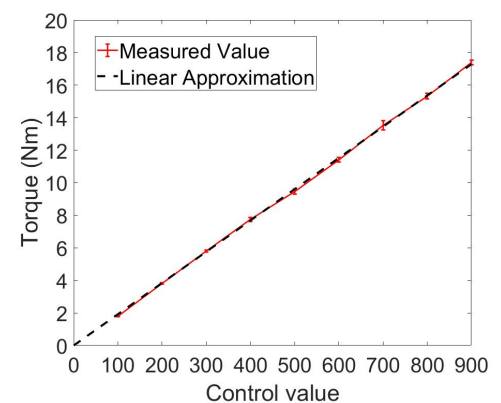
To check the actual capability of the RMD-X8 series actuators, we ran through several tests using the RMD-X8 and RMD-X8 Pro actuators. Although the actuator officially supports “torque current control” according to the provided control protocol sheet, it is not clear whether the control input is target current or target torque value, and its coefficient is not provided either. Thus, we first measured the relationship between control value, input current and output torque of the actuator using a torque sensor and current meter, as shown in Figure 2. Output torque is measured by a BLUETEC BEM-60S3 torque wrench which has maximum operation range of 60 Nm and resolution of 0.01 Nm, and input current is measured by a HTRC inline power analyzer with 0.01 A resolution. For each measurement, the actuator control value is gradually increased to the target value for 2 s and the output torque and input current are measured after they converges to stationary values. 10 tests are performed for each control values to find the average and standard deviation of the measured values. The test results with the RMD-X8 and RMD-X8 Pro motors with an input voltage of 24 V are shown in Figures 3 and 4, which clearly shows that the control signal of “torque current control” is not proportional to the input current, but to the output torque. Surprisingly, two different actuators show different slopes for control value—output torque graphs. We used least square regression to find two different slopes values, which are 52.0639 for RMD-X8 actuator and 43.721 for RMD-X8 Pro actuator. The input current is fairly consistent over multiple trials, showing very low variations over multiple trials. On the other hand, the torque output has small but measurable standard deviation, which we think is mainly due to the cogging of the motor, which are presented as vertical bars in Figures 3b and 4b. The RMD-X8 actuator shows fairly high linearity between the control value and output torque, which has maximum error of less than 1 percent when the control value is larger than 300. The RMD-X8 Pro actuator also shows linearity but with slightly higher error, which goes up to 4 percent.



Figure 2. Actuator test setup.

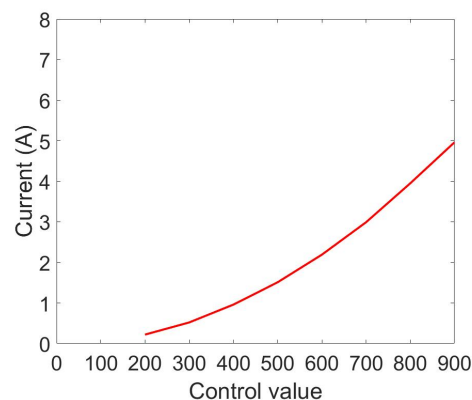


(a) Input Current of RMD-X8 actuator

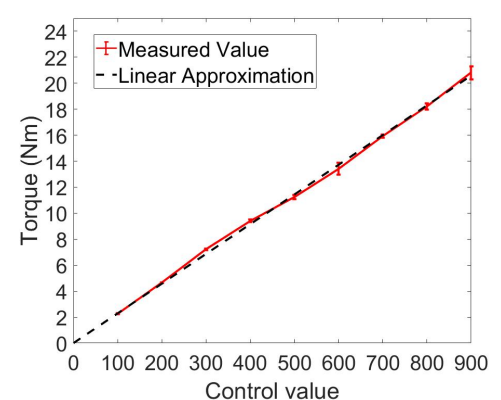


(b) Output Torque of RMD-X8 actuator

Figure 3. Input current and output torque over torque control value for RMD-X8 actuator.



(a) Input Current of RMD-X8 *Pro* actuator



(b) Output Torque of RMD-X8 *Pro* actuator

Figure 4. Input current and output torque over torque control value for RMD-X8 *Pro* actuator.

When we compare the current and torque graphs between two actuators, we see that the input current of RMD-X8 *Pro* actuator is 70.6 percent of that of RMD-X8 actuator on average with the same control value, and the output torque of RMD-X8 *Pro* actuator is 19 percent higher than that of RMD-X8 actuator on average with the same control value. Combining both results give us that the RMD-X8 *Pro* actuator only requires 59.3 percent of input current to generate the same output torque compared to the RMD-X8 actuator. This efficiency difference also affects the thermal characteristics of two actuators. We measured the internal temperature of both actuators using the built in thermal sensor, while letting the actuators generate the same output torque. Figure 5 shows the thermal test results of two actuators, with 5 Nm and 9 Nm of output torque. We found that after the testing period of 420 s, RMD-X8 actuator shows approximately twice the temperature rise compared to

RMD-X8 Pro for both cases; 29 vs. 14 degree Celsius for the 5 Nm case, and 67 vs. 34 degree Celsius for the 9 Nm case.

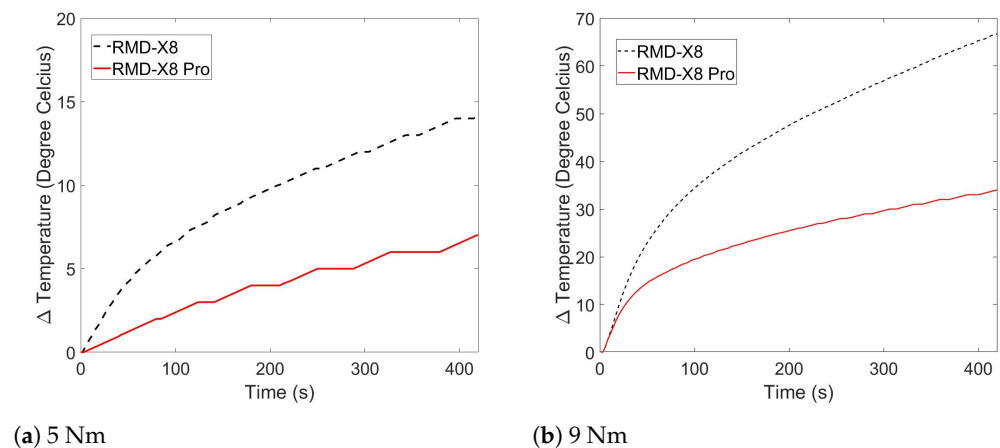


Figure 5. Internal temperature over time for X8 and X8 Pro actuators.

2.2. Leg Design

The main design goals of our 3 degrees of freedom leg is to minimize the limb mass and inertia for highly dynamic locomotion, while leveraging standard 3D printing manufacturing process. Three modular actuators, each controlling hip roll, hip pitch and knee pitch joint respectively, are serially attached to the body without additional support structure holding the joint on the other side. To minimize the leg inertia, the hip pitch and knee pitch actuators are located coaxially right next to the hip roll actuator. To actuate the knee joint from remote actuator, various mechanisms have been used, which includes linkage [11,17,18], timing belt [9,19] and roller chain [20]. Timing belt and roller chain driven knee designs have two main advantages over linkage design; additional gear reduction can be easily added by using different sized pulleys or cogs, and knee joint range of motion is not limited by the linkage. As the knee joints are under the most load for legged robots, it is a common design principle to use additional gear reduction for them if the same actuators are used for both the hip pitch and knee pitch joints. In addition, the additional knee range of motion allows the robot to change the leg configurations between the knee-forward and knee-backward positions, which is not crucial for general locomotions but can be helpful for some special situations, such as climbing high stairs or using one of the limbs for manipulation.

We used a four bar parallelogram linkage for knee actuation despite those advantages, as the linkage mechanism is much simpler and can be fully 3D printed. Compared to other recent quadruped leg designs that use coupler link fully housed inside the upper leg link [21], we use a longer crank/rocker links and external coupler link to lower the peak stress applied to the plastic components under load. To cope with higher load of the knee joint without using additional gear reduction mechanism, we use stronger RMD-X8 Pro actuator with 35 Nm peak torque for the knee joint, and lighter RMD-X8 actuator with 21 Nm peak torque for hip roll and hip pitch joints. To further reduce the peak stress applied to the leg parts, we used hollow squash balls without solid cores as the foot components. After testing multiple squash ball types, we found that the pro grade squash ball, which has the lowest bounce among squash ball types, provides adequate damping and shock absorption properties for our robot. Figure 6 shows the details of our leg design.

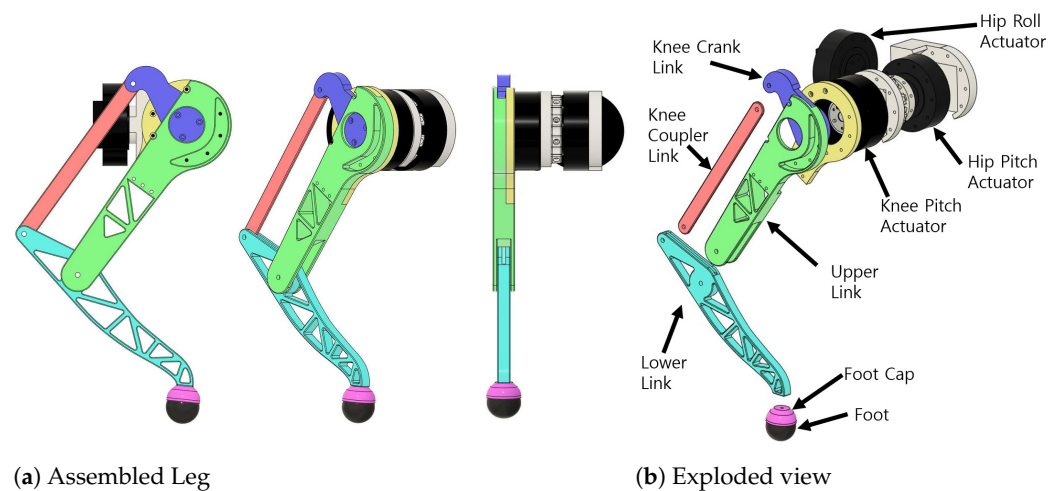


Figure 6. 3DOF Leg with a parallelogram linkage.

In addition to the knee joint angle limit posed by the linkage mechanism, we installed mechanical joint stops for the hip roll and pitch joints to prevent possibly catastrophic cable damage due to the over-rotation of the joints. The resulting range of motion for each joint is -30° to 30° for hip roll joints, -51° to 93° for hip pitch joints, and -134° to -7° for knee joints, which are shown in Figure 7. With the link length of 0.22 meters for both upper and lower links, each leg can produce 210 N (1.68 bodyweights) peak and 80 N (0.64 bodyweights) continuous vertical force in the fully collapsed configuration, which are only slightly less than Mini Cheetah's values of 1.7 and 0.7 bodyweights, respectively.

Finally, we ran finite element analysis (FEA) simulations of the individual leg parts, assuming that the leg is in standing posture and is applied 630 N (5 bodyweights) of downward force, which is equivalent to 20 times the static downward force the robot receives when the robot is standing still. The material properties used for the analysis is shown in Table 2. Based on the simulation result, which is visually shown in Figure 8, we found that the calculated minimum safety factor to be 1.31, which means that the leg parts can withstand approximately 6.5 bodyweights before failure, despite being made of plastic material.

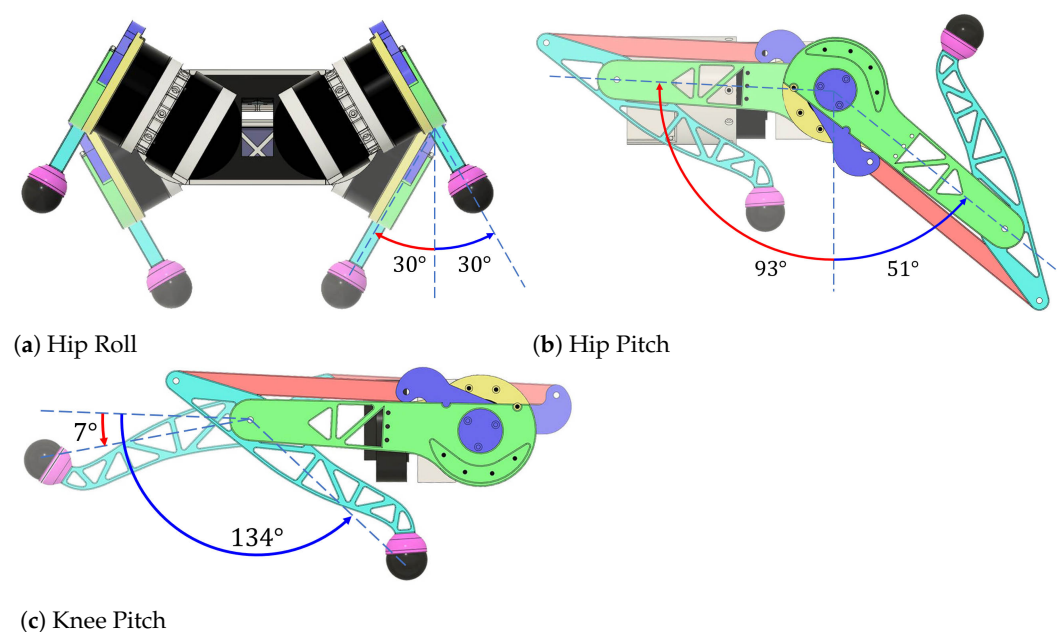


Figure 7. Joint ranges of three leg joints.

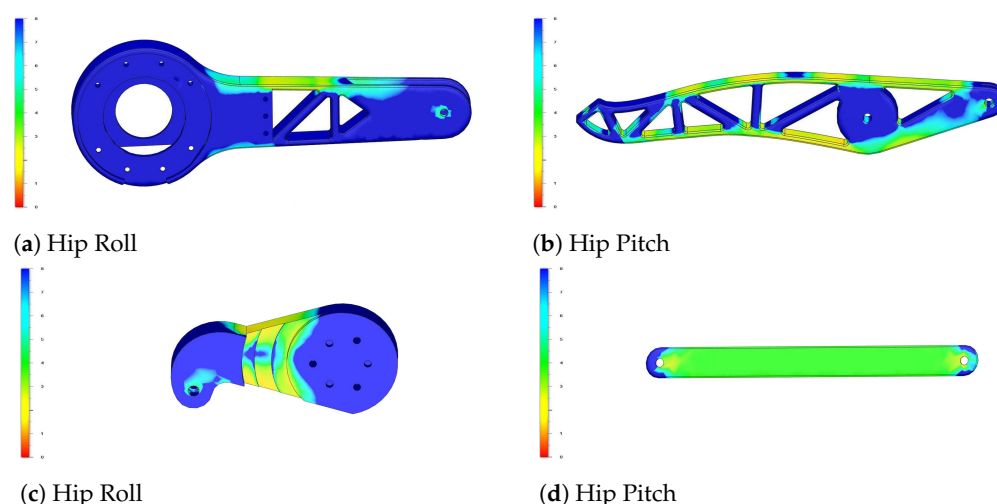


Figure 8. Finite Element Analysis results of leg parts.

Table 2. Material Properties of the PLA plastic used for structural parts.

Density	1.2 g/cm ³
Tensile Yield Strength	65 Mpa
Flexural Strength	75 Mpa
Flexural Modulus	2.102 G/pa
Impact Strength	8.5 KJ/m ³

2.3. Torso Design

Torso houses various electronics components that includes batteries, control computers, IMU and wireless receiver, which are shown in Figure 9. We use two separate single board computers (SBCs) to handle the motion control and perception tasks separately. Motion control is handled by a Lattepanda Alpha single board computer (SBC) with an Intel M3-8100Y CPU, 8 GB memory and 256 GB NVMe storage. No custom power or communication board are used. For communication to the actuators, we use commercially available CANable open-source USB to CAN adapters [22], flashed with candlelight firmware that implements gs_usb protocol, for the low level interface. A total of four CAN adapters are used, one per each leg chain, which allows the close loop torque control of all actuators at the speed of 1 kHz. In addition, the motion control computer is connected to a WIT motion WT901C MEMS IMU with 200 hz data rate for robot posture feedback, and Microsoft XBOX 360 wireless receiver for human control input. Perception is handled by a NVidia Xavier Jetson NX SBC with 8 GB memory and 256 GB NVMe storage, which has 384 CUDA cores that can run computationally intensive deep learning-based perception algorithms which are beyond the capability of CPUs. Intel RealSense D435 RGBD camera is mounted at the front of the robot to provide RGB and depth information to the Jetson NX SBC. We plan to run object and human pose detection algorithms using the data feeds from onboard RealSense RGBD camera to provide situational awareness required for rough terrain traversal and long term autonomy of the robot.

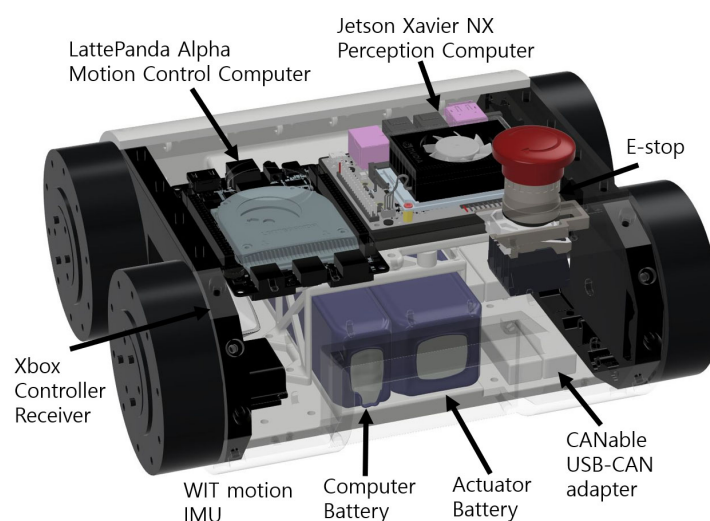


Figure 9. Torso components.

To power the actuators and computers, we have two separate batteries in the battery compartment located at the bottom center of the torso, which doubles as the carrying handle of the robot. Two single board computers are powered by a 3-cell 5200 mAh LiPo battery with 57.72 Wh of capacity. As the measured peak power consumption of LattePanda Alpha computer is roughly 20 W under full load, and the Jetson Xavier NX 15 W, we expect the computer batteries to last approximately one hour and 40 min in the worst case. Leg actuators are powered by a 6-cell 4200 mAh LiPo battery with 93.24 Wh of capacity, which are directly fed into leg actuators using splitter cables.

2.4. 3D Printing of Parts

We used two hobbyist grade 3D printers, QIDI X-Max and FlashForge Creator 3, to print all the structural parts, as shown in Figure 10. Common, low-cost PLA material is used for all parts, whose material properties are summarized in Table 2, and the nozzle travel speed is set to 100 mm/s for both printers. The list of structural parts, amount of material used and time spent for printing are summarized in Table 3. It takes approximately 225 h of printing time to print all the parts. Total material used for the printing is around 3.2 kg, which cost less than \$100 USD. All hardware details are currently released as open-source in [23]. All the parts of the robots are held together by standard hex bolts and nuts, and it takes less than two man hours to fully assemble the robot from individual parts.



(a) Two 3D printers used for printing parts

(b) Printed parts

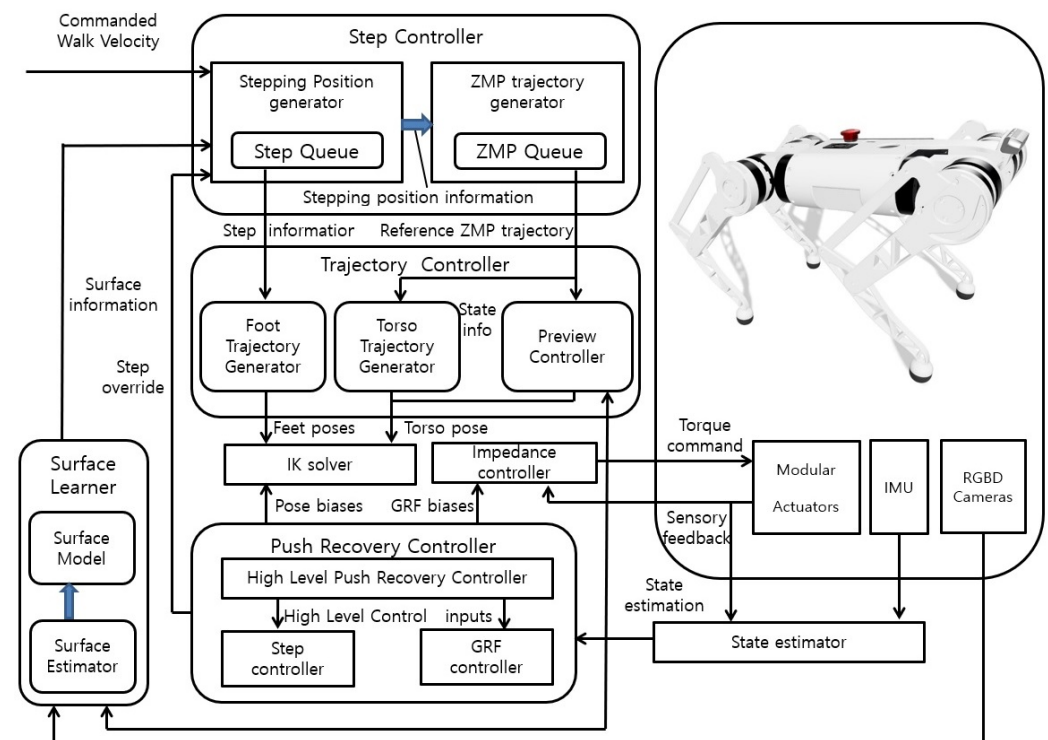
Figure 10. 3D printing of structural parts.

Table 3. Summary of 3D printing results.

Part Name	Qty.	Material Used (g)	Printing Time (hr)	Total Material Used (g)	Total Printing Time (hr)
Hip Interface Link	4	251	15.9	1004	63.8
Knee Crank Link	4	54	3.1	216	12.4
Knee Coupler Link	4	19	1.2	76	4.8
Leg Upper Link	4	199	13.4	796	53.6
Leg Lower Link	4	82	5.9	328	23.6
Foot Mount	4	4	0.3	16	1.2
Torso Side Plate	2	233	19.0	466	38.0
Torso Bottom Plate	1	112	7.1	112	7.1
Torso Top Plate	1	81	5.1	81	5.1
Torso Leg Mount	2	51	4.8	102	9.6
Realsense Mount	1	51	5.5	51	5.5
Total				3248	224.5

3. Motion Control

We implemented a simple motion control structure to test the basic capability of the system, which is based on our previous software framework that have been used for a number position controlled bipedal robots [24]. We plan to adopt advanced quadruped motion control methods in the future, such as real-time model predictive control and active ground reaction force control [25]. The overall architecture of the motion controller is shown in Figure 11.

**Figure 11.** Motion controller architecture of PADWQ.

3.1. Step Controller

At the beginning of each step, the swing leg, the landing positions for swing feet, and corresponding final torso pose need to be determined. We use a simple foot step generation process that can be generalized over a different gait patterns. First the intermediate torso pose is calculated from the current torso pose and the command velocity acquired from the human operator, and the swing feet target positions are calculated based on the

intermediate torso pose. In addition to the command velocity input, the filtered IMU data is also used for step push recovery [26], where a roll angle exceeding the threshold triggers sidestepping behavior of the robot to prevent tripping. Once all four target feet positions are determined, the actual target torso pose is set as the center of gravity of the quadrilateral formed by the four feet positions. Figure 12 shows the process being used to generate crawl, trot and bound gait patterns.

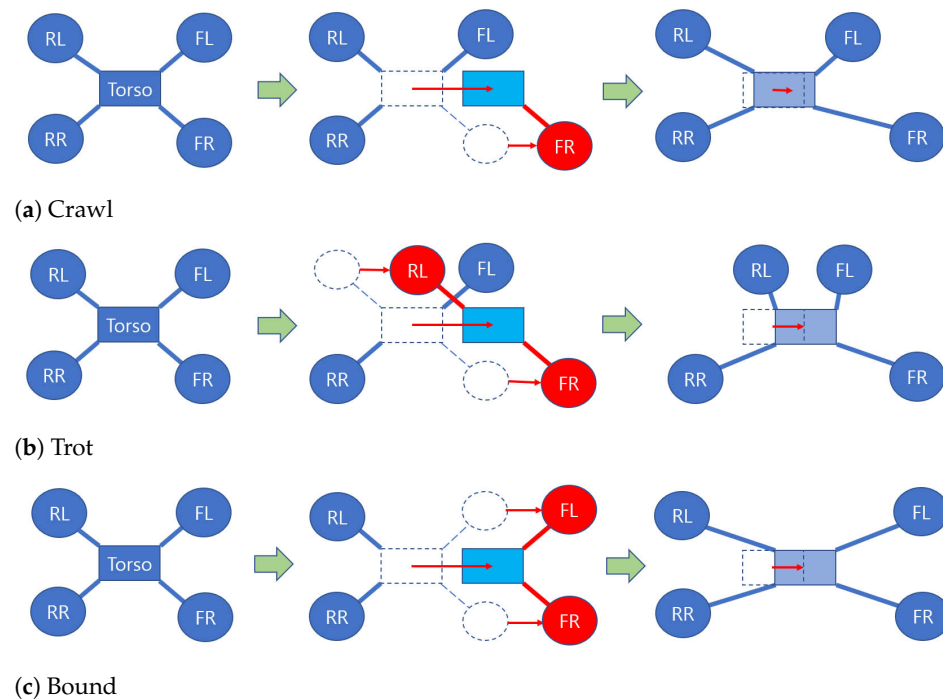


Figure 12. Foot and torso target position generation for various gait patterns.

3.2. Trajectory Controller

Once the initial and final feet and torso positions are determined at the beginning of the step, the feet and torso trajectories are calculated for each time step, and target joint angles for all the actuators are calculated using the inverse kinematics of each leg chain. To generate arbitrary feet trajectories, we use two cubic spline functions that defines the vertical and lateral movements of the foot over time. By changing the breakpoints of the spline functions, one can easily tune the temporal and spatial movement of swing feet. Figure 13 shows the current feet trajectory we use for the locomotion. The torso trajectory is generated using the zero moment point (ZMP) based controller commonly used for bipedal robots, where a cart-table model is used to analytically calculate the torso trajectory that keeps the ZMP of the robot in safe area to keep the robot balanced.

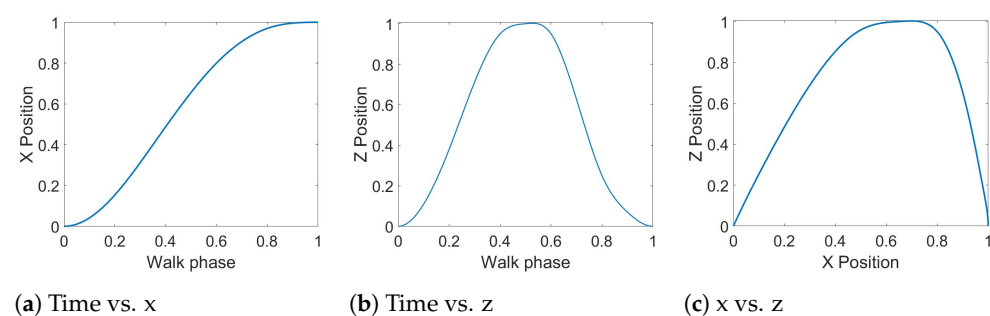


Figure 13. The reference foot trajectory.

3.3. Low Level Motion Control

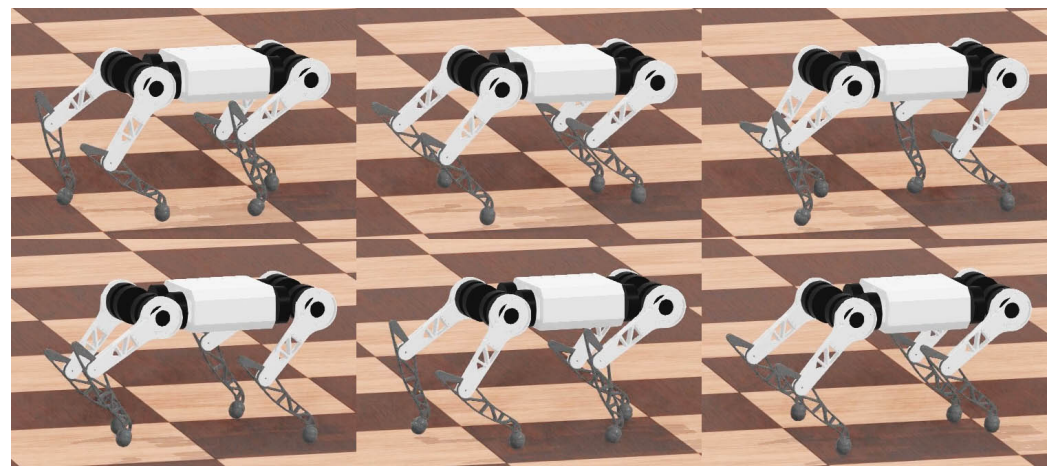
At the lowest level, we run four separate leg communication processes, each handles a leg chain through a CANable USB-to-CAN adapter, which sends torque control values and receives actuator position, velocity and current values at the rate of 1 kHz. We implemented two low level motion controllers that can use the reference leg trajectory input from high level controller. The first one is a joint level PD controller to control the target torque of each joint at this stage, where the target joint angle and gains are updated by the high level locomotion controllers using a shared memory interface. During locomotion, we use P gain of $88 \text{ Nm} \cdot \text{rad}^{-1}$ for support legs, which goes down to $44 \text{ Nm} \cdot \text{rad}^{-1}$ for swing legs to reduce the landing shock. We use D gain of $-0.55 \text{ Nm} \cdot \text{s} \cdot \text{rad}^{-1}$ for hip roll and pitch actuators, and $-0.275 \text{ Nm} \cdot \text{s} \cdot \text{rad}^{-1}$ for knee pitch actuators. The second one is a Cartesian impedance controller, which first calculates the target force output based on the feedforward force input and joint encoder values and then calculate joint torque target values for whole leg chain. At the time of writing the paper, we do not use feedforward torque or force inputs for the controllers to improve the tracking performance of the controllers, or explicit touchdown detection using the proprioception, which are left as future work.

4. Simulation Results

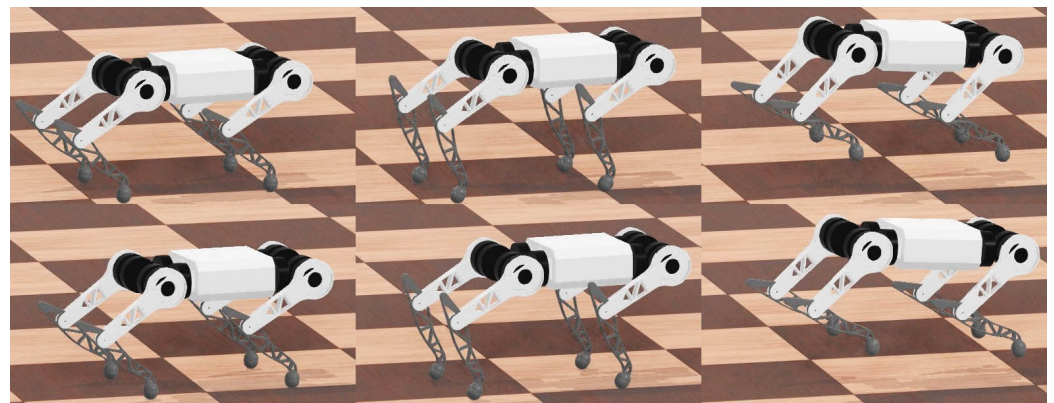
We used the Webots open source robotic simulator [27] to validate the motion controller of the PADWQ. The simulated model of the PADWQ is based on a slightly simplified version of the actual robot, where a single mass is used instead of the four bar linkage mechanism of the upper leg links. The same joint P and D gains, maximum joint velocity and torque values as the actual actuators are used for joint control parameters. As shown in Figure 14, we found that the suggested motion controller is capable of stable dynamic trotting, pronking and bounding with omnidirectional target velocity in the simulated environment. The walk parameters we used for the motion generation are summarized in Table 4.

Table 4. Initial Walk Parameters.

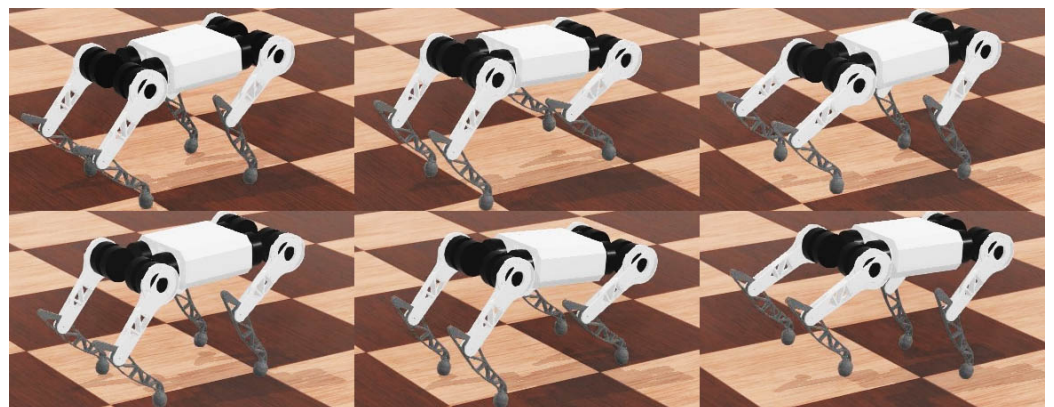
Torso Height	0.35 m
Torso X offset	0.05 m
Stance F-R	0.42 m
Stance L-R	0.30 m
Step Period	200 ms
Step Height	0.05 m
Lift phase	0.1
Land phase	0.9



(a) Trotting



(b) Pronking



(c) Bounding

Figure 14. Various dynamic gaits of PADWQ in a simulated environment.

5. Experimental Results

We performed the preliminary walk test of the PADWQ using a joint-level PD low level controller and the identical set of walk parameters used for the simulation, which is shown in Table 4. We found that the PADWQ is capable of stable dynamic trot-walking, even with a simple joint space PD control and without any parameter tuning. Front walk velocity of 1.0 m/s was measured during initial walk test, as shown in Figure 15.

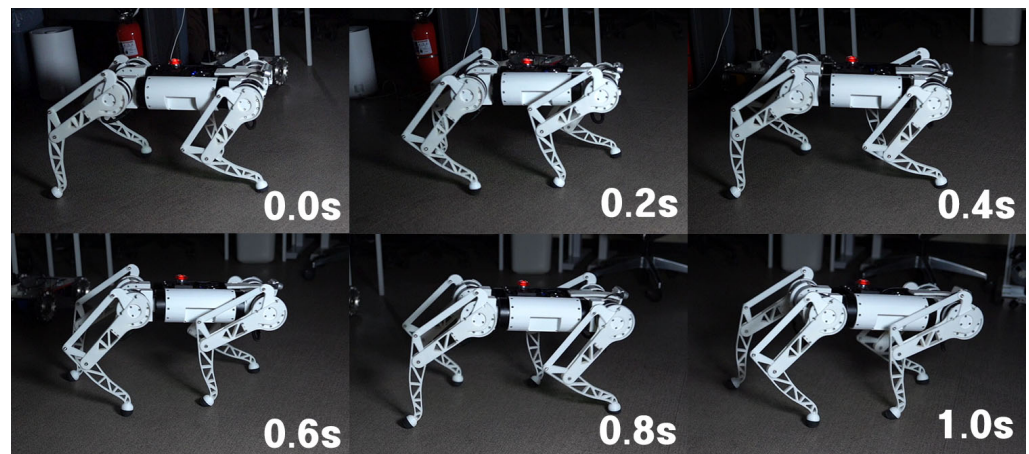


Figure 15. Dynamic trot-walking of PADWQ at 1.0 m/s.

We also let the robot walk in place continuously for long period to measure the power consumption, check possible thermal issues and validate the system integrity of the robot. The robot successfully completed the 50 min of continuous locomotion without any issues, as shown in Figure 16. We found that the typical power consumption during standing still is approximately 45 W, the peak power consumption measured during the test is 188 W, and the average power consumption during the locomotion is 81.6 W. We also measured the internal and external temperature of actuators during the experiment, which is shown in Figure 17. Internal temperature gradually rises during the locomotion, and reaches approximately 60 degree Celsius for knee actuators after the experiment. External temperature measured by a FLIR camera shows quite lower temperature of 46.5 degree Celsius, or 21 degree over the ambient temperature, which is well below the heat deflection temperature of the PLA material. If we extrapolate these power and thermal test results, we can estimate that the PADWQ has approximately one hour of continuous operation time.

Finally, to validate the rough terrain traversability of the robot, we tested the robot to walk over various outdoor terrains that includes asphalt, paved street, packed dirt and grass, which are shown in in Figure 18. During approximately 30 min of outdoor walking test, PADWQ has walked for a distance of 1 km over various surfaces reliably. We expect that adoption of more advanced control algorithms and additional parameter tuning will further improve the dynamic performance of the robot.

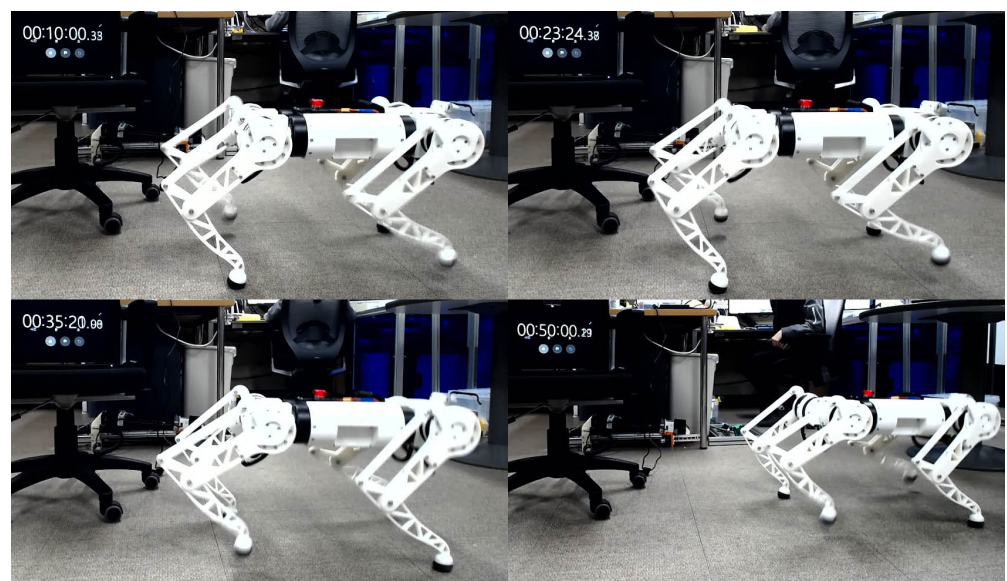


Figure 16. Continuous walking experiment.

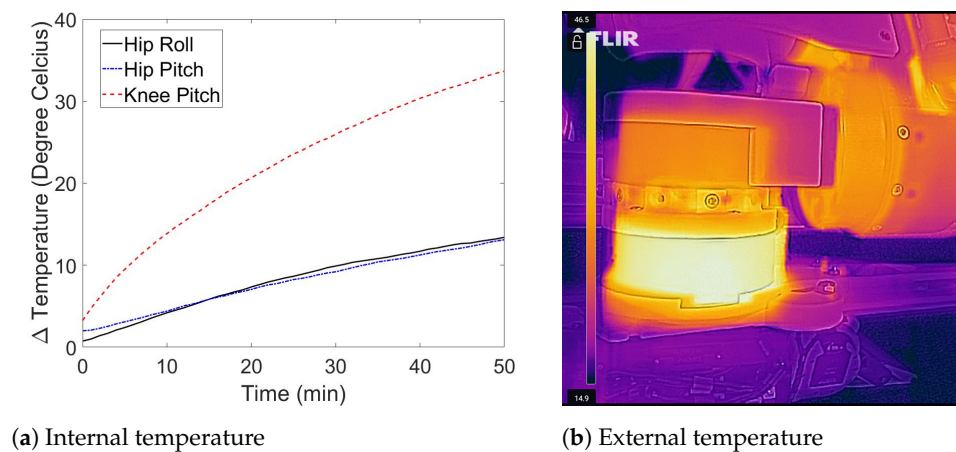


Figure 17. Actuator temperature during the 50 min of continuous walking.

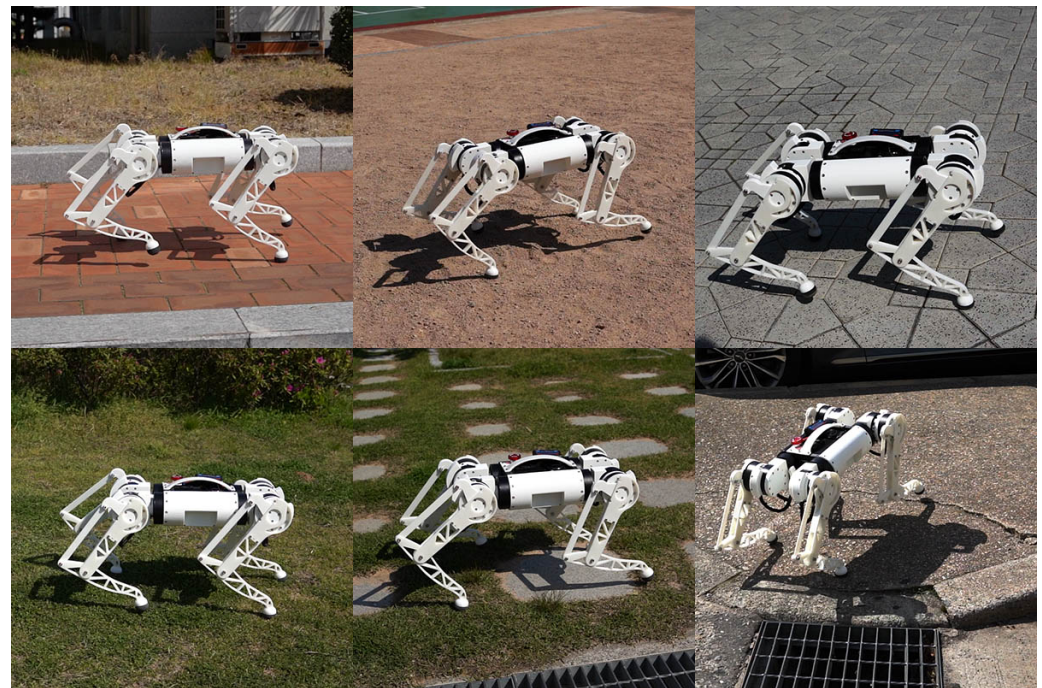


Figure 18. PADWQ walking over various outdoor terrains.

Discussions

We found that a combination of commercial off-the-shelf components and standard low cost 3D printed structural parts can make a capable, highly dynamic and affordable quadruped. In addition, preliminary tests have shown that a simple joint level PD controllers and trajectory-based motion controller is capable of letting the robot stably walk over several outdoor terrains.

We are currently working on implementing advanced control approaches to fully use the potential of the hardware. For the low level control, we plan to first use the Cartesian space impedance control with target feet ground reaction forces to better handle uneven and inclined surfaces. In addition, we plan to incorporate the estimated reaction force, body angular velocity and acceleration values to better stabilize the torso during locomotion.

In addition, one aspect we have not fully investigated in this work is the flexibility of the structural parts. We found the 3D printed leg parts to be noticeably less rigid than metal counterparts. Although this does not affect high frequency gait we have been using, it may become an issue when the robot is taking slower and more precise steps, or has

longer limb lengths. We plan to properly analyze the effect of limb compliance and possible compensation method as a future work.

6. Conclusions

In this paper, we present an open-source 12.7 kg quadruped with torque controlled quasi-direct drive joints. It is entirely made of commercial off-the-shelf components and standard 3D printed structural parts, which greatly lowers the entry barrier to dynamic quadruped robots. Initial tests using a simple motion controller shows that the quadruped is capable of highly dynamic locomotion, and is reliable and robust for a long continuous operation. Table 5 shows the cost breakdown of the PADWQ based on the open market prices at the time of writing this paper. The total cost for building a robot is approximately \$7700 USD, which comes down to around \$7100 when we exclude the cost of RGBD sensor and perception computer. Electronic parts and 3D printing material account for only 20.7 percent of the total cost, or roughly \$1600 USD, and actuator cost makes up the remaining 79.3 percent. Table 6 shows the comparison of current open-source torque controlled quadrupeds. PADWQ cost more to build than other open source robots, but has arguably unrivalled versatility due to its much higher payload, splashproof actuators, onboard sensor and GPU-equipped perception computer, and long battery life. Future work will include adopting more advanced control algorithms to release the full potential of the hardware, as well as adding application specific hardware to the robot such as sensor suites or lightweight manipulators.

Table 5. Cost Breakdown of PADWQ.

Component	Unit Cost (USD)	Qty	Total Cost (USD)	Percentage
RMD-X8 Actuator	359	12	4308	56.0
RMD-X8 Pro Actuator	448	4	1792	23.3
LattePanda Alpha	399	1	425	5.5
Jetson Xavier NX	399	1	399	5.2
WIT WTC901C IMU	34	1	34	0.4
XBOX 360 receiver	24	1	24	0.3
CANable USB-CAN	39	4	156	2.0
RealSense D435	170	1	170	2.2
6 s 4200 mAh Battery	120	1	120	1.6
3 s 5200 mAh Battery	68	1	68	0.9
3D Printing Material	30	3.2	96	1.2
Others	100	1	100	1.3
Total			7692	
Total w/o Perception			7123	

Table 6. Comparison of Current Open-Source Torque Controlled Quadrupeds.

Robot	DOF	Body Length (m)	Leg Length (m)	Mass (kg)	Cost (USD)	Fabrication Method
PADWQ	12	0.42	0.44	12.7	7692	3D Printed Plastic
Solo	8	0.42	0.32	2.2	4720	3D Printed Plastic
Stanford Doggo	8	0.42	0.25	4.8	3000	Lasercut Carbon/Aluminium

Author Contributions: Methodology, S.-J.Y.; Software, T.K. and S.-J.Y.; Hardware, J.K., T.K. and D.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the MOTIE under the Industrial Foundation Technology Development Program supervised by the KEIT (No. 20005032) and National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No.2019R1F1A1063846).

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

PADWQ	PNU Adaptive Dynamic Walking Quadruped
GPU	Graphics Processing Unit
DOF	Degree of Freedom
DIP	Dual Inline Package
5 FEA	Finite Element Analysis
PD	Proportional Derivative
SEA	Series Elastic Actuator
QDD	Quasi-Direct Drive
CNC	Computer Numerical Control
UART	Universal Asynchronous Receiver-Transmitter
CAN	Controller Area Network
LiPo	Lithium Polymer
ZMP	Zero Moment Point
NVMe	Non-Volatile Memory express
IMU	Inertial Measurement Unit
SBC	Single Board Computer
PLA	Polyactic Acid

References

1. Raibert, M.; Blankespoor, K.; Nelson, G.; Playter, R. BigDog, the Rough-Terrain Quadruped Robot. *IFAC Proc. Vol.* **2008**, *41*, 10822–10825. [\[CrossRef\]](#)
2. Khan, H.; Kitano, S.; Frigerio, M.; Camurri, M.; Barasuol, V.; Featherstone, R.; Caldwell, D.G.; Semini, C. Development of the lightweight hydraulic quadruped robot—MiniHyQ. In Proceedings of the 2015 IEEE International Conference on Technologies for Practical Robot Applications (TePRA), Woburn, MA, USA, 11–12 May 2015; pp. 1–6.
3. Lee, Y.H.; Lee, Y.H.; Lee, H.; Kang, H.; Lee, J.H.; Phan, L.T.; Jin, S.; Kim, Y.B.; Seok, D.; Lee, S.Y.; et al. Development of A Quadruped Robot System with Torque-Controllable Modular Actuator Unit. *IEEE Trans. Ind. Electr.* **2020**. [\[CrossRef\]](#)
4. Hutter, M.; Gehring, C.; Jud, D.; Lauber, A.; Bellicoso, C.D.; Tsounis, V.; Hwangbo, J.; Bodie, K.; Fankhauser, P.; Bloesch, M.; et al. ANYmal—A highly mobile and dynamic quadrupedal robot. In Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, Korea, 9–14 October 2016; pp. 38–44.
5. Seok, S.; Wang, A.; Otten, D.; Kim, S. Actuator design for high force proprioceptive control in fast legged locomotion. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura-Algarve, Portugal, 7–12 October 2012; pp. 1970–1975.
6. Bosworth, W.; Kim, S.; Hogan, N. The MIT super mini cheetah: A small, low-cost quadrupedal robot for dynamic locomotion. In Proceedings of the 2015 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), West Lafayette, IN, USA, 18–20 October 2015; pp. 1–8.
7. Park, H.W.; Wensing, P.M.; Kim, S. High-speed bounding with the MIT Cheetah 2: Control design and experiments. *Int. J. Robot. Res.* **2017**, *36*, 167–192. [\[CrossRef\]](#)
8. Hooks, J.; Ahn, M.S.; Yu, J.; Zhang, X.; Zhu, T.; Chae, H.; Hong, D. ALPHRED: A Multi-Modal Operations Quadruped Robot for Package Delivery Applications. *IEEE Robot. Autom. Lett.* **2020**, *5*, 5409–5416. [\[CrossRef\]](#)
9. Katz, B.; Carlo, J.D.; Kim, S. Mini Cheetah: A Platform for Pushing the Limits of Dynamic Quadruped Control. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 6295–6301.
10. Chen, Y.; Nguyen, Q. Adaptive Force-based Control for Legged Robots. *arXiv* **2021**, arXiv:cs.RO/2011.06236
11. Li, A.; Wang, Z.; Wu, J.; Zhu, Q. Bound Controller for a Quadruped Robot using Pre-Fitting Deep Reinforcement Learning. *arXiv* **2020**, arXiv:2011.00446.
12. García-Cárdenas, F.; Soberón, N.; Ramos, O.E.; Canahuire, R. Charlotte: Low-cost Open-source Semi-Autonomous Quadruped Robot. In Proceedings of the 2020 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC), Ponta Delgada, Portugal, 15–17 April 2020; pp. 281–286.
13. Boney, R.; Sainio, J.; Kaivola, M.; Solin, A.; Kannala, J. RealAnt: An Open-Source Low-Cost Quadruped for Research in Real-World Reinforcement Learning. *arXiv* **2020**, arXiv:2011.03085.
14. Sun, T.; Xiong, X.; Dai, Z.; Manoonpong, P. Small-Sized Reconfigurable Quadruped Robot With Multiple Sensory Feedback for Studying Adaptive and Versatile Behaviors. *Front. Neurobot.* **2020**, *14*, 14. [\[CrossRef\]](#) [\[PubMed\]](#)
15. Kau, N.; Schultz, A.; Ferrante, N.; Slade, P. Stanford Doggo: An Open-Source, Quasi-Direct-Drive Quadruped. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 6309–6315.
16. Grimminger, F.; Meduri, A.; Khadiv, M.; Viereck, J.; Wüthrich, M.; Naveau, M.; Berenz, V.; Heim, S.; Widmaier, F.; Flayols, T.; et al. An Open Torque-Controlled Modular Robot Architecture for Legged Locomotion Research. *IEEE Robot. Automat. Lett.* **2020**, *5*, 3650–3657. [\[CrossRef\]](#)

17. Spröwitz, A.; Tuleu, A.; Vespignani, M.; Ajallooeian, M.; Badri, E.; Ijspeert, A.J. Towards dynamic trot gait locomotion: Design, control, and experiments with Cheetah-cub, a compliant quadruped robot. *Int. J. Robot. Res.* **2013**, *32*, 932–950. [CrossRef]
18. Ahn, D.; Lee, J.S.; Cho, B.K. Development of a Quadruped Jumpable Robot PongBot-Q. *J. Korea Robot. Soc.* **2020**, *17*, 18–23.
19. Ramos, J.; Katz, B.; Chuah, M.Y.M.; Kim, S. Facilitating Model-Based Control Through Software-Hardware Co-Design. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 566–572.
20. Bledt, G.; Powell, M.J.; Katz, B.; Di Carlo, J.; Wensing, P.M.; Kim, S. MIT Cheetah 3: Design and Control of a Robust, Dynamic Quadruped Robot. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 2245–2252.
21. Unitree A1 Robot. Available online: <https://www.unitree.com/products/a1/> (accessed on 27 March 2021).
22. CANable: An Open-Source USB to CAN Adapter. Available online: <https://canable.io/> (accessed on 27 March 2021).
23. Kim, J.; Kang, T.; Song, D.; Yi, S.J. PADWQ Open-Source Project Repository. Available online: https://github.com/SJ-YI/PADWQ_open_source (accessed on 27 March 2021).
24. McGill, S.G.; Yi, S.J.; Yi, H.; Ahn, M.S.; Cho, S.; Liu, K.; Sun, D.; Lee, B.; Jeong, H.; Huh, J.; et al. Team THOR's Entry in the DARPA Robotics Challenge Finals 2015. *J. Field Robot.* **2017**, *34*, 775–801. [CrossRef]
25. Ding, Y.; Pandala, A.; Park, H. Real-time Model Predictive Control for Versatile Dynamic Motions in Quadrupedal Robots. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 8484–8490.
26. Yi, S.; Zhang, B.; Hong, D.; Lee, D.D. Online learning of low dimensional strategies for high-level push recovery in bipedal humanoid robots. In Proceedings of the 2013 IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, 6–10 May 2013; pp. 1649–1655. [CrossRef]
27. Michel, O. Webots: Professional Mobile Robot Simulation. *J. Adv. Robot. Syst.* **2004**, *1*, 39–42.