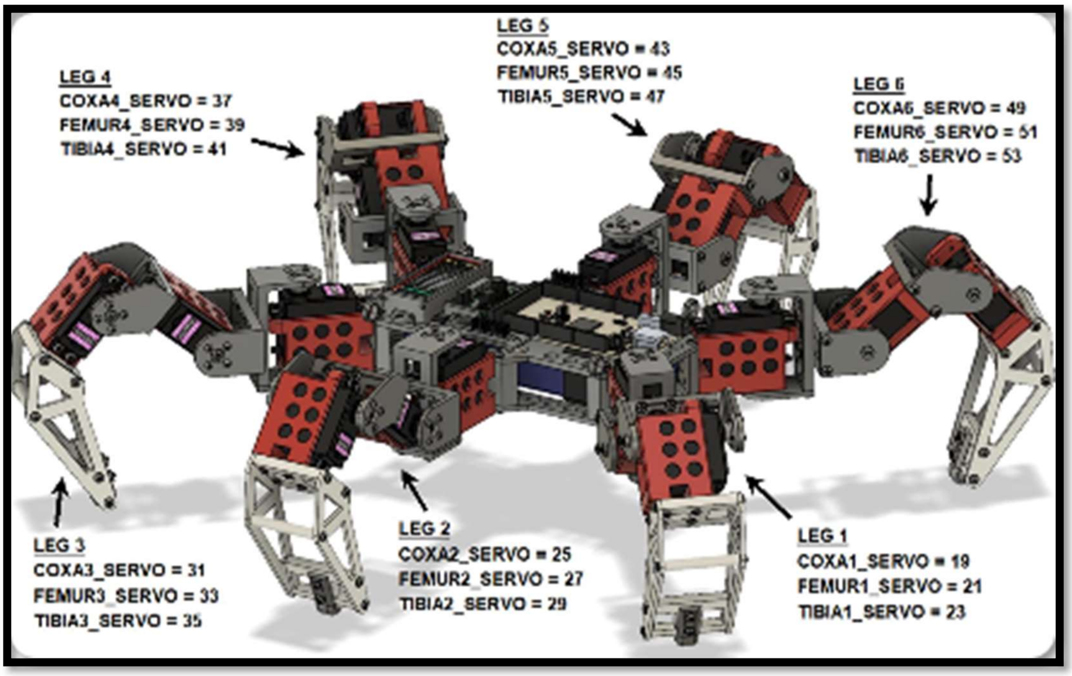
“Design Analysis of Hexapod Robot”

Team Pheonix,Department of Automation and Robotics, Amrutvahini College of Engineering,Sangamner,Ahamednagar.

Email:-Kawadeganesh81@gmail.com ,punamkhilari4@gmail.com ,pragati7219@gmail.com



Introduction

We present the design, simulation, and control of a hexapod robot using tools available in MATLAB software. In addition, we design and implement a dynamic model (using the Simscape Multibody™ toolbox) as well as a three-dimensional model of the robot, using Virtual Reality Modeling Language (VRML), that help to visualize the robot’s walking sequence. This three-dimensional model is interconnected with the Simscape Multibody™ blocks using MATLAB’s virtual reality blocks. Apart from this, and following specific requirements, we design and implement a Proportional– Integral–Derivative controller in order to obtain a pre-established displacement for the robot that, thanks to the developed computer simulations, proved to be satisfactory. Special emphasis is put in obtaining a modular representation of the dynamic model of the studied robot because it will permit to design more sophisticated nonlinear controllers in future works, allowing a good dynamic behavior of the robot in front of environmental perturbations, an issue that will become evident through computer simulations of its displacement

.

Equipments Used In Hexapod Robot

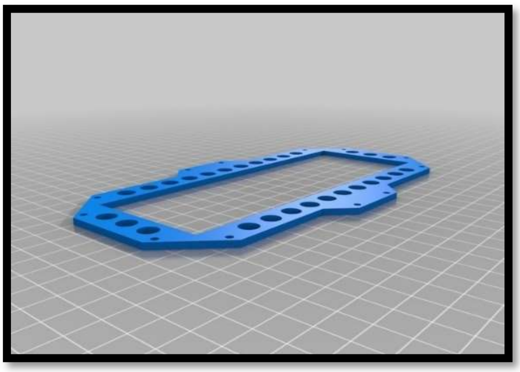
Sr No	Product Name	Quantity
1	Ardino Mega	1
2	3D printing parts	124
3	Ps2 Controller	1
4	Servo Motors	18
5	5500 Mah Battery	1
6	20A Buck Convertor	1
7	Pca9685	1
8	Battery Connector	1
9	Perfboard	2
10	Wires	1
11	624z Bearing	1
12	Screws	438
13	Elctronics	1
14	Strips	2
15	Toggle Switch	1

3D parts used in projects

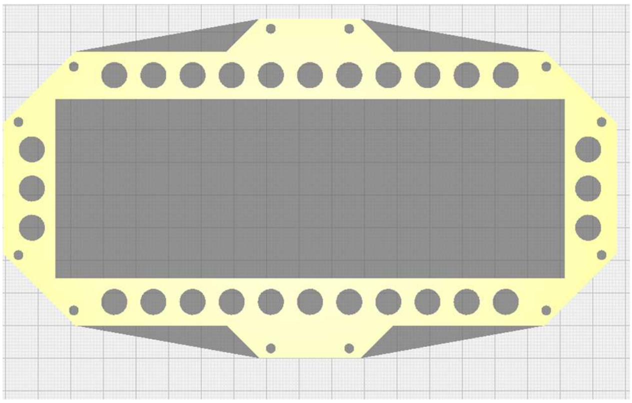
Sr No	Product Name	Quantity
1	Body Bottom Plate	1
2	Body Riser	6
3	Body Top Plate	1
4	Femur Bracket End Cap	12
5	Femur Bracket	12
6	Receiver Holder	1
7	SBEC Holder	1
8	Servo Bearing Center	18
9	Servo Mount	18
10	Tibia Base Plate	6
11	Tibia Bracket End Cap	6
12	Tibia Bracket	6

13	Tibia Foot Bumper	6
14	Tibia Foot Plate	6
15	Tibia Side 1	6
16	Tibia Side 2	6
17	Tibia Spacer Tube	6
18	Wire Guide	6

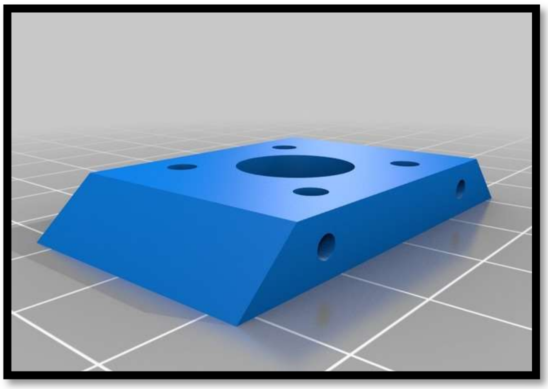
1) Body Bottom Plate:-



Part Dimensions:-188×104×3 mm
Allowable Tolarance-
Length:- 188±0.1 mm
Breadth:- 104±0.1 mm
Height :- 3±0.05 mm

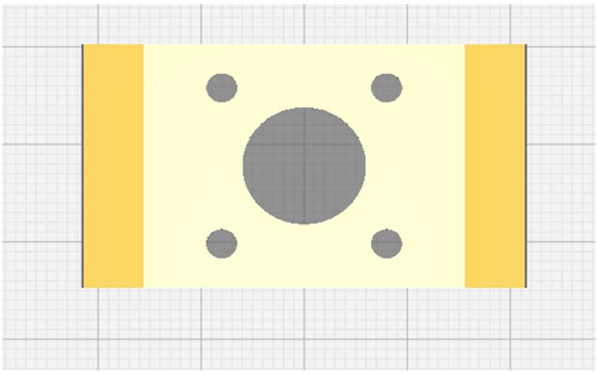


2) Body Riser:-

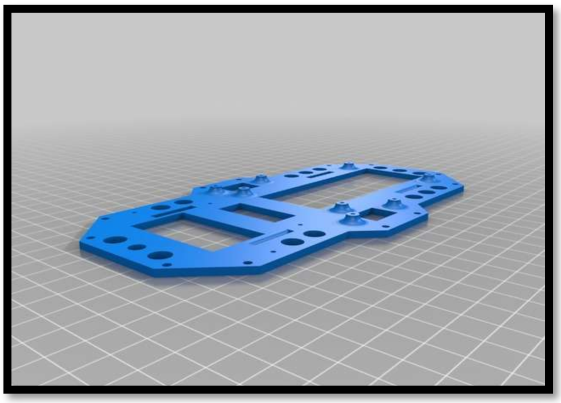


Part Dimensions:-43×25×6 mm
Allowable Tolarance-
Length:- 43±0.05 mm
Breadth:- 25±0.1 mm
Height :- 6±0.10 mm

.



3) Body top plate:-



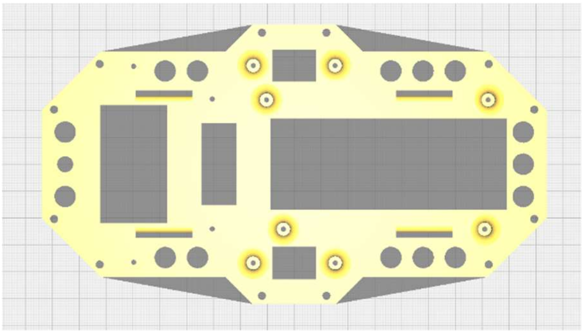
Part Dimensions:-188×104×6.5 mm

Allowable Tolarance-

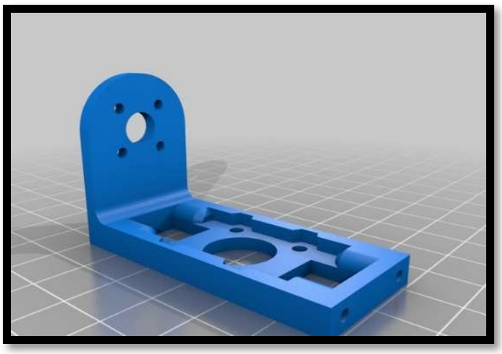
Length:- 188±0.1 mm

Breadth:- 104±0.1 mm

Height :- 6.5±0.05 mm



4) Femer Bracket:-



Part Dimensions:-25×58×38 mm

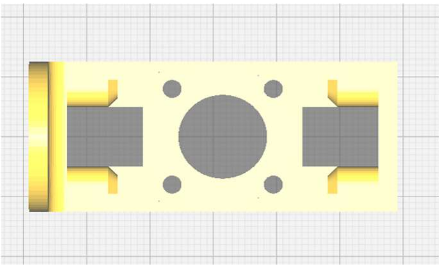
Allowable Tolarance-

Length:- 25±0.05 mm

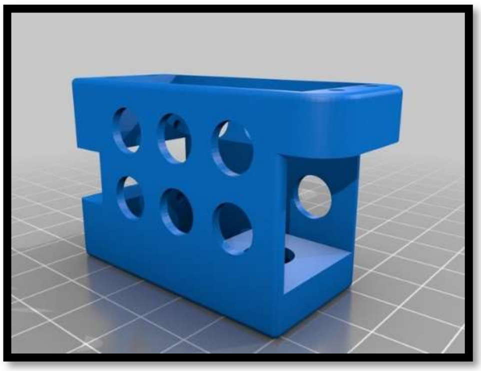
Breadth:- 58±0.1 mm

Height :- 38±0.1 mm

.



5) Servo Mount:-



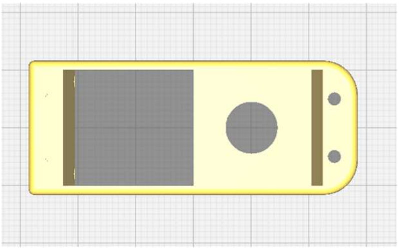
Part Dimensions:-23.3×57×35.7 mm

Allowable Tolarance-

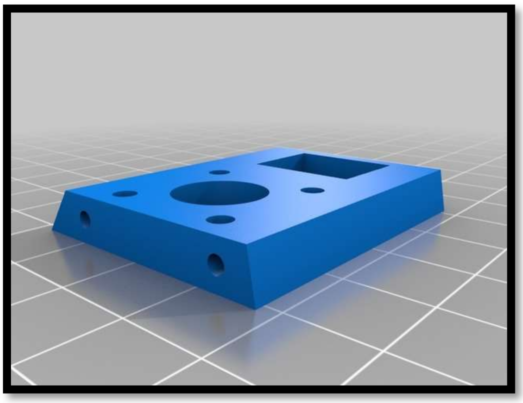
Length:- 23.3±0.05 mm

Breadth:- 57±0.05 mm

Height :- 35.7±0.05 mm



6) Tibia Base plate:-



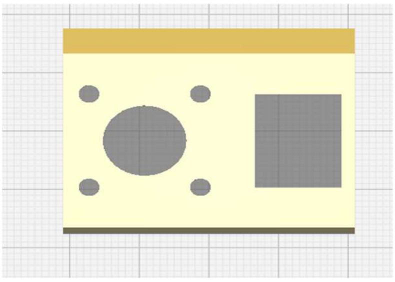
Part Dimensions:-42×35×6 mm

Allowable Tolarance-

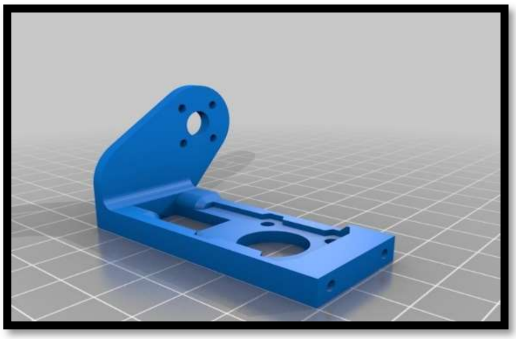
Length:- 42±0.025 mm

Breadth:- 35±0.1 mm

Height :- 6±0.05 mm

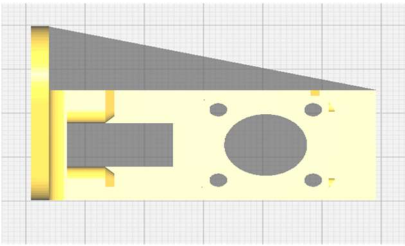


7) Tibia Brachet:-

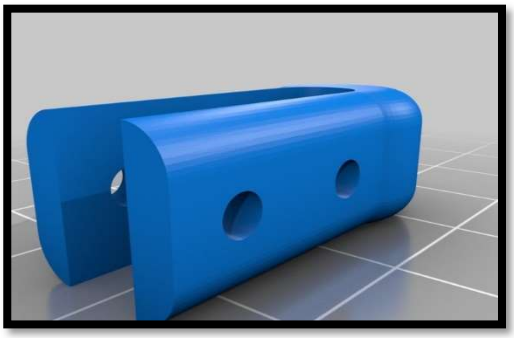


Part Dimensions:-39.5×58.2×35 mm
Allowable Tolarance-

Length:- 39.5±0.025 mm
Breadth:- 58.2±0.1 mm
Height :- 35±0.05 mm

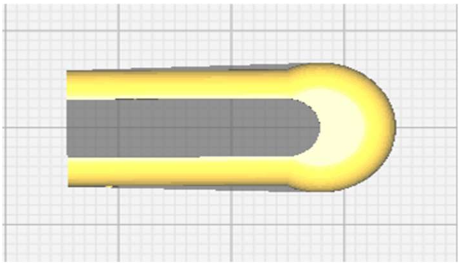


8) Tibia foot Bumper:-

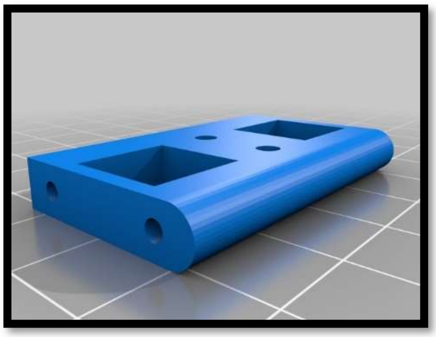


Part Dimensions:-29×13.4×12 mm
Allowable Tolarance-

Length:- 29±0.001 mm
Breadth:- 13.4±0.001 mm
Height :- 12±0.001 mm

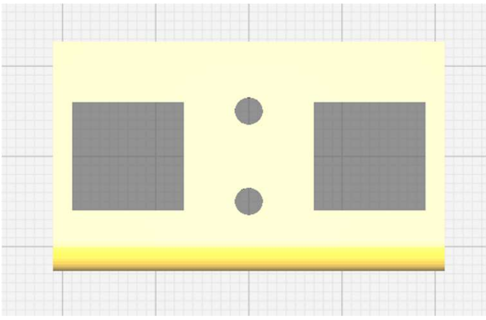


9) Tibia Foot plate:-

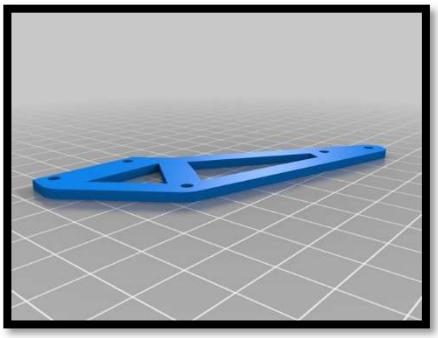


Part Dimensions:-42×25.7×6 mm
Allowable Tolarance-

Length:- 42±0.15 mm
Breadth:- 25.7±0.1 mm
Height :- 6±0.05 mm



10) Tibia Side:-



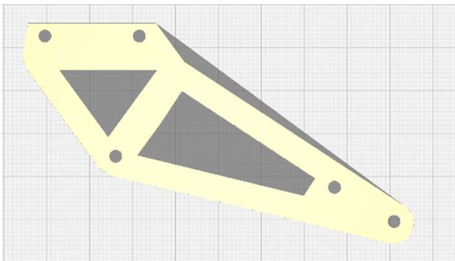
Part Dimensions:-91.1×51.3×2.4 mm

Allowable Tolarance-

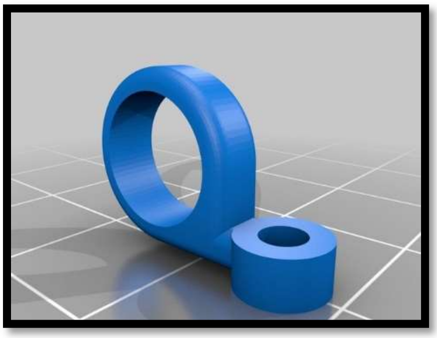
Length:- 91.1±0.1 mm

Bredth:- 51.3±0.01 mm

Height :- 2.4±0.001 mm



11) Wire Guide:-



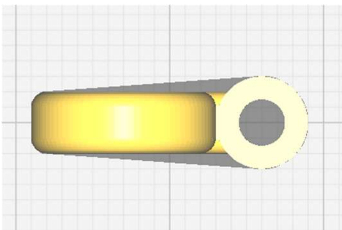
Part Dimensions:-6×18×12.6mm

Allowable Tolarance-

Length:- 6±0.05 mm

Bredth:- 18±0.05 mm

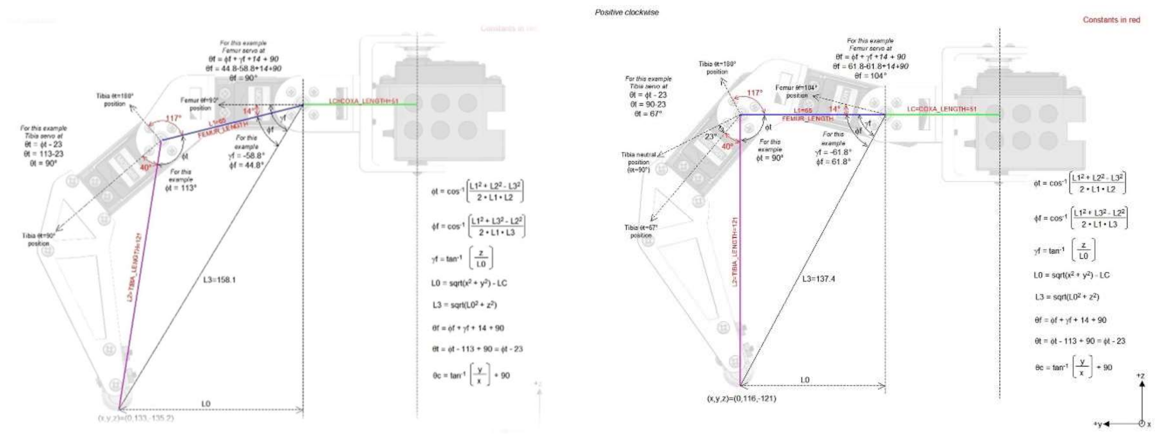
Height :- 12.6±0.05mm



3D Printed Parts:-

We designed the robot in Solid Works and PTC Creo6.0 Software and printed the parts on Creality CR 10S 3D printer. The parts were designed with low weight in mind (thin walls, lightening holes, etc.) to maximize the chances of this working with the servos I had chosen. The Black PLA parts were printed a bit on the hot side (210° C) for better layer-to-layer adhesion and part strength. All parts were printed with a 0.4mm nozzle at 0.2mm layer height.

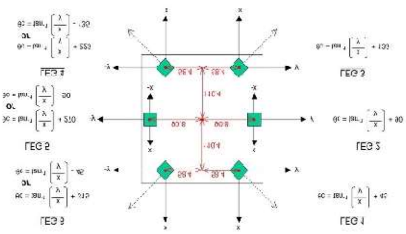
Mechanism:-



The robot works using inverse kinematic (IK) calculations:-

the servo positions are computed based on the desired x,y,z coordinates of the tip of each foot. Knowing a few constants – the tibia, femur, and coxa lengths and the offset angles between the axis of the servos and the structure permits the desired servo angles for all possible x,y,z coordinates to be calculated. Each leg is mounted at a different angle and the offsets of each leg from the center of the body is different, so the computed coxa angle varies for each leg.

When walking, the robot uses the remote joystick inputs – x and y from the



right joystick and rotation from the left joystick – to tell the body which way to move. The gait engines compute the individual foot tip positions required to accomplish the desired movement. Then the IK calculation is applied to each leg’s foot tip coordinate to tell the individual servos where to go to position the legs as requested by the gait engine.

There are also modes programmed to stand in-place and translate or rotate the body. The translation is just a simple addition of an x, y, or z offset to the coordinate system. Body rotation is more involved, but is derived as shown in below where the x, y, and z offsets are computed using matrix multiplication.

First rotate the X and Y axes:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & (\cos X) & (-\sin X) \\ 0 & (\sin X) & (\cos X) \end{bmatrix} \times \begin{bmatrix} (\cos Y) & 0 & (\sin Y) \\ 0 & 1 & 0 \\ (-\sin Y) & 0 & (\cos Y) \end{bmatrix} = \begin{bmatrix} (\cos Y) & 0 & (\sin Y) \\ (-\sin X)(-\sin Y) & (\cos X) & (-\sin X)(\cos Y) \\ (\cos X)(-\sin Y) & (\sin X) & (\cos X)(\cos Y) \end{bmatrix}$$

Now rotate around the Z axis:

$$\begin{bmatrix} (\cos Y) & 0 & (\sin Y) \\ (-\sin X)(-\sin Y) & (\cos X) & (-\sin X)(\cos Y) \\ (\cos X)(-\sin Y) & (\sin X) & (\cos X)(\cos Y) \end{bmatrix} \times \begin{bmatrix} (\cos Z) & (-\sin Z) & 0 \\ (\sin Z) & (\cos Z) & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} (\cos Y)(\cos Z) & (\cos Y)(-\sin Z) & (\sin Y) \\ (-\sin X)(-\sin Y)(\cos Z)+(\cos X)(\sin Z) & (-\sin X)(-\sin Y)(-\sin Z)+(\cos X)(\cos Z) & (-\sin X)(\cos Y) \\ (\cos X)(-\sin Y)(\cos Z)+(\sin X)(\sin Z) & (\cos X)(-\sin Y)(-\sin Z)+(\sin X)(\cos Z) & (\cos X)(\cos Y) \end{bmatrix}$$

Then rotate the vector input by the product of the above:

$$[\ (X)\ \ (Y)\ \ (Z)\ ] \times \begin{bmatrix} (\cos Y)(\cos Z) & (\cos Y)(-\sin Z) & (\sin Y) \\ (-\sin X)(-\sin Y)(\cos Z)+(\cos X)(\sin Z) & (-\sin X)(-\sin Y)(-\sin Z)+(\cos X)(\cos Z) & (-\sin X)(\cos Y) \\ (\cos X)(-\sin Y)(\cos Z)+(\sin X)(\sin Z) & (\cos X)(-\sin Y)(-\sin Z)+(\sin X)(\cos Z) & (\cos X)(\cos Y) \end{bmatrix} =$$

$$\begin{aligned} Xoutput &= (X)(\cos Y)(\cos Z) + (Y)(-\sin X)(-\sin Y)(\cos Z) + (Y)(\cos X)(\sin Z) + (Z)(\cos X)(-\sin Y)(\cos Z) + (Z)(\sin X)(\sin Z) \\ Youtput &= (X)(\cos Y)(-\sin Z) + (Y)(-\sin X)(-\sin Y)(-\sin Z) + (Y)(\cos X)(\cos Z) + (Z)(\cos X)(-\sin Y)(-\sin Z) + (Z)(\sin X)(\cos Z) \\ Zoutput &= (X)(\sin Y) + (Y)(-\sin X)(\cos Y) + (Z)(\cos X)(\cos Y) \end{aligned}$$

Finally, simplify the above to get X, Y, and Z equations for the output vector in terms of the inputs:

$$\begin{aligned} Xoutput &= (X)(\cos Y)(\cos Z) + (Y)(\sin X)(\sin Y)(\cos Z) + (Y)(\cos X)(\sin Z) - (Z)(\cos X)(\sin Y)(\cos Z) + (Z)(\sin X)(\sin Z) \\ Youtput &= -(X)(\cos Y)(\sin Z) - (Y)(\sin X)(\sin Y)(\sin Z) + (Y)(\cos X)(\cos Z) + (Z)(\cos X)(\sin Y)(\sin Z) + (Z)(\sin X)(\cos Z) \\ Zoutput &= (X)(\sin Y) - (Y)(\sin X)(\cos Y) + (Z)(\cos X)(\cos Y) \end{aligned}$$

References:
https://en.wikipedia.org/wiki/Rotation_matrix
https://en.wikipedia.org/wiki/Matrix_multiplication

Calculations can also be done in a different order as a double check:

First rotate the vector input around the X axis:

$$[\ (X)\ \ (Y)\ \ (Z)\ ] \times \begin{bmatrix} 1 & 0 & 0 \\ 0 & (\cos X) & (-\sin X) \\ 0 & (\sin X) & (\cos X) \end{bmatrix} = [\ (X)\quad (Y)(\cos X)+(Z)(\sin X)\quad (Y)(-\sin X)+(Z)(\cos X)\ ]$$

Now rotate that result around the Y axis:

$$[\ (X)\quad (Y)(\cos X)+(Z)(\sin X)\quad (Y)(-\sin X)+(Z)(\cos X)\ ] \times \begin{bmatrix} (\cos Y) & 0 & (\sin Y) \\ 0 & 1 & 0 \\ (-\sin Y) & 0 & (\cos Y) \end{bmatrix} =$$

$$[\ (X)(\cos Y)+(Y)(-\sin X)(-\sin Y)+(Z)(\cos X)(-\sin Y)\quad (Y)(\cos X)+(Z)(\sin X)\quad (X)(\sin Y)+(Y)(-\sin X)(\cos Y)+(Z)(\cos X)(\cos Y)\ ]$$

Then rotate the result around the Z axis:

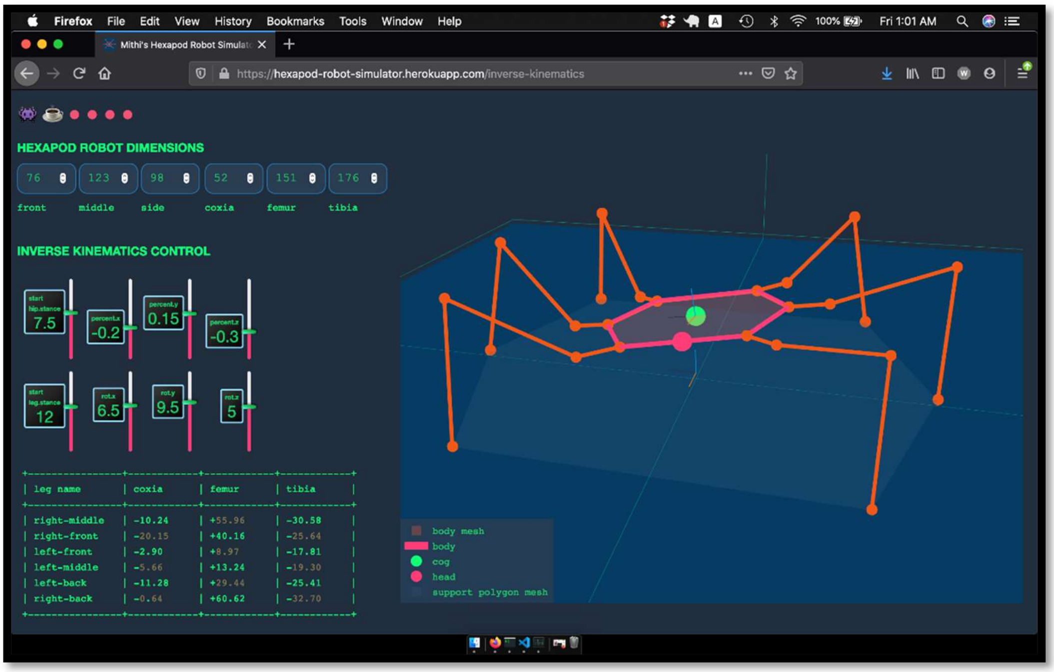
$$[\ (X)(\cos Y)+(Y)(-\sin X)(-\sin Y)+(Z)(\cos X)(-\sin Y)\quad (Y)(\cos X)+(Z)(\sin X)\quad (X)(\sin Y)+(Y)(-\sin X)(\cos Y)+(Z)(\cos X)(\cos Y)\ ] \times \begin{bmatrix} (\cos Z) & (-\sin Z) & 0 \\ (\sin Z) & (\cos Z) & 0 \\ 0 & 0 & 1 \end{bmatrix} =$$

$$\begin{aligned} Xoutput &= (X)(\cos Y)(\cos Z) + (Y)(-\sin X)(-\sin Y)(\cos Z) + (Y)(\cos X)(\sin Z) + (Z)(\cos X)(-\sin Y)(\cos Z) + (Z)(\sin X)(\sin Z) \\ Youtput &= (X)(\cos Y)(-\sin Z) + (Y)(-\sin X)(-\sin Y)(-\sin Z) + (Y)(\cos X)(\cos Z) + (Z)(\cos X)(-\sin Y)(-\sin Z) + (Z)(\sin X)(\cos Z) \\ Zoutput &= (X)(\sin Y) + (Y)(-\sin X)(\cos Y) + (Z)(\cos X)(\cos Y) \end{aligned}$$

Finally, simplify the above to get X, Y, and Z equations for the output vector in terms of the inputs:

$$\begin{aligned} Xoutput &= (X)(\cos Y)(\cos Z) + (Y)(\sin X)(\sin Y)(\cos Z) + (Y)(\cos X)(\sin Z) - (Z)(\cos X)(\sin Y)(\cos Z) + (Z)(\sin X)(\sin Z) \\ Youtput &= -(X)(\cos Y)(\sin Z) - (Y)(\sin X)(\sin Y)(\sin Z) + (Y)(\cos X)(\cos Z) + (Z)(\cos X)(\sin Y)(\sin Z) + (Z)(\sin X)(\cos Z) \\ Zoutput &= (X)(\sin Y) - (Y)(\sin X)(\cos Y) + (Z)(\cos X)(\cos Y) \end{aligned}$$

Simulation Analysis:-



In order to carry out a modeling and further simulation of a hexapod robot, it is necessary to take into account the robot's physical characteristics (mass, dimensions of thorax, measures of links, and the inertia matrix). In this case, we will employ the model. which details the robot's size, as well as the other parameters. However, some parameter modifications will be made, trying to get as close as possible to real conditions. Basically, we will employ the robot geometry.

To obtain the three-dimensional (3D) model of the hexapod robot, we will employ the VRML language, developing a robot model using simple geometrical figures. This language allows to obtain a complex model, simply using a group of basic 3D bodies (cubes, spheres, cones, etc.). There are many alternatives to develop a 3D model using VRML language: one of them consists in programming it directly using its commands or employing a graphical editor that eases the design. In this last case, a script is developed containing the instructions that model the robot.

Conclusion

The Dynamic Analysis and simulation of hexapod robot via MATLAB Simulation is addressed in this paper. Forward and inverse kinematics are considered to develop a complete dynamic model of the hexapod. For tripod walking gait, trajectory of each leg is generated for support and transfer phase. The dynamic model is presented to determined the joints’ torque for each leg. Furthermore, feet-ground interaction forces was also considered to obtain the feet force distributions. The main advantages of the MATLAB Simulation software is that model is its simplicity compared to the traditional modeling tools, as it is independent of robot differential equations. This advantage arises for complex Hexapod.