Department of Electrical and Computer Engineering
North South University

Senior Design Project

ACROBOT

Design and Fabrication of Small Robots With Inverted Pendulums

Shahriar Ratul 2111514642

Faculty Advisor:

Dr. Shahnewaz SiddiqueProfessor

ECE Department

Spring, 2025

APPROVAL

Shahriar Ratul (ID # 2111514642) from the Electrical and Computer Engineering Department of North South University have worked on the Directed Research Project titled *"ACROBOT: Design and Fabrication of Small Robots with Inverted Pendulums"* under the supervision of Dr. Shahnewaz Siddique in partial fulfillment of the requirement for the degree of Bachelor of Science in Engineering and has been accepted as satisfactory.

Supervisor's Signature

………………………………………

Dr. Shahnewaz Siddique

Professor

Department of Electrical and Computer Engineering

North South University

Dhaka, Bangladesh.

Chairman's Signature

………………………………………

Dr. Mohammad Abdul Matin

Professor

Department of Electrical and Computer Engineering

North South University

Dhaka, Bangladesh.

DECLARATION

This is to declare that this project is our original work. No part of this work has been submitted elsewhere, partially or fully, for the award of any other degree or diploma. All project-related information will remain confidential and shall not be disclosed without the formal consent of the project supervisor. Relevant previous works presented in this report have been properly acknowledged and cited. The plagiarism policy, as stated by the supervisor, has been maintained.

Students' names & Signatures

1. Shahriar Ratul

_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

ACKNOWLEDGEMENTS

The authors would like to express their sincere gratitude to their project and research supervisor, **Dr. Shahnewaz Siddique**, Professor, Department of Electrical and Computer Engineering, North South University, Dhaka, Bangladesh, for his unwavering support, insightful guidance, and valuable advice throughout the development of this project. His continuous encouragement and mentorship were instrumental in the successful completion of both the research and this report.

We would also like to thank the **Department of Electrical and Computer Engineering, North South University**, for providing the necessary resources and facilities that enabled us to carry out our project effectively.

Furthermore, we extend our appreciation to our families and loved ones for their unconditional support, patience, and encouragement throughout this journey.

ABSTRACT

ACROBOT

Design and Fabrication of Small Robots With Inverted Pendulums

In this work, we present the design and fabrication of a high-mobility hexapod robot integrated with the dynamic balancing principle of inverted pendulums. A hybrid robotic system with the goal of stable locomotion and autonomous self-balancing by combining the versatility of a hexapod with the dynamic stability features of the balancing bot Cubli. In this project, hardware components such as servos, microcontrollers, and sensors were used. Inverse kinematics are designed and implemented to control the robot's movements. 3D printed components and embedded system programming were used in the fabrication of the system. The robot exhibits the capacity to stay balanced while changing positions and walking. This development offers advancement in robot mobility, particularly for the robots that require static and dynamic stability. The results from this project have implications in fields such as robotics, autonomous systems, and advanced locomotion technology. We have addressed the mathematical complexity involved in balancing and controlling such hybrid robots and have simplified it to a level that allows operation on low processing power.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1 Introduction

1.1 Background and Motivation

Robotics as a field is evolving very quickly, particularly in the area of systems capable of operating in different complex environments, executing intricate tasks. In this respect, hexapod robots have been developed and designed to walk on rough terrain because of their great stability and load distribution. Furthermore, control systems based on inverted pendulum balancing, as demonstrated in the Cubli, have enabled self-stabilizing robots to adjust dynamically to predefined postures and maintain a self-calibrating condition.

The aim of this project stems from the intention to combine two areas of study: multi-legged locomotion and dynamic balancing. Active balancing systems have not yet been extensively researched in conjunction with hexapod systems that use static stabilization and maneuverability. This project intends to modify a hexapod robot by placing a balancing system inspired by inverted pendulums, which will improve the modularity and maneuverability of these robots. This motivation arises particularly from the need to efficiently perform search-and-rescue operations, planetary exploration missions, and in cases where conventional wheeled or tracked robots fail to meet requirements.

1.2 Purpose and Goal of the Project

This project aims to develop a hexapod robot that ‘walks’ and ‘balances’ itself via the inverted pendulum principles. Walking on six legs complements the hybrid system’s stability while the Cubli-like mechanisms provide balance control during walking. By using sensor feedback, servo motors, and kinematic computations for tracking movement, the project introduces a new form of legged locomotion through dynamic shifting of position during walking.

1.3 Organization of the Report

That report is divided into several chapters in order to present the work systematically done in the project.

- Chapter 2 Reviews of the existing research on hexapod locomotion and active balance-control mechanisms with reaction wheels.
- Chapter 3 Details the methodology associated with system design, components, and implementation methods.
- Chapter 4 discusses the experimental setup, results, analysis, and observations.
- Chapter 5 socioeconomic and environmental aspects of the project are described.
- Chapter 6 presents the project timeline and budget.
- Chapter 7 describes how the project fits into broader engineering problem domains and activities.
- Chapter 8 concludes the report with a summary, identifies limitations, and provides future improvement directions.
- Finally, references in the entire report are listed at the end of this paper in IEEE format.

Chapter 2 Research Literature Review

2.1 Existing Research and Limitations

Inspired by the locomotion of insects, hexapod robots have been widely studied for their stability and adaptability in traversing uneven terrains. Their six-legged design provides unparalleled balance and fault tolerance, making them suitable for various applications, including exploration, surveillance, and search-and-rescue missions. For instance, the RHex robot, developed by researchers at the University of Pennsylvania, demonstrated remarkable agility and robustness in navigating complex environments. Ours is primarily known as a 3R serial open-loop robot where each of its legs contains three rotational joints (coxa, femur, and tibia), which translate to three degrees of freedom (DOF) per leg. This robot can efficiently walk on diverse surfaces without tracks or wheels.

Advancements in fabrication techniques have further propelled hexapod research. Zhu et al. (2021) showed a method for adaptive gait planning for hexapod locomotion over unstructured terrains, while Guo et al. (2020) developed a terrain-adaptive biomimetic hexapod that utilizes real-time feedback. One of the widely used locomotion techniques in hexapod robots is tripod gait. In tripod gait, three legs move while the other three provide support, ensuring balance and energy efficiency. The gait is achieved by calculating joint angles based on desired foot positions in space. This approach is known as inverse kinematics (IK). A simplified implementation of tripod gait with simple geometry and IK was proposed in the paper "Hexapod Leg Coordination Using Simple Geometry: Tripod Gait and Inverse Kinematic Approach." Despite these advances, most designs continue to struggle with controlling their power consumption, energy efficiency, and complicated control algorithms. Their control systems often do not account for real-time balance adjustments, limiting their performance in unpredictable environments. This gap highlights the need for integrating dynamic balancing mechanisms into hexapod designs.

At the same time, Cubli is a project by Rüegg et al. (ETH Zurich, 2013), which is a robot in the form of a cube that can balance itself on its edges or corners utilizing 3-axis reaction wheels. This system demonstrated very fine control and dynamic balancing and gave valuable insight into how a robot could be stabilized using reaction wheels

Regardless, the application of active balancing control for legged robots still remains largely untouched. Integrating the static stability of hexapods with the dynamic balance control of the Cubli will likely result in robots that can more easily adapt to varying terrains and tasks. This approach attempts to offer a solution to the problem of adaptability and stability central to robotics.

Chapter 3 Methodology

3.1 System Design

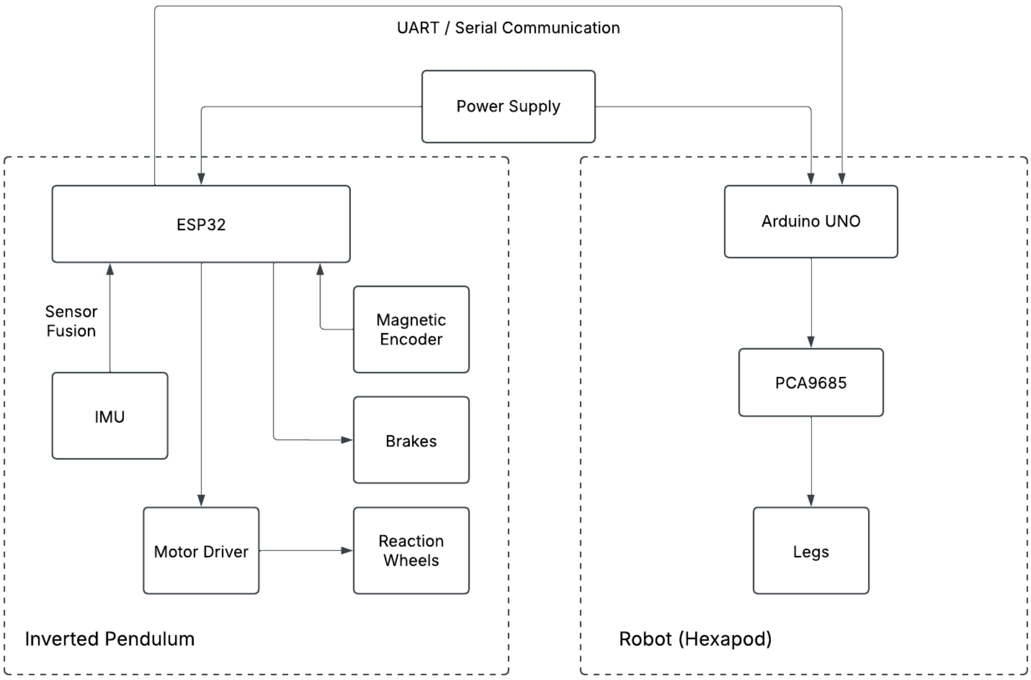


Figure 3.1.a : System Block Diagram

The block diagram above gives a high level description of how the hardware of the project was implemented. The Inverted Pendulum, like the Cubli, consists of a 6-axis IMU (MPU6050) which along with the Mahony sensor fusion algorithm was used to get the orientation of the robot. The 12-bit magnetic encoder (AS5600) was used to get the angle of the reaction wheel. This data was then processed to handle wrap-around (encoder jumps from 4095 to 0 and vice versa) and numerically differentiated to get the angular speed of the reaction wheels. Each motor had a reaction wheel connected to it and an encoder to calculate its angular speed. The brakes were used to induce high torque through the reaction wheel in order to “jump” from the rest position to its balancing position. All of these along with the controller implemented inside the ESP32 was controlling the inverted pendulum.

From there the ESP32 was also communicating through UART/Serial Communication to work together with the robot, which in our case was the hexapod. Using UART simplified the communication between the microcontrollers, and allowed for the robots to work in unison.

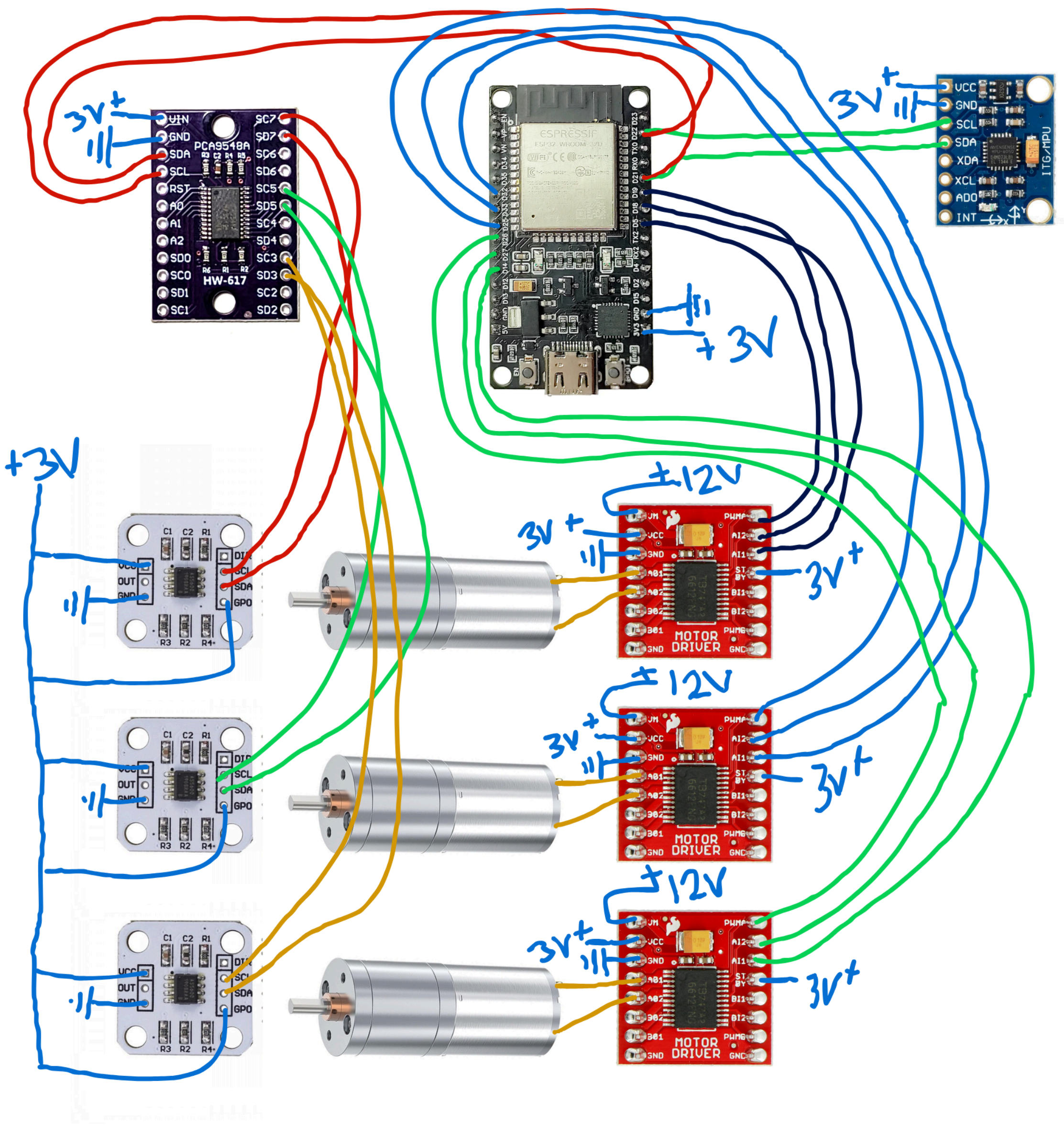


Figure 3.1.b : Cubli Circuit Diagram

The image above shows the connections in the inverted pendulum system. All the cables were managed effectively, in the hardware implementation, and connections were checked with code.

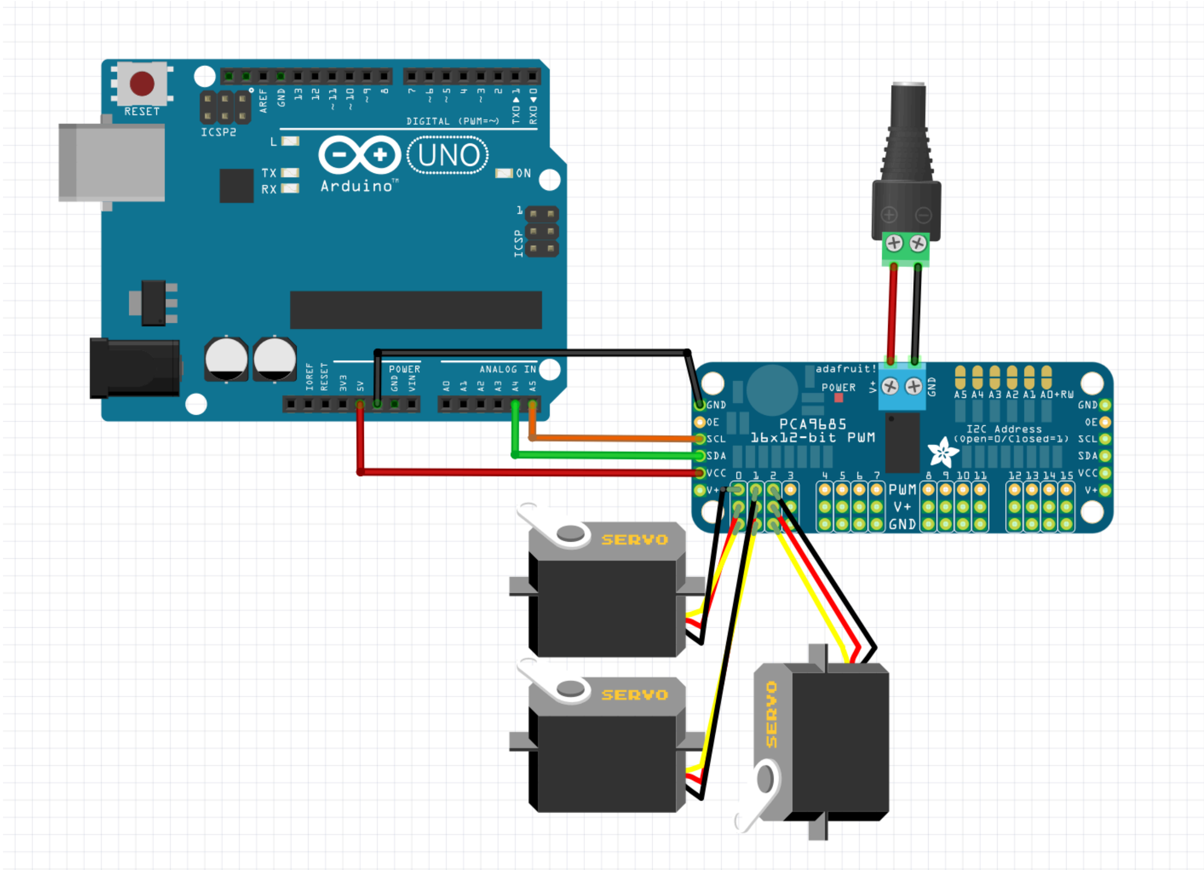


Figure 3.1 c : Hexapod Circuit Diagram

The image above shows the connections in the circuit diagram of a leg from the hexapod robot system. Where the Arduino , PWM servo driver and the servo motors of the leg are connected to each other for its functioning.

3.2 Hardware and/or Software Components

Our project was a combination of both hardware and software, and complexity lied in both areas. One of the first things we coded was the sensor fusion algorithm to calculate the orientation (euler angles, or quaternions, or rotation matrix) from the IMU data. We used the MPU6050 as our 6-axis IMU, since it is cheap and we do not need magnetometer data from other 9-axis IMU like MP9050. Finding the euler angles (roll, pitch, and yaw) was very simple. Using some basic trigonometry along with the fact that gravity is a constant acceleration always acting downwards is one way of calculating orientation. Another way is to numerically integrate the gyroscope data. However both methods have their cons, the first method is susceptible to noise and high frequency disturbances, and the second method has a tendency to drift since the sensor noise is also getting integrated. A simple way to fix this is with a complementary filter which is a weighted average of both methods (using accelerometer and gyroscope).

However, trying to represent orientation using the minimal number of variables/coordinates (equal to the degrees of freedom), the configuration space cannot be represented smoothly, resulting in singularities like gimbal lock. Gimbal lock results in a loss of degree of freedom after two axes become aligned. The way to fix this is to shift from the explicit representation of orientation by euler angles (utilizing 3 variables) to an implicit representation like quaternions (utilizing 4 variables) or rotation matrix (utilizing 9 variables). This was done by using nonlinear complementary filters like the Mahony and Madgwick filters. Madgwick filters are better but it is also more time intensive. Mahony is faster and has the advantage of working better without a magnetometer reading, which is used to calculate yaw or heading. But since yaw is not a necessary orientation data we need, we opted to use the mahony algorithm initially, but later switched to the madgwick algorithm. Using a kalman filter was also a viable option, but it required way more computation.

We used a Dual TB6612FNG as the motor driver, for its compact design and efficiency. Even though one module can control two motors, we opted for it to control one in order to allow greater current flow to the one motor. A buck converter connected to a laptop charger was used to provide a constant 12V to the motor driver and gave high current.

The AS5600 was used to get the orientation of the reaction wheel. The body of the Cubli was designed to attach the module in front of the reaction wheel and a radially magnetized magnet was superglued to the wheel. The AS5600 uses multiple hall effect sensors to give us 12-bit information about the orientation. This can later be used to get the change in orientation and numerically differentiated to get angular speed of the wheel. Some if statements were used to handle wrap-around (encoder jumps from 4095 to 0 and vice versa). But the challenging part in this was to handle the I2C communication. As each AS5600 had the same I2C address (0x36), there were bound to be collisions using the same I2C bus. The ESP32 did have 2 I2C buses but we had 3 encoders. There were software solutions, but that sometimes required updating libraries (as many used the Wire.h library by default). So we opted for a hardware solution using the 8-channel I2C multiplexer PCA9548A. This shortened the code and was an elegant addition.

For the motor we used a GA25-370 12V motor. Most Cubli implementations used a Nidec motor, which had a builtin encoder and provided decent torque. But the unavailability of it in Bangladesh and made the cheap and available GA25-370 our motor of choice. The gearbox of this motor gave decent torque and using a brushed DC motor, though being less efficient that brushless one, is much more simpler. And the lack of incremental encoders in the GS25-370 was solved with the magnetic absolute encoders.

And at last we used the ESP32 as the brains of it all. The inclusion of a lot of PWM pins was vital in controlling all the modules. It was also more powerful and cheaper than the arduino. A raspberry Pi is also a viable alternative but it is very expensive and overkill for the inverted pendulum system. All the coding was done using the Arduino IDE after installing the relevant packages to run ESP32 codes. It is here where all the control algorithms are stored. The code can be viewed in Github. https://github.com/BunnyWarlock/Cubli-ESP32/tree/main

The PID controllers are often used when there is no deeper understanding of the physical system. Though this is not the case here, a PID controller does act as a good starting point for trying to control a robot. PID stands for Proportional-Integral-Derivative. It is a controller that strives to minimize the present error, past error and future error. The P stands for proportional to error at time, I is proportional to integral of present time and D is proportional to derivative of present time. Below is the discrete version of the PID algorithm. The proportional part is the main part of the controller, the integral part prevents steady state error and the derivative part smooths the whole control.

$$u[k] = K_p e[k] + K_i \sum_{i=0}^{k} e[i]\Delta t + K_d \frac{e[k] - e[k-1]}{\Delta t}$$

The PID algorithm would output how much current the motor should be provided with. This is fine since the rule of thumb for motors is that current is proportional to torque, and it is the torque that is balancing the inverted pendulum. To provide this current we need to first convert it to voltage. We can do this using the electrical motor equation.

$$V(t) = L\frac{di(t)}{dt} + Ri(t) + K_e\omega(t)$$

For motors the time constant of the RL circuit is very small compared to the mechanical time constants of the rest of the system, and so we can safely set L to zero. The motor constant can be calculated empirically or using the values in the datasheet. It is to be noted that we do not need to complicate our motor model by modelling the gearbox or transmission separately. We can abstract this away by simply calculating an equivalent motor constant for the motor and gearbox combined. The resistance can be calculated using a multimeter. So, after getting the voltage we can easily convert this to a PWM signal.

$$PWM = \frac{Voltage}{Voltage_{Supply}} * PWM_{MAX}$$

The setpoint can be measured by reading the roll and pitch data near the balancing point. Thus the PID controller is complete. There is still scope for improvements, like including a low-pass filter on the derivative, clamping the integral to prevent integral windup, etc. We can also add a variate target angle, this helps prevent continuous rotation and also allows the controller to account for uneven terrain, and increases robustness.

Implementing the LQR control is where it gets interesting. LQR stands for Linear-Quadratic-Regulator. Implementing it requires us to create the state space representation of the system and then linearising it (either using the Jacobian or making qualitative assumptions like using the small angle approximations for sin) around a fixed point, where the system is in “equilibrium” (like the balancing point). Doing this sounds like a hassle but it is worth it since LQR gives us an optimal controller for the given cost matrix Q (penalizes deviations in the state variables) and R (discourage aggressive actuation). It does this by minimizing the cost function.

$$J = \int_0^{\infty} \left(\mathbf{x}^\top(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{u}^\top(t)\mathbf{R}\mathbf{u}(t) \right) \, dt$$

And solving it using the equivalent algebraic Riccati equation. However, doing it by hand is unnecessary as MATLAB provides all the tools at hand.

First we must create a non-inertial frame {B} attached to the Cubli body.

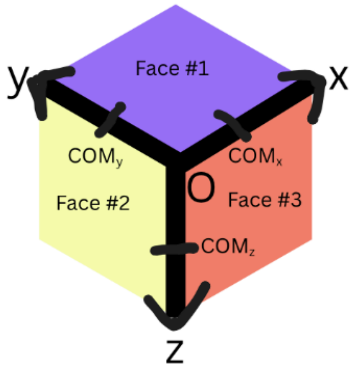
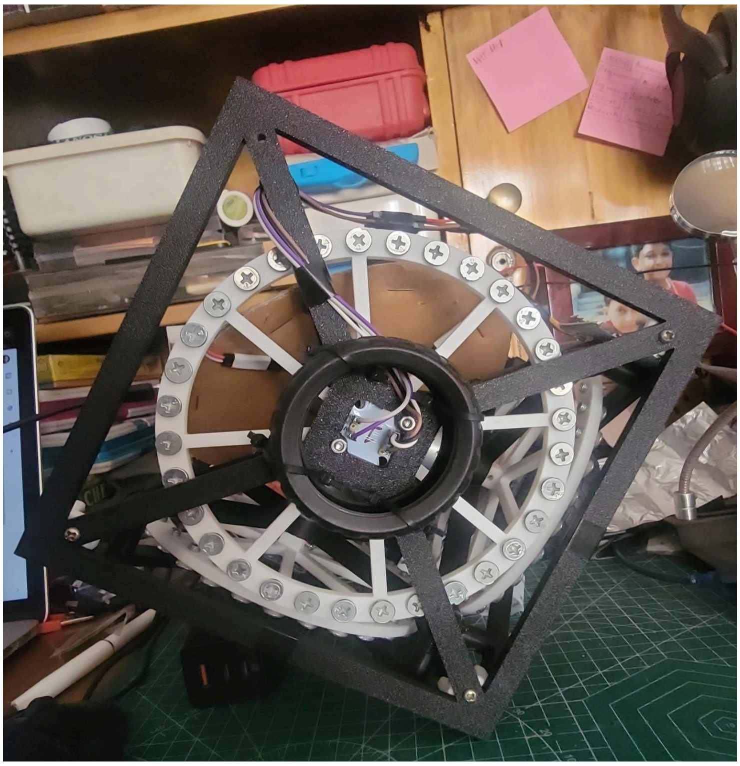
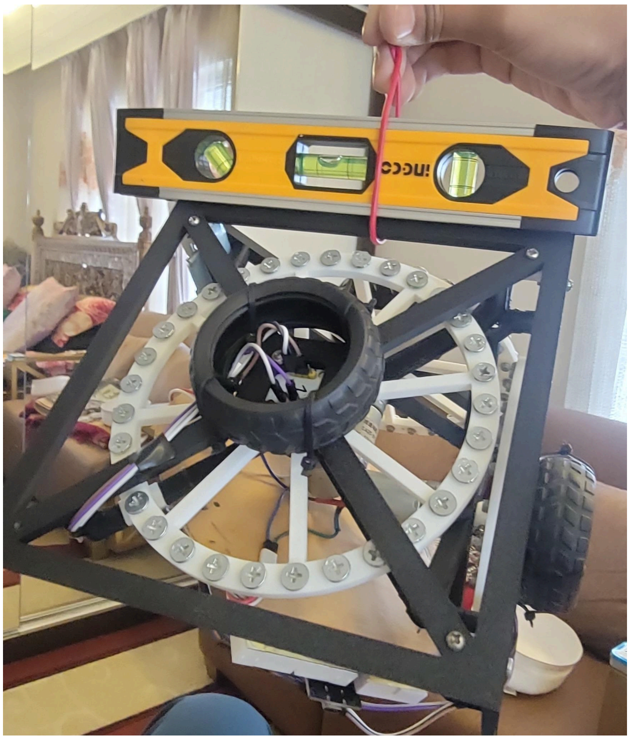


Figure 3.2 a : A non-inertial frame

Let's assume we will balance the Cubli on the edge represented by the z-axis. This would mean that face #1 would be facing us.



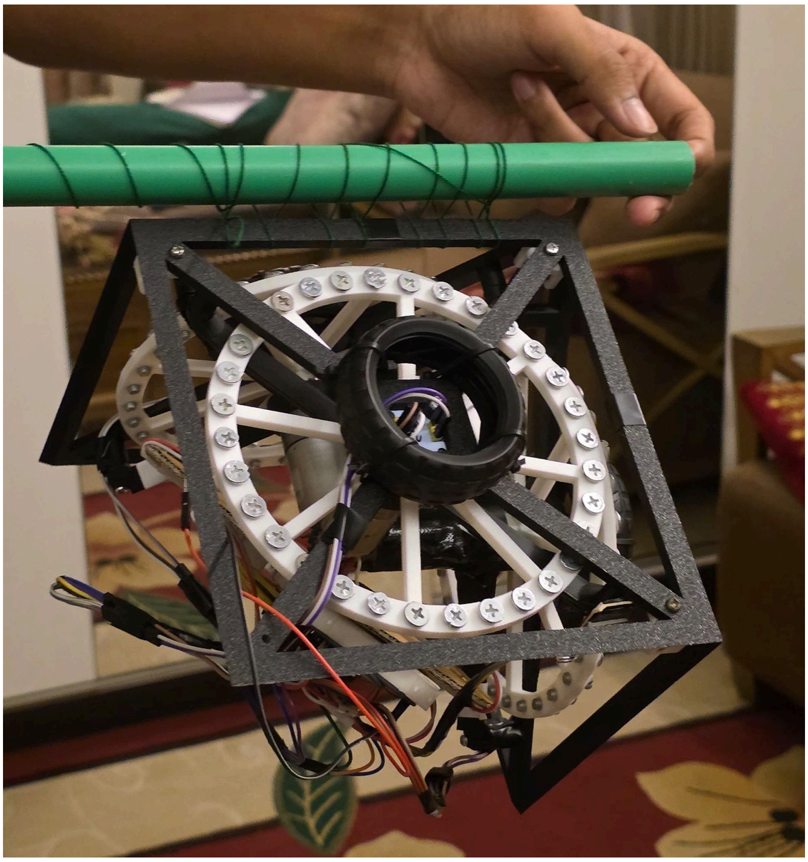
First we must locate the center of mass (COM) of the whole Cubli. To do that we can find the point, at each axis, which is the minimum distance away from the COM. To do this we can hang the cubli for each axis along with a level indicator. If the Cubli is hanging and the level is fine then we know that we have found the point. The picture below shows a demonstration.



Repeating this step for each axis we will find the coordinates of the COM in our non-inertial frame (by taking a measurement from the corner where all the axis intersect or the origin to that point using a ruler). Following this we must now find the inertia of Cubli with respect to the z-axis. To do this we can let the Cubli hang freely about the z-axis and then letting it oscillate freely like a pendulum and then calculate the time period. And putting it in this formula

$$I_{bz} = M_b g d_z \left(\frac{T}{2\pi}\right)^2$$

Where M_b is the mass of the body, d_z is the minimum distance from the z-axis to the COM, and I_bz is the inertia of the body with respect to the z-axis. To let the body hang freely about the z-axis, we can utilize some strings, a pipe and a rolling joint. This can be made easily by letting the string make a 8 pattern between the z-axis and the pipe. This results in a very low friction joint.



We can then measure the time it takes to do a few oscillations and then take linearly interpolating the time for one oscillation. We can also easily calculate d_z

$$d_z = \sqrt{(}COM_x^2 + COM_y^2)$$

Putting all the values in the formula we would get the total inertia. The inertia of the reaction wheel can be calculated easily by considering the 3D printed part as a ring and the M5 bolts as a point mass.

We will now take the state space representation derived in [5], by modelling the system using Newtonian mechanics, implementing the motor equations in the model equations and then linearizing it at the top balancing point.

$$A = \begin{bmatrix} 0 & 1 & 0 \\ \frac{k_{mgl}}{I_{bz}} & -\frac{b_{bz}}{I_{bz}} & \frac{K_t K_e}{R I_{bz}} + \frac{b_R}{I_{bz}} \\ -\frac{k_{mgl}}{I_{bz}} & \frac{b_{bz}}{I_{bz}} & -\frac{I_{bz}+I_R}{I_{bz} I_R}(b_R + \frac{K_t K_e}{R}) \end{bmatrix} B = \begin{bmatrix} 0 \\ -\frac{K_t}{R I_{bz}} \\ \frac{I_{bz}+I_R}{I_{bz} I_R}\frac{K_t}{R} \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} D = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

where

$$x = \begin{bmatrix} \theta \\ \dot{\theta} \\ \omega_R \end{bmatrix} u = \begin{bmatrix} v \end{bmatrix}$$

Some of the subscripts were changed here to match the context.

$$k_{mgl} = M_b d_z g$$

Here b_bz is the viscous friction of the Cubli body w.r.t the z-axis. But for edge balancing we can safely assume b_bz to be zero. Putting all the values we easily get the matrix A, B, C, D.

We can create the Q and R cost matrix and then decide the sampling rate of the discrete controller. Then we can plug it into Matlab to get the gain matrix K_d.

```
>> Ts = 0.001; % Sampling time in seconds

>> sys_d = c2d(ss(A, B, C, D), Ts)

>> Ad = sys_d.A;

>> Bd = sys_d.B;

>> K_d = dlqr(Ad, Bd, Q, R)
```

In code we can now easily get the control voltage V by applying the control law u = -K_d*x. Convert the voltage to PWM using the equation earlier and it is done. We can also find a limit for the recovery angle by dividing the supply voltage with the first element of K_d.

To test the system, we made a hexapod. The six legs have 3 degrees of freedom. And we used servos for its actuation. At this point, the goal was to balance the cube on top of a walking hexapod. The hexapod leg has three joints femur, tibia and coxa. The hexapod was made to walk using inverse kinematics, where the target foot positions are chosen and the coxa, femur, and tibia angles are calculated to reach them. By following tripod gait.

TABLE I. A SAMPLE SOFTWARE/HARDWARE TOOLS TABLE

Tool	Functions	Other similar Tools (if any)	Why selected this tool
MPU6050	6-axis IMU	MPU9050	Cheap and 9-axis IMU is overkill
Madgwick filter	Sensor fusion to get orientation data from IMU readings	Complementary filters, Mahony filter, Kalman filters	More robust and uses quaternions, without too much computations.
Dual TB6612FNG	Motor driver	L298N H-Bridge	Cheap, efficient, and compact
Buck converter	Power supply	DC Power Supply	Cheap, small and functional
AS5600	Magnetic encoder	AS5600L	Cheap, and the need for a modifiable I2C address was removed by the multiplexer
PCA9548A	8-channel I2C Multiplexer	Software solutions	Less hassle
GA25-370	Geared Brushed DC Motor	Nidec motors with encoder.	Cost and availability
ESP32	Microcontroller	Arduino, Raspberry Pi	Cheap and powerful enough
Servo MG90	Actuates hexapod legs (rotation + position)	SG90, MG996R	Cheap and small

Arduino Uno R3	**Main microcontroller for hexapod (controls servos & logic)**	**ESP32, Raspberry Pi Pico, STM32**	**Easy to program, Large community support**
Bambu Lab X1 Carbon 3D Printer	**Fabricates hexapod legs and mechanical parts**	**Prusa i3 MK3S, Creality Ender 3**	**High precision access from NIRO Lab**
16-Channel Motor Controller (PCA9685 or similar)	**Provides PWM control for up to 16 servos**	**Custom drivers, multiple Arduino PWM pins**	**Simplifies servo control, offloads timing from Arduino**

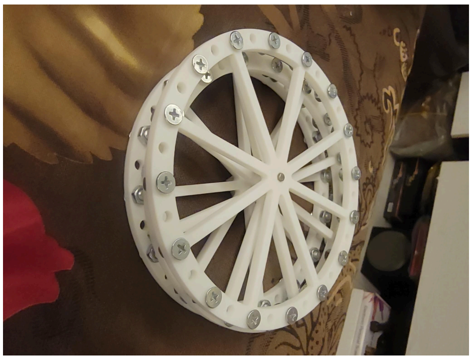
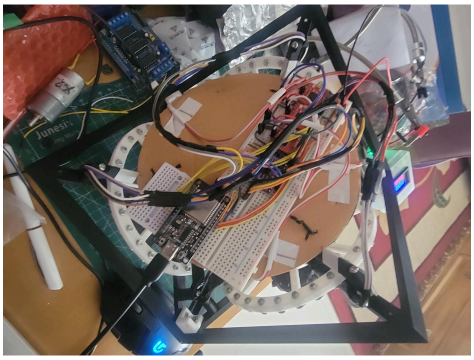
3.3 Hardware and/or Software Implementation

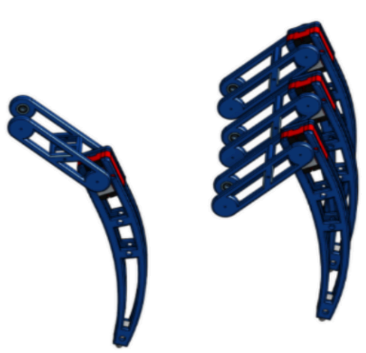
The body of the Cubli was 3D printed in Bambu Lab Carbon X1 . One prototype was made first and then the final Cubli was made taking into account all the things learned from the first prototype. Everything was screwed using M3 nuts and bolts. The reaction wheel had holes as far as possible from the center to hold M5 screws. This helped increase the inertia of the reaction wheel drastically. All the wiring was done using the breadboard wires and extra attention was given to cable color and cable management, which helped the final build look neat.

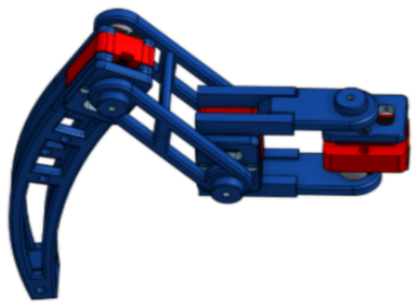
The entire circuit was implemented on a breadboard, with surface mount wires acting as the connectors. The IMU was placed separately, parallel to one of the faces for better orientation approximation.

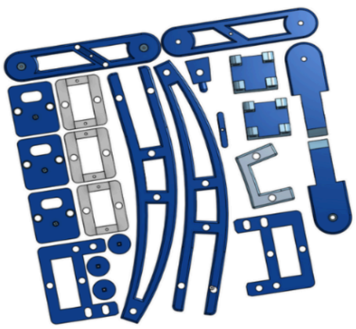
The encoders were bolted to the faces and a tire was placed on top of it, to protect the wires and cushion the falls. A platform was made using cardboard to hold the breadboard.
The body parts of the hexapod were also 3D printed. We experimented by prototyping. The first prototype of the leg was very modular and had metal screws to fit everything in. The metal screws made the legs heavier, so we made the legs screwless in the second prototype.
The second prototype leg was used to make the final hexapod robot.

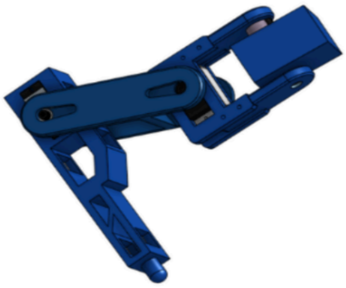


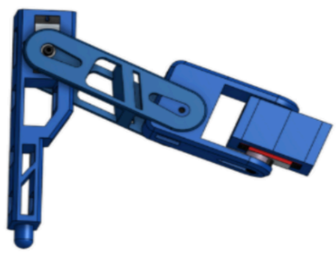




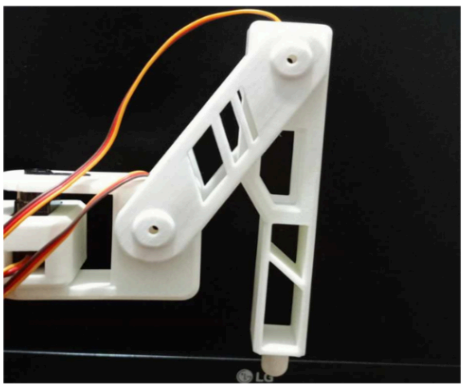



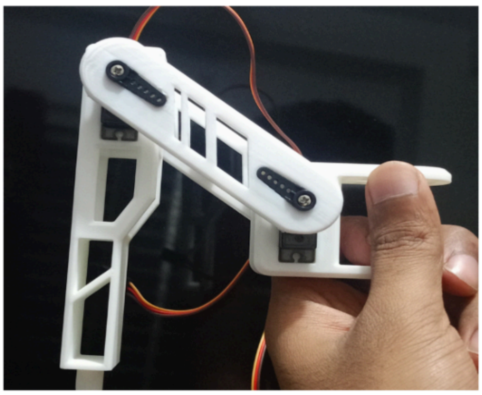


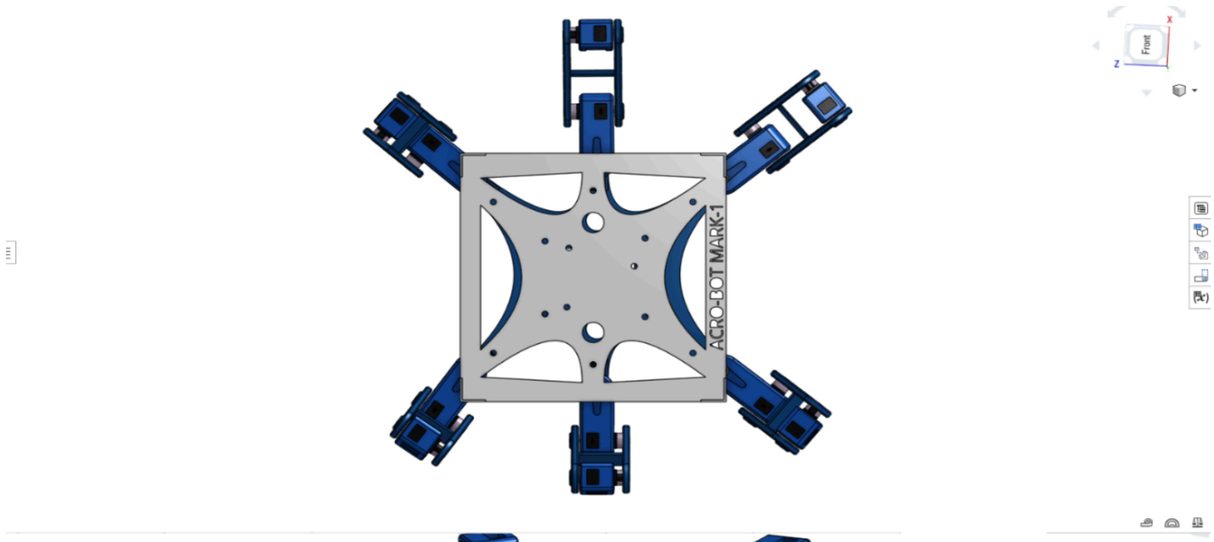


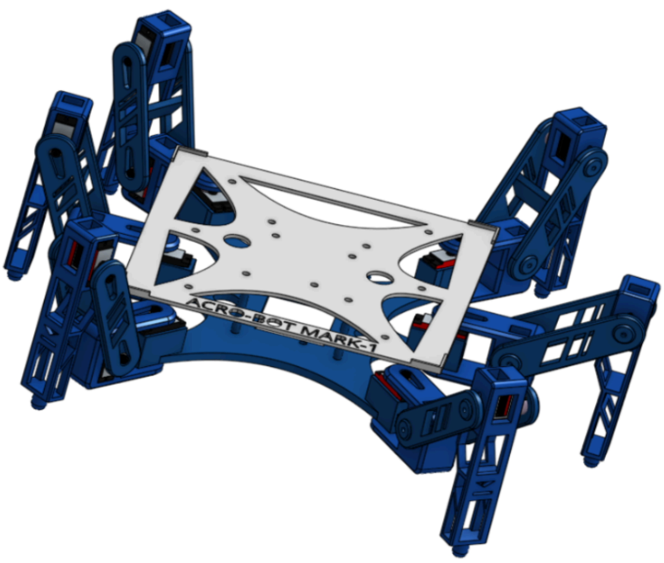


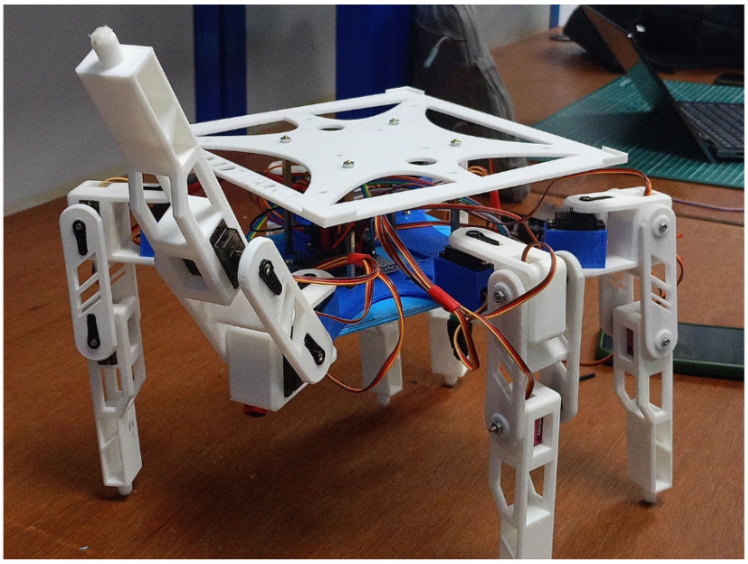


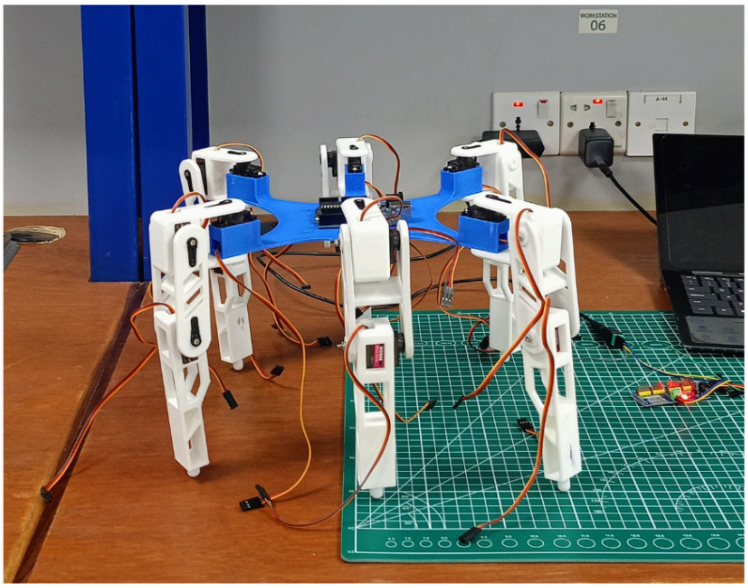


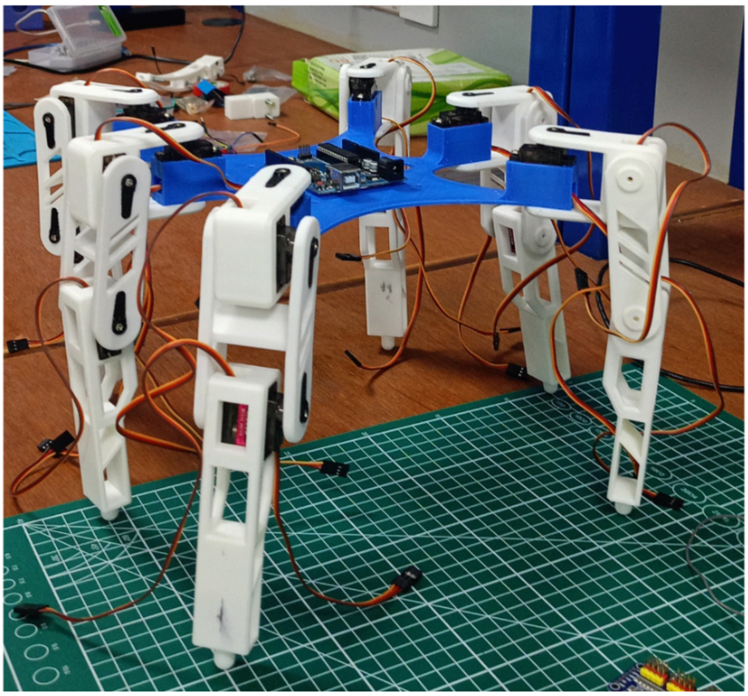



ACRO-BOT MARK-1


ACRO-BOT MARK-1




06



Chapter 4 Investigation/Experiment, Result, Analysis and Discussion

Through trial and error and count we have figured out many things but we have also learned many things. We made the hexapod using servo that were too weak to carry its weights which limited the testing of the Cubli. Running the PID and LQR control on the Cubli made us understand how lacking the motor we used was. The low availability of high torque motors in Bangladesh was very disheartening.

We also underestimated the huge learning curve of taking on a robotics project with almost no experience in the field. It did not help that the Cubli was a project normally done by graduate mechanical engineering students or for PHD thesis.

However, digging through the math it is easy to see how to apply the Cubli to help it balance robots of any shape and size. The edge balancing will be the most useful feature. Corner balancing requires high level math like rotation matrix, and the Kane equation for modelling. But the benefit of corner balancing would add to the overall mobility of the robot is low.

Digging through the math it is also easy to see how a lower gravity environment like space can help the Cubli shine. More testing is required and a generalized method can be created with more effort.

Chapter 5 Impacts of the Project

5.1 Impact of this project on societal, health, safety, legal and cultural

Based on the balancing capabilities of the Cubli, a hybrid hexapod robot has been created which has the potential to change the world as we know it. This robot can be very helpful in search and rescue missions across varied, hard to reach locations where traditional robots using wheels or treads would prove futile. Since the robot possesses both static and dynamic balancing, it can help with rescuing supplies and aiding injured individuals, thus helping save lives and promoting public safety.

Such a robot would improve health and safety by performing dangerous tasks– dealing with a collapsed building or exposed chemicals and nukes– without putting a human's life at risk. Since there is no human intervention, the dangers of physical and chemical harm are eliminated.

The project fulfills the requirements of legal and ethical aspects of responsible robotics. There is no need for surveillance systems which raises privacy concerns. Furthermore, the use of non-humanoid design allows the avoidance of culturally sensitive anthropomorphism termed as Human-Robot Interaction which has the potential for discomfort.

5.2 Impact of this project on the environment and sustainability

This project aims to protect the environment through design and fabrication. The body of the hexapod robot is primarily designed from PLA, which is manufactured from renewable resources, making the plastic both compostable and recyclable. This alleviates the environmental burden during production while ensuring that unused parts do not contribute greatly to long-term plastic waste pollution. Additionally, 3D printing adds to the environmental benefits by further streamlining the materials used in construction through on-demand part fabrication.The robot's modular design allows selective targeted maintenance and enhancement to be made on the robot, rather than having the whole robot replaced.

It can be deployed in sensitive or delicate fields, e.g. for monitoring natural habitats, inspecting farmlands or surveying affected areas after a disaster without damaging the ecosystem of the region. Also, its low-impact, low-footprint, low-energy consumption design and long field service life make it an environmentally friendly tool for long-term field use and contribute to green robotics and responsible technology deployment.

Chapter 6 Project Planning and Budget

The 22-week project began with the Literature Review and Research of Methodology. In Weeks 1–2, we focused on understanding the dynamics of Cubli and its balancing mechanisms. In Weeks 2–5, The requirement analysis and planning were carried out, establishing technical goals and allocating resources. Simultaneously, from Weeks 1–5, the simulation environment was developed using PyBullet to analyze the balance behavior.

In Weeks 3–7, we explored advanced dynamics and motion principles including Newton-Euler formalism, Kane's method, and dynamic modeling strategies. This was followed by Weeks 7–10, which were dedicated to studying LQR, fuzzy logic, and foundational control system strategies.

The mechanical design and 3D modeling of the structure was carried out from Weeks 9–13. Followed by the Core dynamic implementation, particularly on the inverted pendulum model was done from Weeks 14–19. In parallel, Kinematic Modeling and Gait analysis were conducted during the same week, enabling coordinated movement planning.

In Weeks 17–22, system representation and mathematical modeling using Jacobians, state-space models, and transfer functions were carried out. Hardware modeling and integration progressed alongside during Weeks 17–22, involving motor configuration and the hardware assembly. Finally, sensor fusion and final system integration were completed in Weeks 18–22, achieving stable, coordinated control of the robot

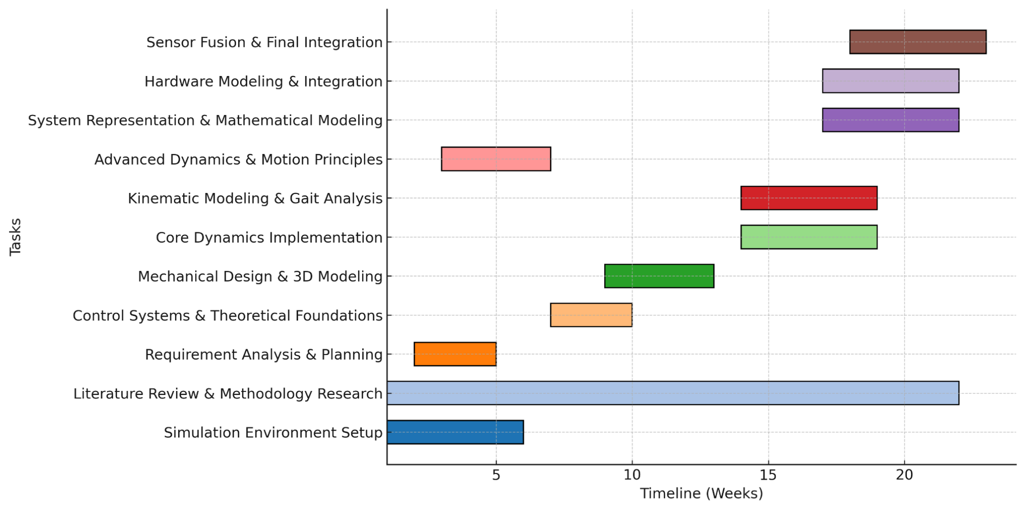


Figure 6.a: Project Gantt Chart (Week 1 - 22)

The budget provides the estimated cost of the essential electronic and mechanical components for our robot in this project. This budget includes twelve distinct components which consist of servomotors, gear motors, an Arduino, a LiPo battery, a Bluetooth module together with control units, power supply elements and structural accessories and bearings along with wiring. The prices maintain unit expenses within reasonable ranges while controlling prototype numbers for balancing reliability with affordability. The overall cost sums to 11,318 BDT which offers an economic and functional budgetary analysis for academic and engineering planning efforts.

TABLE II. COMPONENT-WISE BUDGET SUMMARY

PRODUCT	**UNIT PRICE (BDT)**	**QUANTITY**	**PRICE (BDT)**
SERVO MOTOR	209	18	3762
GEAR MOTOR	950	3	2850
PWM SERVO DRIVER	510	1	510
BEARING	60	18	1080
COUPLER	190	1	190
ARDUINO	349	1	349
BUCK CONVERTOR	99	1	99
LIPO BATTERY	1649	1	1649
BLUETOOTH MODULE	349	1	349
WIRES (SET)	100	3	300
BREADBOARD	180	1	180
TOTAL			**11,318 BDT**

7.1 Complex Engineering Problems (CEP)

TABLE III. COMPLEX ENGINEERING PROBLEM ATTRIBUTES TABLE

Attributes		Addressing the complex engineering problems (P) in the project
P1	Depth of knowledge required (K3-K8)	The project requires knowledge of Electrical Circuits, Electronics (K3), Wireless Communication, Embedded System, Sensors and Instrumentations (K4), Designing and Simulation (K5), Engineering & IT (Circuit Design/Smartphone Application) Tools (K6), Involve Environmental Effects (K7), Scientific Research Papers (WK8).
P2	Range of conflicting requirements	The requirement for structural stability and balance clashes with the limitations of weight for mobility. Lighter materials offer higher speed and power efficiency but decrease mechanical durability, which impacts payload and stability. Servos must be powerful enough to allow movement of the legs, but efficient enough to maintain runtime over time
P3	Depth of analysis required	A range of design approaches may be taken with respect to leg configuration, sensor positioning, control (PID, LQR), and choice of microcontroller. Control power, response time, and control mechanization had to be balanced during simulations and tests evaluating a design in control speed.
P4	Familiarity of issues	Understanding servo control, Inertial Measurement Units (IMU), balance algorithms, and hexapod robot gait generation are critical. The integration of balancing logic into a walking Robot is uncommon and requires searching through academic and open-source robotics literature.
P5	Extent of applicable codes	There is no existing code or standard for this project.
P6	Extent of stakeholder involvement	

		The robot is mainly an academic project, but there are potential investors from the search and rescue, surveillance, and environmental inspection industries. Stakeholder concerns would include balance, adaptability, battery duration, and sturdiness.
P7	Interdependence	The system comprises several interdependent subsystems: servo motors, IMU sensors, Arduino controllers, power supply circuits, and 3D printed mechanical parts.

Table III. demonstrates a sample complex engineering problem attribute.

7.2 Complex Engineering Activities (CEA)

TABLE IV. COMPLEX ENGINEERING PROBLEM ACTIVITIES TABLE

Attributes		**Addressing the complex engineering activities (A) in the project**
A1	Range of resources	This project involves human resources (4 group members), money, modern tools (3D printer, PyBullet, OnShape, Arduino IDE), hardware components (Microcontrollers, Motor drivers, Motors etc.) etc.
A2	Level of interactions	This academic project involves interactions between group members to design the body parts, installing robot parts, designing the control system, also interaction with the instructor to solve the errors faced etc.
A3	Innovation	The innovation of this project is to develop a generalized and simplified algorithm to make the control of a robot, that uses conservation of angular momentum, more simpler and generalised and less resource intensive.
A4	Consequences to society / Environment	This can greatly impact the rescue missions in critical and vulnerable areas and reduce the life risk of rescuers.
A5	Familiarity	Needs to be familiar with the various sensors, microcontrollers, motors, UART communication system, 3D designing tools, control systems, Inverse Kinematics, Robot Gaits.

Chapter 8 Conclusions

8.1 Summary

This project is about the design and fabrication of a small robot using a 3D inverted pendulum. Throughout this project a control system based on a 3D inverted pendulum is developed and integrated with a small legged robot to achieve stable and responsive movement. In order to simplify the integration process, a generic control algorithm was proposed. The algorithm is adaptive in nature and therefore can be utilized on various small robots easily without much calculation or complex tuning. To demonstrate the effectiveness of this algorithm, a robot using a 3D Pendulum Control System is designed and developed. The control algorithms and equations were tested on this robot to evaluate its ability to maintain balance dynamically. To keep this prototype robot cost-effective, locally available hardware is used and expenses are minimized during the design and fabrication phases.

8.2 Limitations

The main limitations of this project are hardware-related. High quality components aren't readily available in the local market and purchasing components from international suppliers is time consuming and expensive. Locally available servo motors aren’t strong enough to lift the total body weight and also the geared motors don't provide the necessary RPM or torque to drive the reaction wheels effectively.

8.3 Future Improvement

Upgrading Hardware Components:

For enhancing performance, subsequent models of the robot may have superior motors and sensors. With the greater number of efficient actuators, larger torque and faster response rate, the robot's dynamic stability and mobility shall improve significantly.

Compact and Lightweight Design:

Mechanical structure with minimal weight and space optimization will help the robot move faster and in a more agile way. Due to lighter weight materials such as carbon fiber or higher-order composite, improved balance might be established through less power motor.

References

1. [1] A. S. Johnson, J. E. Clark, and D. E. Koditschek, "RHex: A simple and highly mobile hexapod robot," *The International Journal of Robotics Research*, vol. 20, no. 7, pp. 616–631, 2001.
2. [2] Y. Zhu, J. Wang, and B. Wang, "Adaptive gait planning for hexapod robot locomotion over unstructured terrain," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 6, pp. 5462–5471, Jun. 2021.
3. [3] S. Guo, X. Yang, and T. Wang, "Design and terrain-adaptive locomotion control of a biomimetic hexapod robot," *IEEE Access*, vol. 8, pp. 201578–201589, 2020.
4. [4] A. Hossain and M. M. Rahman, "Hexapod leg coordination using simple geometry: Tripod gait and inverse kinematic approach," *International Journal of Robotics and Automation*, vol. 36, no. 2, pp. 115–123, 2021.
5. [5] M. Rüegg, T. Muehlebach, and R. D'Andrea, "The Cubli: A cube that can jump up and balance," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3722–3727, Nov. 2013.
6. [6] Y. Zhu, C. Wang, B. Liu, and Q. Chen, "Adaptive Gait Planning for Hexapod Locomotion Over Unstructured Terrain," *Robotics and Autonomous Systems*, vol. 139, pp. 103716, 2021.
7. [7] H. Guo, L. Zhang, Y. Wang, and Y. Yang, "Design and Implementation of a Terrain-Adaptive Biomimetic Hexapod Robot," *IEEE Access*, vol. 8, pp. 184921–184931, 2020.
8. [8] H. Rüegg, T. Gmür, M. Huber, and R. D'Andrea, "Cubli: A Reaction Wheel-Based Cube That Can Jump Up and Balance," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Tokyo, Japan, 2013, pp. 3722–3727.

9. [9] Belascuen, Gonzalo & Aguilar, Nahuel. (2018). Design, Modeling and Control of a Reaction Wheel Balanced Inverted Pendulum. 1-9. 10.1109/ARGENCON.2018.8646093.