INDIAN INSTITUTE OF TECHNOLOGY KANPUR

ROBOFEST GUJARAT 2023

Ideation Report - HEXAPOD

Mentor

Prof. K. R. Guruprasad, Associate Professor, IIT Kanpur

Team

Aniket Sen

Aryan Kumar

Emaad Ahmed

Harshit Bhushan

Snehal Kumar

Contents

Chapter 1

Introduction

1.1 About Us

We are a group of students working as a team in the robotics club of our institute, IIT Kanpur. We are highly passionate, highly motivated, and excited about developing a “Hexapod”. The team is made up of aspiring engineers from a variety of IIT-K departments and study programmes. For Robofest Gujarat 3.0, we are striving hard to get a practical understanding regarding the development of the six-legged bot, which can move, rotate and climb obstacles.

1.2 Why ”Hexapod”?

A multi-legged robot has the ability to move on irregular surfaces and can be used for various purposes depending upon the scenario and the number of legs. Multi-legged bots have more static stability while moving on irregular surfaces than wheeled or tracked bots. Hexapod is a multi-legged robot with six legs, one of the most stable-legged robots. Hexapod has greater flexibility and stability to move on irregular platforms using three legs. It has a smaller size, higher stiffness, dynamics, precision, and greater flexibility.

1.3 About the robot

The project aims to build a six-legged walking robot capable of basic mobility tasks such as walking forward, backward, rotating in place, and raising or lowering the body height. The legs will be of a modular design and will have three degrees of freedom each. This robot will serve as a platform onto which additional sensory components could be added or which could be programmed to perform increasingly complex motions. The direct and inverse kinematic analysis for each leg has been considered to develop an overall kinematic model of the robot. The trajectory of each leg is also considered for both swing and support phases when the robot walks with a tripod gait in a straight path.

This report discusses the components that make up our final design.

Chapter 2

Problem Statement

2.1 Primary Objective

A locomotion-based Hexapod (6 legs). The main task is to control the locomotion of the hexapod, make it walk some distance (forward, backward, right, and left) and make it climb the obstacle with at least 2 repetitions. No readymade kits/chassis are allowed. No external controls are allowed.

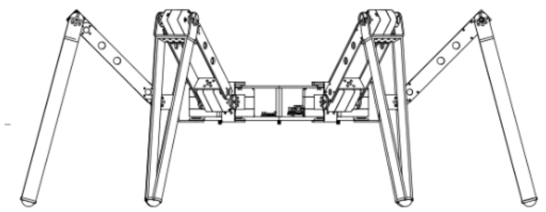
2.2 Grand finale Objective

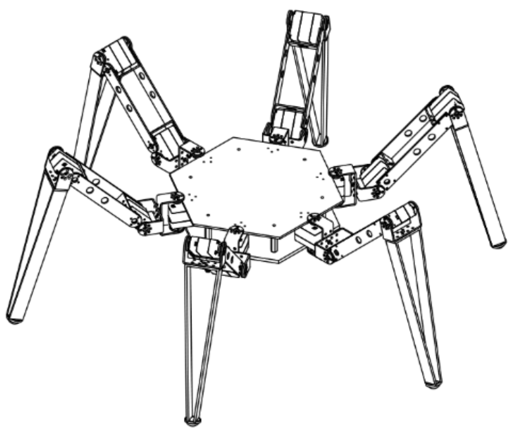
Wirelessly controlled locomotion-based hexapod that can move a distance at any degree from its home position. It should be able to travel a distance of 500 m and climb up as well as get down 2 obstacles/ stairs that are 6 inches high in between the start and stop points. It needs to be battery-operated only.

Chapter 3

Technical details

3.1 Full Assembly





To keep the center of gravity at the midpoint of the body, the legs are fixed at the opposite ends so that the robot can move continuously and balance itself. The Arduino and motor driver are fixed on the middle part of the body.

3.2 Assembly

3.2.1 Full Assembly

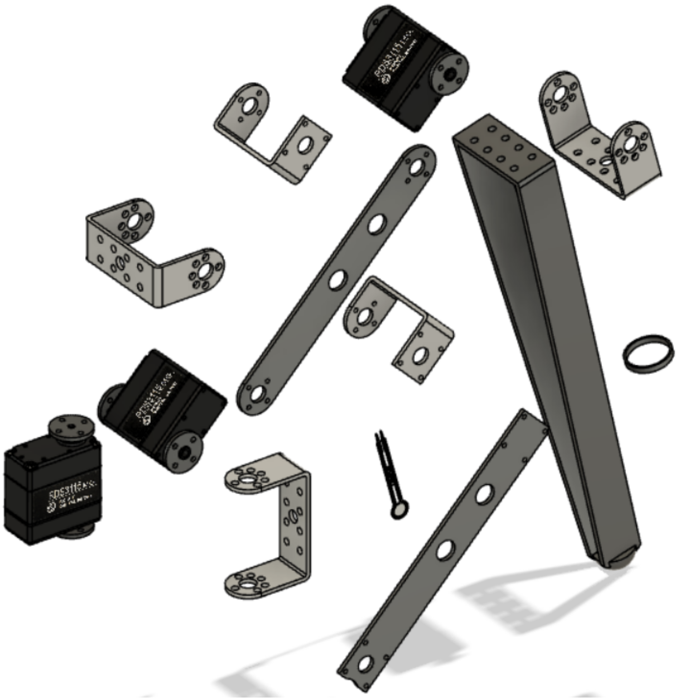
The main chassis of the bot is hexagonal in shape and consists of two plates made up of Aluminium. Which ensures lighter weight and more strength. Two plates are separated by 40mm standoffs. The middle spaces are used to house the servo motors as well as other electrical components.

Dimensions: 70x70x27 cm

3.2.2 Arm

Each leg of the robot has the same functions and six degrees of freedom. It constitutes three servo motors, of which two move in a vertical plane while one in the horizontal plane. The only difference between the legs is the motion algorithm. The legs are designed in such a way that they can walk on irregular platforms and can climb terrains.

Lengths: Coxa:6cm Femur:16cm Tibia:26cm



Exposed Arm of the Bot

3.3 Components

3.3.1 Structure

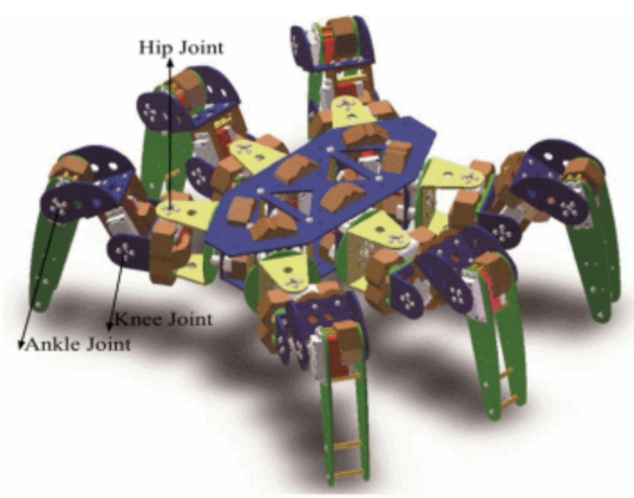
1. 5mm Aluminium Sheet(205cm3)
2. 3mm Aluminium Sheet(400cm3)
3. Servo Motor Mounts(*30)
4. M2, M4 bolts and nuts
5. M4x40mm Male-female standoffs

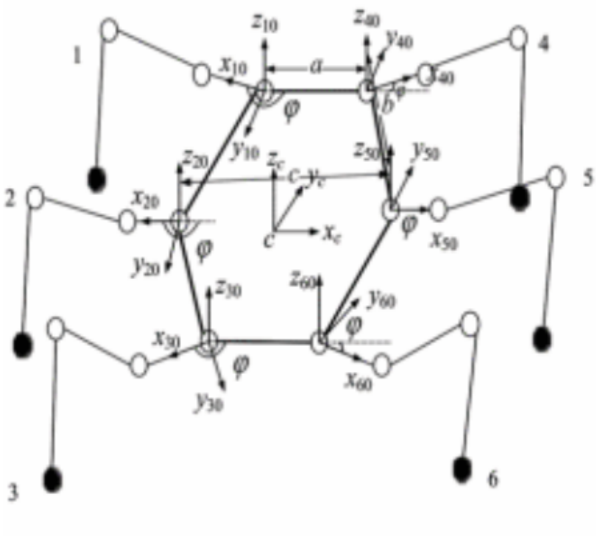
3.3.2 Electronics

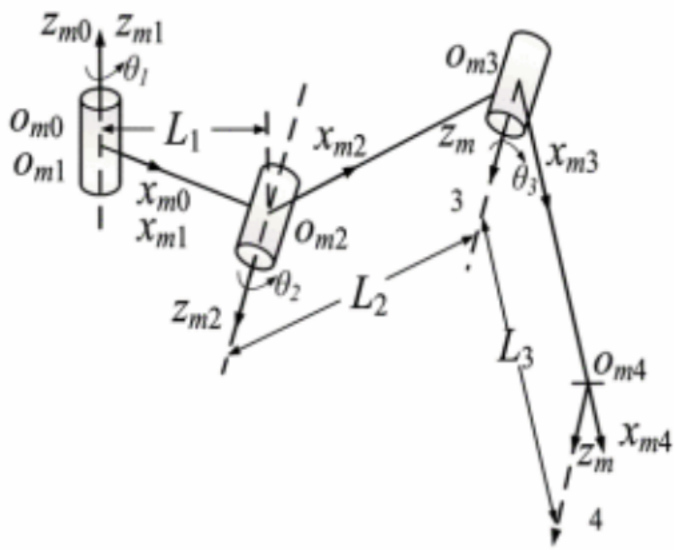
1. RC Transmitter and receiver
2. Adafruit 16-Channel 12-bit servo shield
3. Arduino Uno
4. Servo Motor

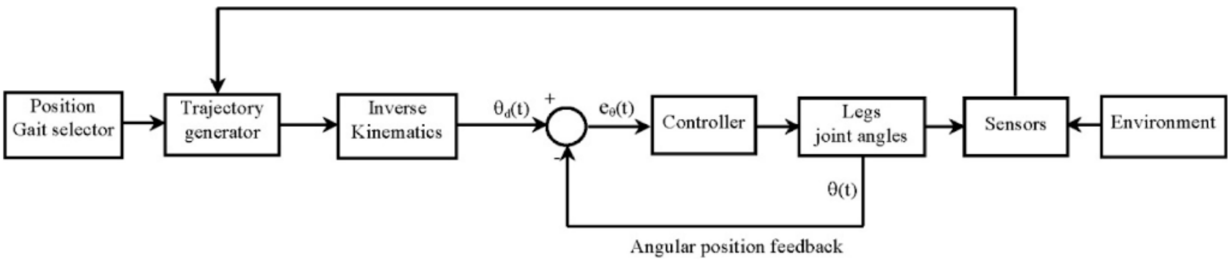
Chapter 4

Functioning





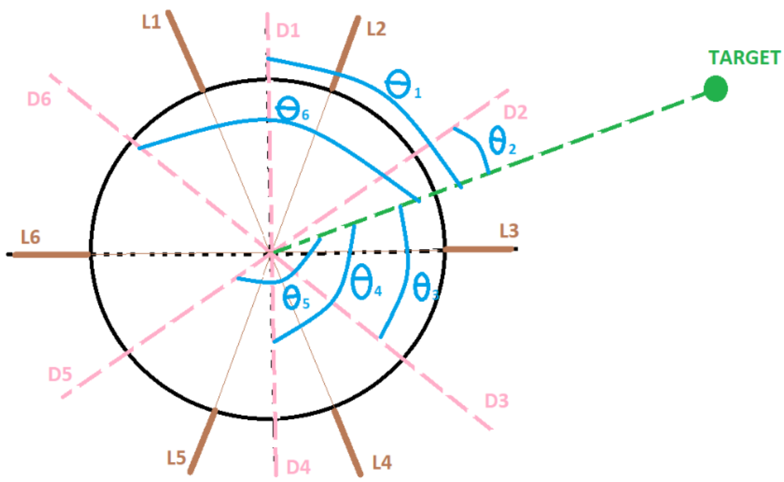




4.1 Gait styles

The hexapod walking motion, or gait, has a number of advantages over bipedal or wheeled locomotion styles. Although wheeled systems are typically faster on level ground, hexapedal locomotion produces the fastest movement of all legged robots. Unlike wheeled robots, hexapods can safely maneuver through uneven terrain and over smaller obstacles. Hexapods can also turn without requiring forwards or backward motion. They maintain superior stability when walking by keeping their center of mass within a triangular area created by any three planted legs.
Hexapedal gait styles can be broken down into two subcategories; rectangular leg orientations and hexagonal leg orientations. Hexagonal leg orientations generally allow for more complex movements, easier turning, and increased mobile stability. Alternatively, rectangular leg orientations can produce motion with simpler locomotion systems at the cost of sacrificing some stability.
The most common hexapedal gait style is the 3+3 "tripedal" gait, which separates the legs on either side of the robot into two triangular groups. Each group contains two outside legs on one side of the robot and one middle leg on the opposite side of the robot. During motion, the groups alternate moving forwards or backward, ensuring that three legs are in contact with the ground at all times. This style of gait produces the fastest hexapedal motion. Therefore, we have utilized it.

4.2 Navigation

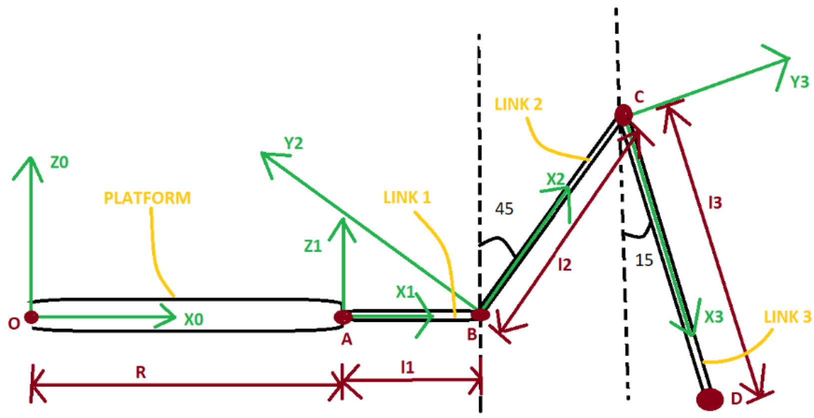


The bot has 6 possible directions of motion which are symmetric (60 degrees between

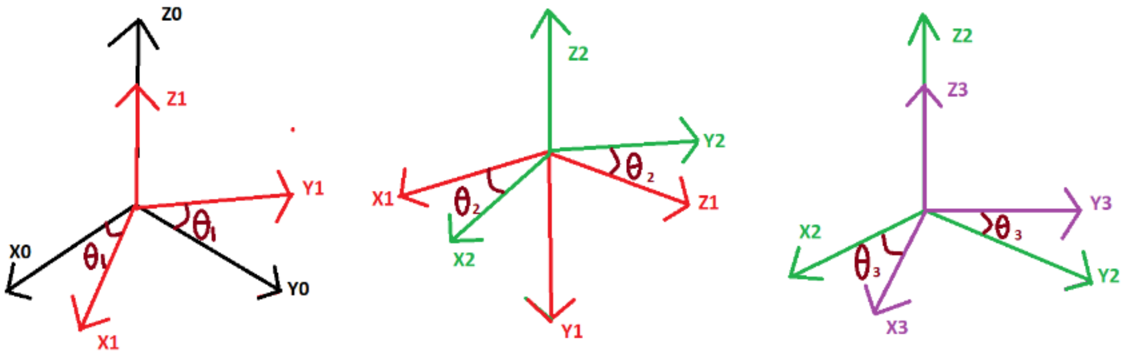
each direction)
Li → leg of the bot
Di → direction of travel
θi → angle between Di and the line joining target and center of the bot
The bot will execute two sets of algorithms to reach a target point on the same plane. It will first turn itself towards the target (turning) and then it will reduce the distance between the target and the center to zero or a threshold value (translation).
Turning:- The bot will align the nearest direction of travel with the traveling direction. It will do so by first measuring the angle from the green line to each direction of travel Di to get θi. It will then execute the following:
θi = θi (θi ¡ 180)
= 360 - θi (θi ¿ 180)
Thereafter the direction giving the lowest θ i will be aligned with the target direction.

4.3 Inverse kinematics

We shall look at inverse kinematics of one hexapod arm. We shall apply the Denavit-Hartenberg algorithm for assigning coordinates to each link of one hexapod arm.



{E0, O, X0} → BFCS to platform
{E1, A, X1} → BFCS to link1
{E2, B, X2} → BFCS to link2
{E3, C, X3} → BFCS to link3
[R1], [R2], [R3] → rotation matrices
{ E0, O, X0 } — [R1] → $\{E1, A, X1\} - - - [R2] \rightarrow \{E2, B, X2\} - - - [R3] \rightarrow \{E3, C, X3\}$



$$R1 = \begin{bmatrix} \cos(\theta 1) & \sin(\theta 1) & 0 & R\cos(\phi) \\ -\sin(\theta 1) & \cos(\theta 1) & 0 & R\sin(\phi) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} R2 = \begin{bmatrix} \cos(\theta 2) & 0 & \sin(\theta 2) & l1 \\ -\sin(\theta 2) & 0 & \cos(\theta 2) & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R3 = \begin{bmatrix} \cos(\theta 3) & \sin(\theta 3) & 0 & l2 \\ -\sin(\theta 3) & \cos(\theta 3) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$Instaticstate,$

$\theta 1$ depends on position of arm on the platform

$\theta 2 = 45$ degrees

$\theta 3 = 105$ degrees

Point D is the foot of the hexapod leg on the ground. To move the legs and the bot, we will decide where the initial and final position of the feet of the legs will be in the robot base frame. Using inverse kinematics, we will find the initial(θinitial) and final(θfinal) angles of the joints(servo motors) of each leg.

[rD]E0 = [R1(θ1)]E0[R2(θ2)]E1[R3(θ3)]E2[rD]E3

[rD]E3 = (l3)X3

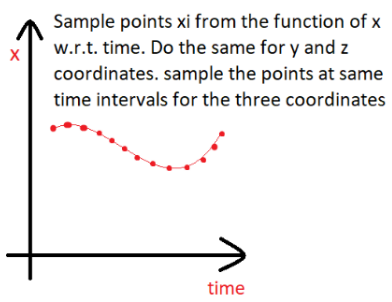
We decide on [rD]E0 based on the motion required

We are then left with 3 equations and 3 variables $\theta 1$, $\theta 2$, $\theta 3$ which we solve to get the required servo angles.

Thereafter we will apply trajectory planning with which we will basically come up with a function of θ w.r.t time(t). Using this we will find out how each servo will rotate to go from the initial(θinitial) to the final(θfinal) position

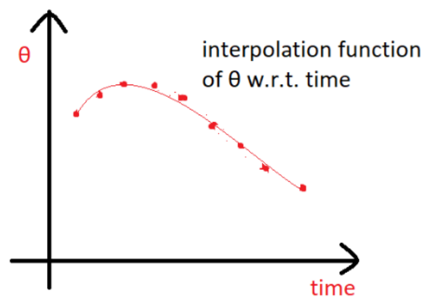
4.4 Trajectory Planning

We employ trajectory planning to find out how each servo will rotate with time to make the leg move in a desired manner. We will have an initial and a final angle for the servo based on the initial and final positions of the end effector using inverse kinematics. We then have to come up with a function of the angles w.r.t. time. We take some points (xi, yi, zi) lying on the locus of the end effector. Based on the speed of the end effector we want, we plot the points as a function of time. For our model, we would prefer constant speed of the end effector.



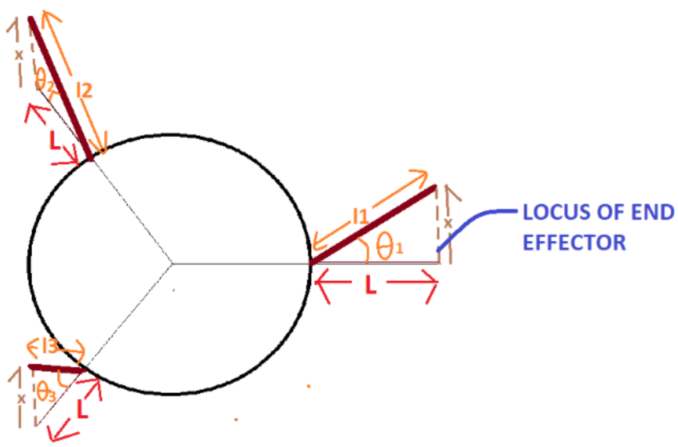
Now on each sample point we apply inverse kinematics to get the angle for the servos at each point. We plot the angles of the servos as a function of time.

Thereafter we interpolate the points to obtain a function of the servo angle as a function of time. We will then input these servo functions in software to generate the desired limb motion.



4.5 Gait motion planning

4.5.1 Forward Propelling



The three alternate feet on the ground will propel the body forward. In their static state they would be at radial length L as shown in the diagram. However to propel the legs will come to their static state from their respective length li at angle θi as shown. The locus of the tip of feet will be a straight vertical path of distance x as shown. In this way the bot is propelled forwards a distance x and no lateral motion is brought into play.
We will use the following equations:

1. x = L tan(θ1)

2. x = l1sin(θ1)

3. x = [$\sqrt{3}(l2cos\theta2$ - L) + l2 sin θ2] / 2

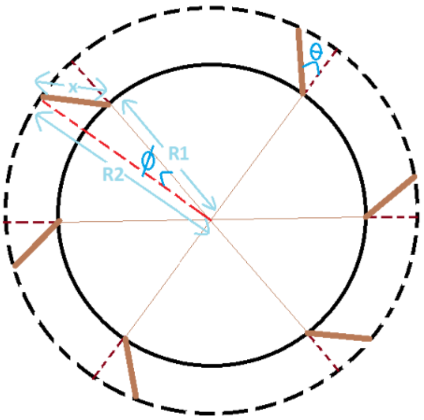
4. x = [$\sqrt{3}(L - l3cos\theta3)$ + l3 sin θ3] / 2

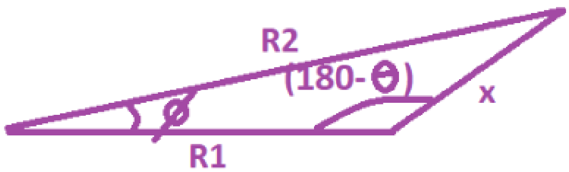
5. l3 cos (θ3 - 600) - L / 2 = 0

6. l2 cos (θ2 + 600) - L / 2 = 0

Now we decide on a propelling distance 'd'. We then have to follow the propelling motion such that x initial = d and x finial = 0.
We know the value of x and L. We have a total of 6 equations and 6 variables, solving which we will get the values of li and θi.
Thereafter we apply trajectory planning.

4.5.2 Turning





To turn by an angle ϕ, all the legs will rotate by angle θ and simultaneously extend the legs to length x so that the end effectors do not slip on the ground. The angle θ and the length x required for rotating by a given angle ϕ can be found using the following relations derived from sine and cosine rules:

1. x2 + x[2(R1 + L)cos(θ)] + [(R12 + L)^2 - (R2)^2] = 0

2. θ = sin-1[R2sin(ϕ)/x]

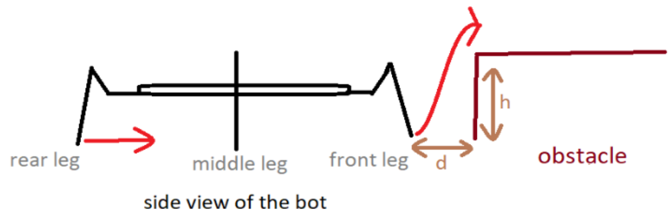
We will decide on a constant angular speed of rotation i.e. ϕ as a linear function of time.

4.5.3 Dealing with Obstacles

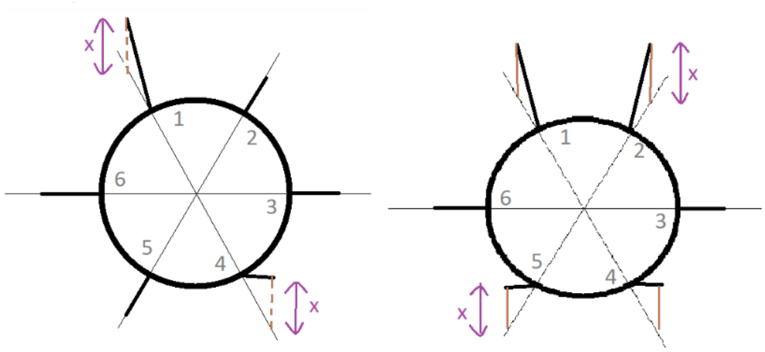
When the robot encounters an obstacle with a threshold maximum height and a threshold minimum height, it will execute a climbing sequence to overcome the obstacle.
In climbing motion, unlike walking motion in which there are 3 legs moving simultaneously, there will be 2 legs (diagonally opposite moving at a time. This is done to gain extra stability at uneven terrain, by using the support of four legs instead of three.

Following are the necessary motion algorithms that will be inbuilt in the bot for dealing with obstacles:

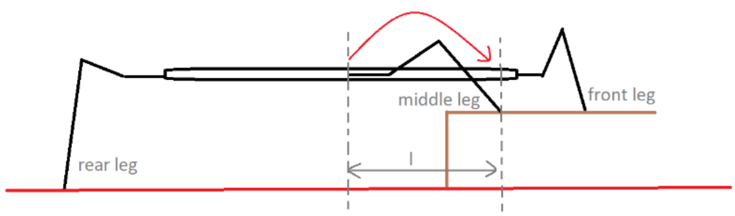
1. **<u>GAIT 1:</u> The bot on the flat ground just encountered an obstacle**



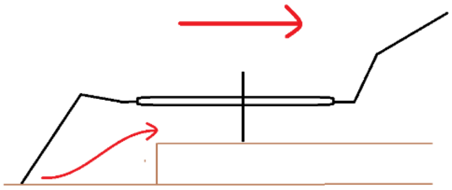
First leg 1 goes up height h and goes straight forward a distance x while leg 4 goes forward distance x simultaneously. Keep x = d + y0 where y0 is some threshold value we will decide. Thereafter leg 2 moves like leg 1 and leg 5 moves like leg 4.

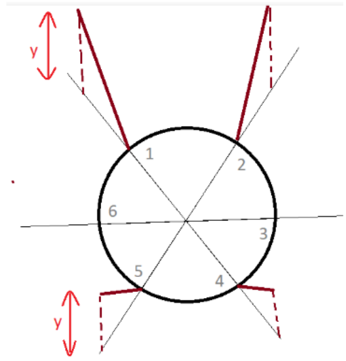


Legs 3 and 6 will move up a distance h and move forward a distance x+l (where l is some threshold value depending on d and R) consecutively to rest on the obstacle.
Then the centre of the bot will push itself upwards height h and forward distance x. This will take place when all the legs simultaneously go down by distance h and the end effectors move straight back by a distance x.



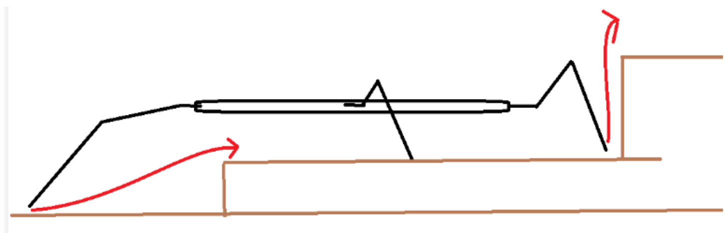
2. **<u>GAIT 2:</u> The front half of the bot is on the obstacle and the rear part has to be moved up / Bot moves straight forward**





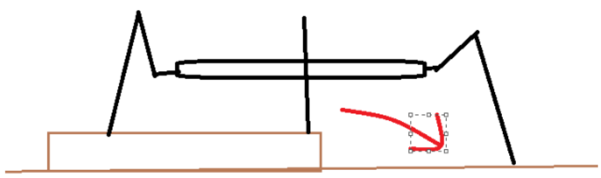
Legs 1 moves forward a distance x and leg 4 moves up a height h and moves forward a distance x. Then legs 2 and 5 move in the same manner. Legs 3 and 6 move forward by distance x-l. Then the centre of the bot will push itself upwards height h and forward distance x.

3. **<u>GAIT 3:</u> The front half of the bot is on the obstacle and it has to climb a consecutive obstacle**



It will follow a similar algorithm as situation (1).Leg 1 and leg 5 will both go up a height h and go forward distance x. And then legs 2 and 4 will follow the same. Thereafter, legs 3 and 6 will go up a height h and go forward a distance x consecutively. In this way, legs 1, 2, 3 and 6 will be on obstacle-2 and legs 4 and 5 will be on obstacle-1. Finally, the centre of the bot will go up a height h and move forward a distance x.

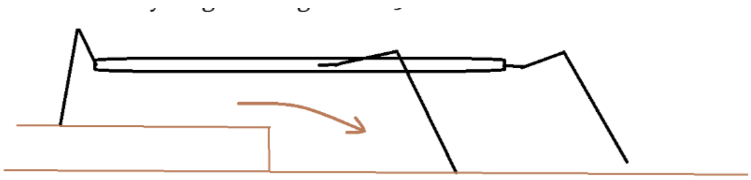
4. **<u>GAIT 4:</u> Whole bot is on the obstacle and the bot has to climb down**



Leg 1 will move forward by x and downwards by h while leg 4 will move forwards by x. Legs 2 and 5 will move in the same manner. Legs 3 and 6 will move forward by x. Thereafter the centre of the bot will move forwards by x.

5. **<u>GAIT 5:</u> The rear half of the bot is on the obstacle and the bot has to brought down / bot moves straight forward** Leg 1 will move forward

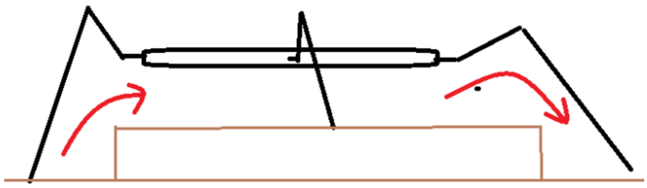
by distance x while leg 4 will move forward by distance x and downwards by height h. Legs 2 and 5 will follow suit.



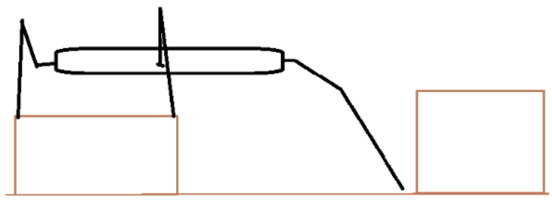
Legs 3 and 6 will move forward a distance x-l. Thereafter, the bot will push itself forward by distance x and downwards by height h.

6. **<u>GAIT 6:</u> The front half of the body is on the obstacle and the bot has to climb down**

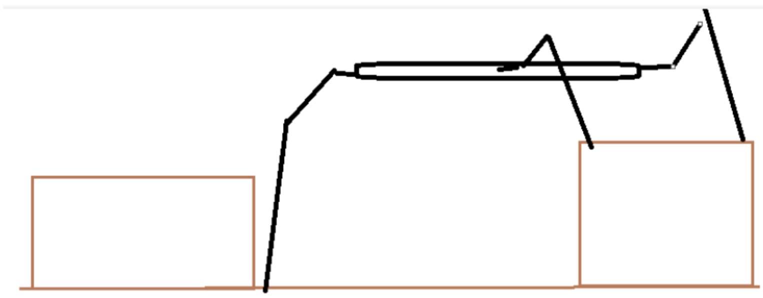
   Leg 1 will move forward by distance x and downwards by height h while leg 4 will move upwards by height h and move forwards by distance x. Legs 2 and 5 will move in the same manner. Legs 3 and 6 will move forward by distance x. Thereafter the bot will push its centre forward by distance x.



7. **<u>GAIT 7:</u> Rear half of the bot is on the obstacle and the bot has to climb up**



   Leg 1 will move upwards by h and forward by x while leg 4 will move forward by x and move down by height h. Legs 2 and 5 will move in the same manner. Legs 3 and 6 will move forward by length x.



4.6 Control Sequence

For controlling the bob, there will be the following modes on the controller:

1. A 360 degree knob to specify direction
2. Start button to start moving in the selected direction
3. Obstacle button

The bot will move on receiving commands from the controller using combinations of gait motions provided above.
For obstacle detection, the bot will rely on a set of sensors and a feedback loop which will detect the obstacles and provide feedback based on which the bot will execute certain gait motions (from above specified gaits) to overcome the obstacle.

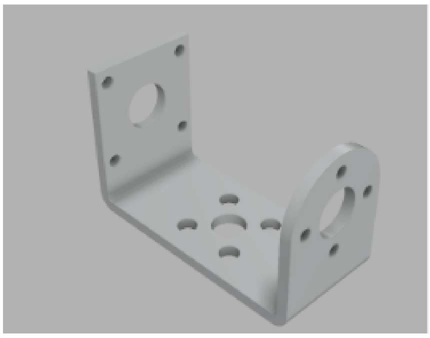
Chapter 5

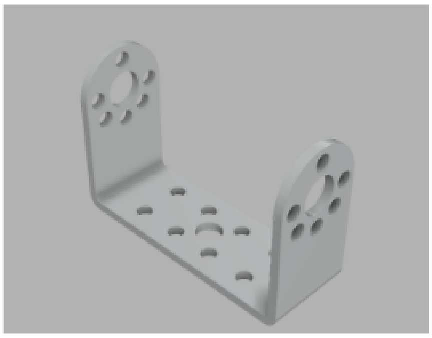
Methodology

5.1 Mechanical Hardware

We have used Autodesk Fusion 360 to design and model the base of the Hexapod.

We used a range of simulation tools and stress analysis, to ensure the structural integrity and performance of the chassis. The chassis will comprise two thick circular aluminium plates and join them together via small rods and space in between them would be used to contain the various sensors and board. We settled on the chassis configuration through several iterative design and simulation procedures and found the design ideal for the bot.





The materials which we would be using to build the core part of our model would be Aluminium. Aluminium is sturdy and lightweight, we thought it would be the best for the bot's body. We would be using the biaxial structure for the arms to bind them effectively with the servos. The inner leg section is two pieces separately machined and then bonded together as joints for movement. The type of bonding agent has not been finalised yet. The leg sections will be bolted to the servo motors using small (approx. M4) bolts at the servo mounting flanges and the servo mounting disk (attached to the output shaft).

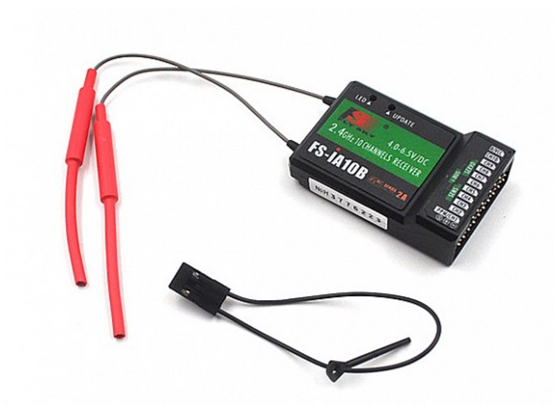
5.2 Electrical Hardware

The Electrical components used for this model are:

5.2.1 RC Transmitter and receiver

RC (radio-controlled) transmitter and receiver are used to wirelessly control a hexapod robot. The transmitter is typically a handheld device that the user holds and uses to send commands to the robot. The commands are transmitted via radio waves to the receiver, which is located on the hexapod. The receiver then processes the commands and sends the appropriate signals to the hexapod's control unit to control its movement.

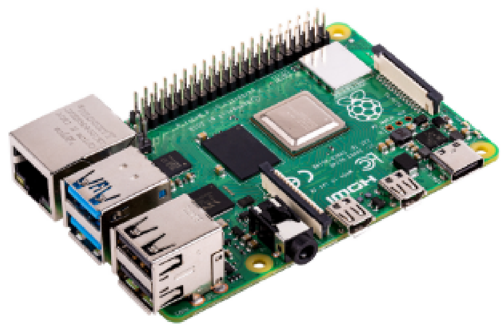
We chose the FlySky FS-i6 controller as its 10 channels which can be used to control the movement of the robot, such as moving forward, backward, left, and right, turn, climb, descend, etc. This will also provide us room to add more controls and equipments.



The receiver Fs-Ia10B is connected to the hexapod's control unit and receives the commands from the transmitter. It decodes the commands and sends them to the control unit which then generates the control signals for our servos.

5.2.2 Raspberry Pi 4

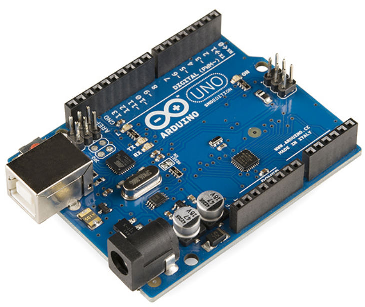
A companion computer is required for processing various sensor data and object detection and identification. We have chosen Raspberry Pi 4 Model B with 8 GB ram as our primary computer. With its computing power, we can do almost all required processes onboard the hexapod. This helps us in eliminating the need to send raw data to a surface computer then process it there and send the required action back to it. Its open-source nature and universal help support help one to perfectly implement it in their use environment.



5.2.3 Arduino Uno

An Arduino Uno microcontroller is used; it has 14 digital pins, and 6 analog pins. Out of 12 digital pins, 6 can also be used as PWM inputs. It processes all the parameters and executes the respective action as per the program written into it.

The Arduino uno will be used to control the servo motors that drive the movement of the hexapod's legs. It can send commands to the motors, telling them which position to move to, and monitor their movement to ensure that they are following the desired trajectory.



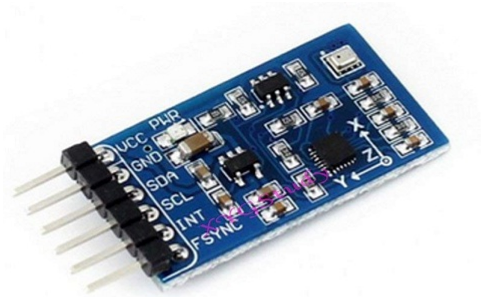
The Arduino uno will process sensor data from the hexapod's sensors, such as infrared sensors, ultrasonic sensors,imu sensors or cameras, and use that data to make decisions about how the hexapod should move.

5.2.4 Adafruit 16-Channel 12-bit servo shield

Two Servo shields are used to receive a command signal from the Arduino, amplifies the signal and transmits the electric current to the Servo motor. This Servo shield can safely drive up to 16 servos. The PWM controller will drive all 16 channels simultaneously with no additional Arduino processing overhead.

5.2.5 IMU Sensor

IMU (Inertial Measurement Unit) sensors are used in hexapod robots to determine the robot's position, orientation, and movement. IMU sensor can provide a complete 3D representation of the robot's movement and orientation. The data provided by IMU sensor will be used by the control unit to generate control commands that are sent to the servos to control the movement of the robot.



We are using MPU6050 which is used to stabilize the chassis of hexapod in uneven terrain and keep it balanced by measuring hexapod's angular velocity and acceleration.

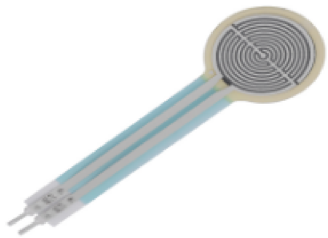
5.2.6 Servo Motor

Servo motors are designed to rotate to a specific position and maintain that position, which makes them well-suited for controlling the movement of the legs in a hexapod.

Eighteen dual shaft servo motors are used in this project. These motors are very powerful and have metal gear. They weigh about 65 grams, and have a stall torque value of 35 kg-cm. Operating voltage is between 6 volts–8.4 volts. Servo motors are connected to servo shields and are powered by external power supply.

5.2.7 Force Sensitive Resistor

Throughout the exploration of an unknown environment, one of the unwanted events that could arise is to place a leg without making ground contact when needed. This situation could lead to a stumble with undesirable consequences. To avoid such a possible disastrous event, a tactile sensor is attached at the bottom of each leg.



5.2.8 LIDAR (Laser Imaging,Detection and Ranging)

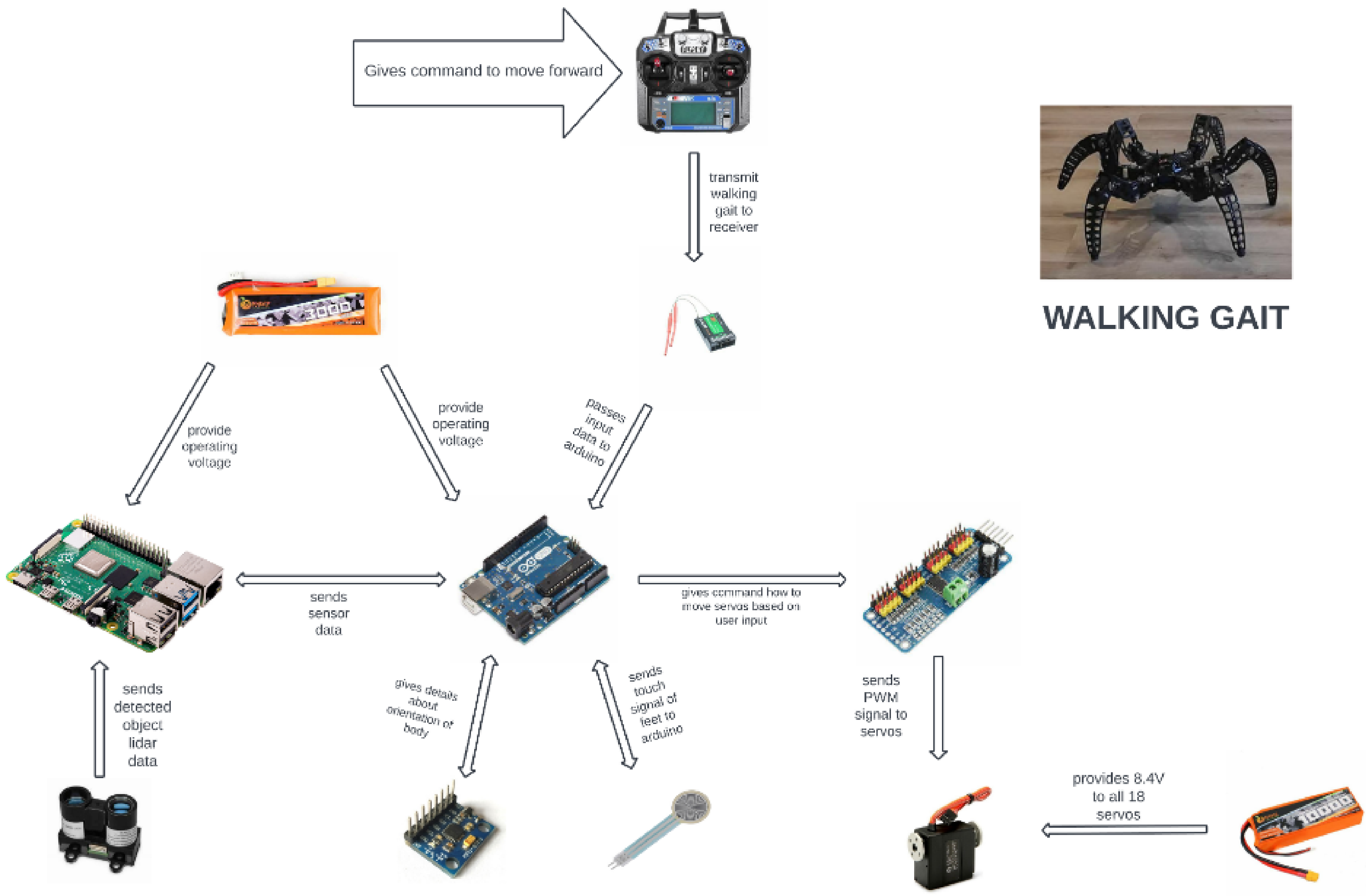
We are using Garmin lidar lite v3 lidar sensor for detecting and identifying the object.We will use the data generated by lidar to get an estimation about the height of object. It is a method for determining ranges by targeting an object or a surface with a laser and measuring the time for the reflected light to return to the receiver.

5.2.9 Power Management System

Knowing that the 18 servos require the most power and that they should be able to demand variable peaks of current, i.e. depending on the load the robot should overcome,we will power the actuators from a separate battery pack. Here to 10000 Mah lipo batteries are acquired which deliver the necessary 8-9 V for all the servos.

The rest of the equipment is fed by a Li-Po rechargeable battery with three cells , thus providing a voltage of 5V and a capacity of 3000 MAh.We are using lithium polymer battery as it is a lower cost and lower capacity battery made from soft lithium polymer cells good for use in the hexapod.

Flowchart showing the Hexapod's electrical hardware



5.3 Software

The software part of a hexapod robot is responsible for controlling the functionality and behavior of the robot. It is used to implement the control algorithms, motion planning, and communication protocols that are necessary for the hexapod to operate.

Control algorithms: We have to implement control algorithms that are used to generate control commands for the servos for different gait sequences. These algorithms will include inverse kinematics etc.

Sensor processing: The software part is used to process the data from the sensors and use it to determine the robot's position, orientation, and movement.

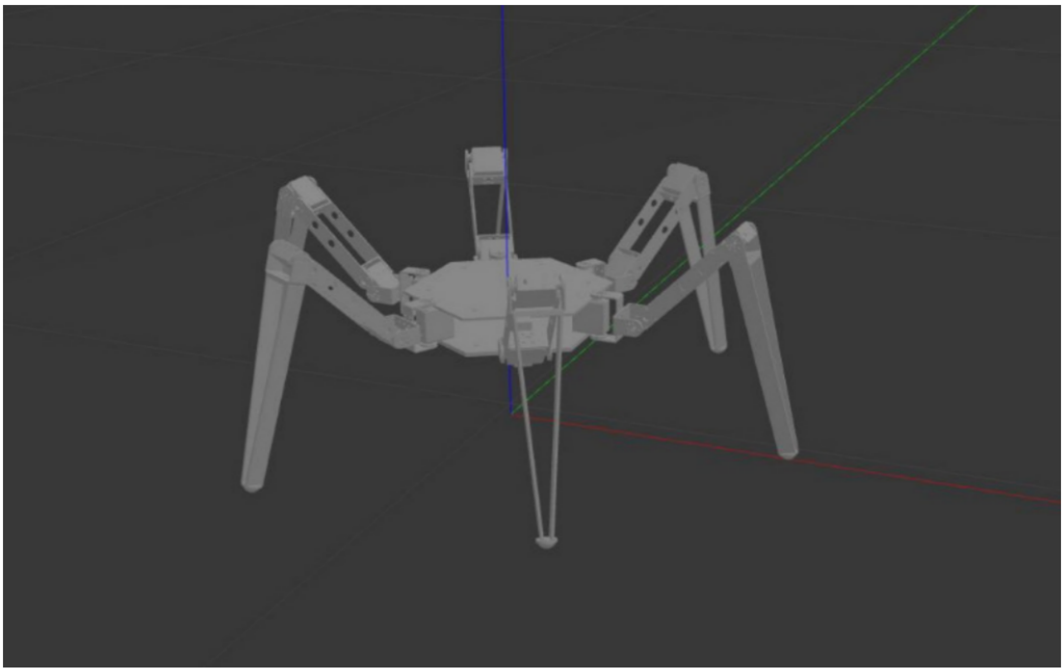
Communication: It is very important to implement communication protocols that allow the hexapod to receive commands from the rc transmitter.

Before moving on to actual bot, we have to test and simulate our bot virtually so that we can minimize the error while implementing it in real world. The methods which we are going to test/simulate out bot and code are:-

Virtual testing: In this we will be simulating the hexapod's movements and environment using Gazebo/Matlab-simulink, which can simulate the dynamics of the robot and its environment. Virtual testing will be useful for testing the robot's movements, stability, and sensor performance in a simulated environment. Simulation will be used to evaluate the robot's performance under different conditions, such as different terrains or payloads. This will allow us to optimize the hexapod's design and control algorithms further to improve its performance. This will also be useful for testing control algorithms as this will save time and resources by identifying any issues or bugs in the algorithm before they occur on the actual robot.

Emulators: We will be testing our code for communication, control, sensor processing etc, in Arduino emulators that will allow us to run and test our code without the need of the physical board, this will be useful for debugging the code and testing it before uploading it to the board.

The simulated image of the project in GAZEBO



Chapter 6

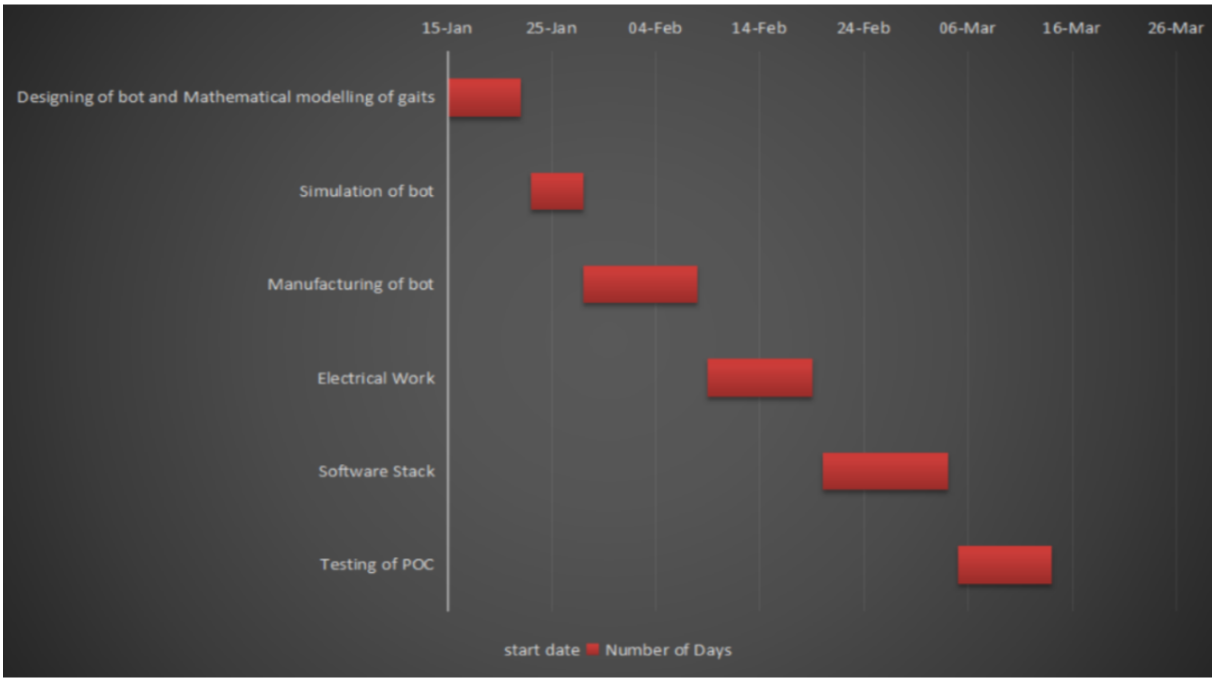
Applications In Society

Hexapod robots are a programmable type of robot with six legs attached to the robot body. The legs consist of servo motors and these servo motors are programmed in such a way that the robot can move within its space.

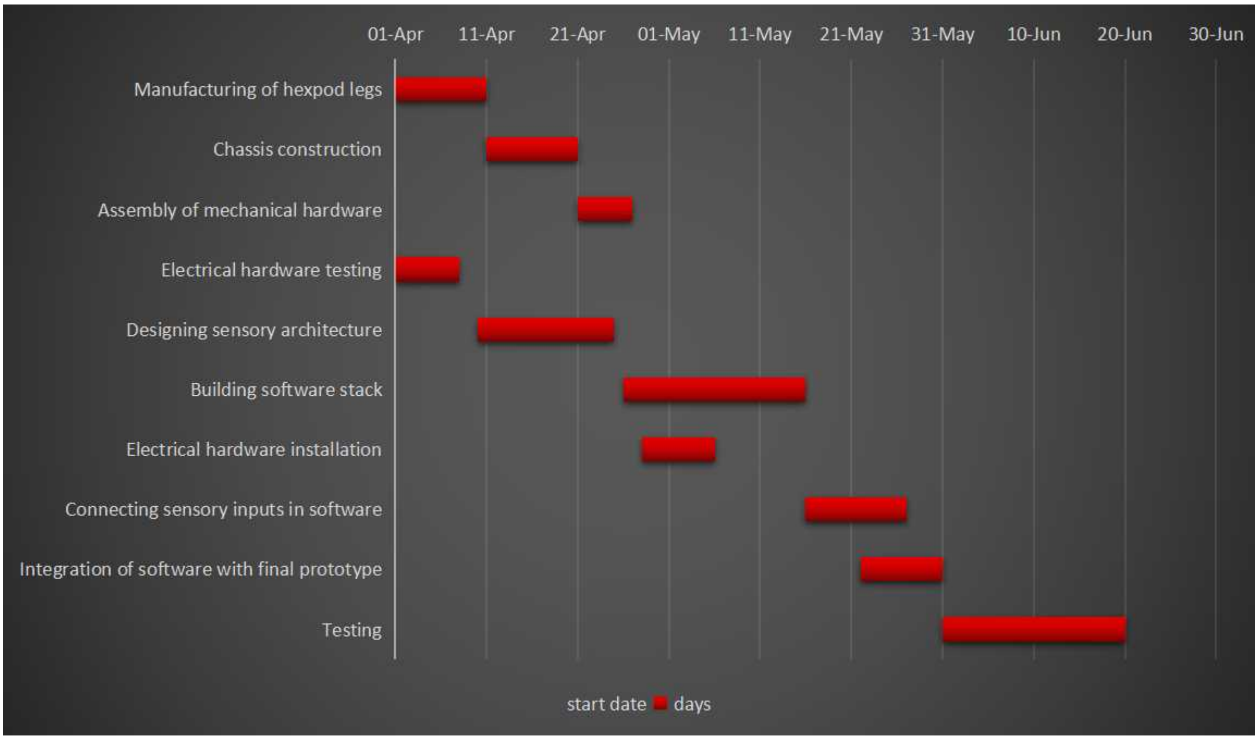
- Hexapod robots are suitable for terrestrial and space applications.
- Hexapod robots have various characteristics which include omnidirectional motion, variable geometry, good stability, access to diverse terrain, and fault tolerant locomotion.
- The main advantage of hexapod robots over wheeled robots is that they can climb over obstacles. In fact, the use of wheels or crawlers limits the size of the obstacle that can be climbed to half the diameter of the wheels whereas the legged robots can overcome obstacles that are comparable with the size of the machine leg.
- Hexapod walking robots have greater mobility in natural surroundings and also benefit from a lower impact on the terrain. It is especially important in dangerous environments like mine fields, or where it is essential to keep the terrain largely undisturbed for scientific reasons.
- Hexapod legged robots have been used in exploration of remote locations and hostile environments such as seabed, in space or on planets in nuclear power stations, and in search and rescue operations. Beyond this type of application, hexapod walking vehicles can also be used in a wide variety of tasks such as forest harvesting, in aid to humans in the transport of cargo, as service robots and entertainment.
- They have a number of advantages over wheeled, quadrupedal or bi-pedal robots. While wheeled robots are faster on level ground than legged robots, hexapods are the fastest of the legged robots, as they have the optimum number of legs for walking. Hexapod robots can traverse uneven ground, step over obstacles and choose footholds to maximise stability and traction.
- Having manoeuvrable legs allows hexapods to turn around on the spot. The robot's centre of mass stays consistently within the tripod created by the leg movements, which also gives great stability.

Chapter 7

Timeline



Timeline of POC



Timeline of Prototype

Chapter 8

References

Hasnaa El Hansali and Bennani Mohammed, "Gait Kinematic Modeling of a Hexapod Robot(IREME)", Mohammed V University in Rabat, 2017 14th Internation Multi-Conference on Systems, Signals and Devices. [Online]

Jianbo Sun, Jie Ren, Yinglian Jin, Binrui Wang and Dijian Chen, "Hexapod robot kinematics modeling and tripod gait design on the foot end trajectory", Macau, 2017 IEEE International Conference on Robotics and Biomimetics(ROBIO). [Online]

Duan Xingji et al., "Tripod gaits planning and kinematics analysis of a hexapod robot", Control and Automation 2009, Christchurch, New Zealand, 2009 IEEE International Conference on Control and Automation.

Shoulin Xu, Bin He and Heming Hu, "Research on Kinematics and Stability of a Bionic Wall-Climbing Hexapod Robot", Applied Bionics and Biomechanics, Volume 2019, Article ID 6146214, 17 pages, April 2019. [Online]

Y. Fan and Y. Zhang, "Six-legged Walking Robot's Gait Planning and Kinematics Analysis under the Environment of Concave Ground," presented at the 4th International Conference on Mechatronics, Materials, Chemistry and Computer Engineering.