A

Semester Project-II

Report

On

"Quadrupedal Spider Robot"

By

1. Fegade Lokesh Manoj (SY-35)
2. Rajput Vaibhav Navalsinh(SY-41)
3. Mahajan Sanket Dipak(SY-42)
4. Borase Yatish Mangesh(SY-46)
5. Badgujar Sachin Shashikant(SY-52)

R. C. PATEL
INSTITUTE OF TECHNOLOGY
An Autonomous Institute

Department of Artificial Intelligence and Machine Learning

The Shirpur Education Society's

R. C. Patel Institute of Technology, Shirpur - 425405.

[2022-23]

A

Semester Project-II Report

On

"Quadrupedal Spider Robot"

In partial fulfillment of requirements for the degree of

Bachelor of Technology

In

Department of Artificial Intelligence and Machine Learning

Submitted By

1. Fegade Lokesh Manoj(SY-35)
2. Rajput Vaibhav Navalsinh(SY-41)
3. Mahajan Sanket Dipak(SY-42)
4. Borase Yatish Mangesh(SY-46)
5. Badgujar Sachin Shashikant(SY-52)

Under the Guidance of

Prof. Shailendra M. Pardeshi

R. C. PATEL
INSTITUTE OF TECHNOLOGY
An Autonomous Institute

The Shirpur Education Society's
R. C. Patel Institute of Technology, Shirpur - 425405.

Department of Artificial Intelligence and Machine Learning

[2022-23]

R. C. PATEL
INSTITUTE OF TECHNOLOGY
An Autonomous Institute

The Shirpur Education Society's

R. C. Patel Institute of Technology
Shirpur, Dist. Dhule (M.S.)

Department of Artificial Intelligence and Machine Learning

CERTIFICATE

This is to certify that the Semester Project-II entitled "**Quadruped Spider Robot**" has been carried out by team:

1. **Fegade Lokesh Manoj(SY-35)**
2. **Rajput Vaibhav Navalsinh(SY-41)**
3. **Mahajan Sanket Dipak(SY-42)**
4. **Borase Yatish Mangesh(SY-46)**
5. **Badgujar Sachin Shashikant(SY-52)**

under the guidance of **Prof. Shailendra M. Pardeshi** in partial fulfillment of the requirement for the degree of Bachelor of Technology in Department of Artificial Intelligence and Machine Learning of Dr. Babasaheb Ambedkar Technological University, Lonere during the academic year 2022-23(Semester-IV).

Date: 20/06/2023

Place: Shirpur

Guide	Semester Project-II Coordinator
Prof. Shailendra M. Pardeshi	Prof. Shailendra M. Pardeshi

H.O.D.	Director
Prof. Dr. R. B. Wagh	Prof. Dr. J. B. Patil

ACKNOWLEDGEMENT

It is a privilege for us to have been associated with Prof. Shailendra M. Pardeshi, our guide during semester project II have been greatly benefited by his valuable suggestion and ideas. It is with great pleasure that we express our deep sense of gratitude to his valuable guidance, constant encouragement and patience throughout this work.

We express our gratitude to Prof. Dr. R. B. Wagh [HOD of AI and Data Science] for his constant encouragement, co-operation and support and also thankful to all people who have contributed in their own way in making this semester project-II success.

We would also like to extend our gratitude towards Director Prof. Dr. J. B. Patil for providing us with all the facility that was required.

Project Team:

1. **Fegade Lokesh Manoj(SY-35)**
2. **Rajput Vaibhav Navalsinh(SY-41)**
3. **Mahajan Sanket Dipak(SY-42)**
4. **Borase Yatish Mangesh(SY-46)**
5. **Badgujar Sachin Shashikant(SY-52)**

PAGE INDEX

FIGURE INDEX

ABSTRACT

This project aims to create a versatile quadruped spider robot, which belongs to the category of multi-legged robots and has four legs for movement. Quadruped robots offer various options for leg placement, leg designs, and gait patterns. The goal is to design a robot with a spider-like appearance that can navigate through different environments, such as forests and large farms. To achieve this, the robot is equipped with a real-time web control system that uses wireless technology to detect the current environmental

Keywords : *Quadrupedal, Spider, Robot, ARDUINO, Wireless Sensor Network.*

CHAPTER – 1 INTRODUCTION

1. INTRODUCTION

Robots are increasingly used nowadays, and they're also being used by humans in a wide variety of applications. It was meant to have a robot that people could use to achieve their goals, as well as be easily controlled by them. This project talks about the design and fabrication of a multipurpose quadruped spider robot. The quadruped robot, defined as a robot with four legs for walking or moving, is included in the category of multilegged robots. In quadruped robots, there are so many possibilities for placement of the legs, leg designs, and gait patterns. This project aims to build a robot with four legs, which looks as if it's a biological spider. It can be handled by a remote control system by using Moblie(App we are using Bluetooth RC controller). This is done by using a wireless device Bluetooth, In this Project we using ARDUINO IDE SOFTWARE to upload the program and some Hardware parts also like Arduino Nano, Servo Motor, HC05 Bluetooth, and Nano 328P Expansion Adapter Breakout Board IO Shield. A type of robot imitating the movement of spiders is called a Four Leg Spider Robot. It's got four legs that let it move in a wide range of directions, including forward, backward, up, and down. The leg muscles of a spider robot tend to be driven by servomotors, which allow them to control with high accuracy and precision. The ability of a spider robot to maneuver difficult terrains, e.g. flat or rocky surfaces, is one of its key advantages. This allows it to be used in outdoor situations or search and rescue operations, which can have difficulty for the movement of a conventional riding or walking robot when they are hampered by obstacles.

1.1. Implementation

Steps on how to create a Quadrupedal Spider Robot

- Define Specifications: Determine the size, weight, locomotion capabilities, and other requirements for your quadrupedal spider robot. This will guide the subsequent design process.

I. Mechanical Design:

Body Structure: Design a lightweight, yet sturdy body structure for the robot. Consider using materials like aluminum or carbon fiber to ensure strength and durability.

Legs: Design and fabricate four spider-like legs with multiple joints for each leg. Use servo motors or other actuators to control the leg movements. Each leg should have sufficient range of motion to enable walking, crawling, and turning.

Sensors: Integrate sensors like proximity sensors, accelerometers, or gyroscopes to provide feedback for balance and orientation control.

Power System: Determine the power requirements for the robot and design a suitable power system. This could include batteries, power distribution, and charging circuits.

II. Electronics:

Microcontroller/Processor: Select a microcontroller or processor board capable of controlling the robot's movements and handling sensor data. Arduino or Raspberry Pi are commonly used platforms for robotics projects.

Motor Control: Interface the microcontroller with motor drivers to control the servo motors or actuators that move the robot's legs.

Sensor Integration: Connect and calibrate the sensors to the micro controller, ensuring proper data acquisition and processing.

III. Control System:

Kinematics: Develop a kinematic model to determine the leg movements required for different locomotion patterns (walking, crawling, turning). This model will serve as the basis for generating control signals.

Gait Generation: Implement gait generation algorithms that define the leg movements and coordination patterns for achieving stable and efficient locomotion.

Feedback Control: Use sensor data to implement feedback control algorithms that adjust leg movements and body posture to maintain stability and adapt to different terrains.

IV. Software Development:

Programming: Write the necessary software code to control the robot's movements, interpret sensor data, and implement the desired behaviors.

User Interface: Develop a user interface (e.g., a mobile app or a computer program) to remotely control the robot or provide high-level commands.

Testing and Debugging: Test the robot's functionality, make adjustments, and debug any issues that arise during the development process.

V. Integration and Assembly:

Assemble the mechanical components, including the body structure, legs, motors, and sensors.

Connect the electronic components, ensuring proper wiring and secure connections.

Integrate the control software with the microcontroller and perform initial tests.

VI. Iteration and Improvement:

Evaluate the robot's performance and make necessary adjustments to improve its locomotion, stability, or other desired features.

Working and its Functions

Architecture

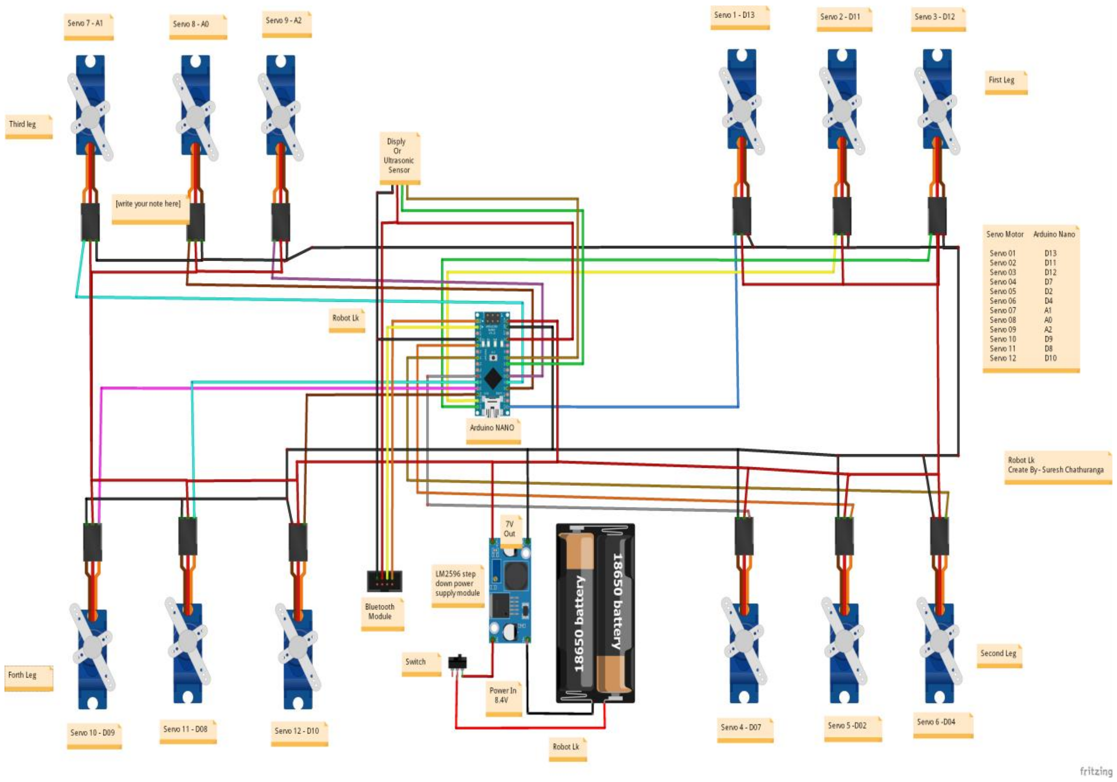

Figure (1): Servo Connection

Identify the servo motors: Determine the number of servo motors you have and their corresponding functions (e.g., hip joint or knee joint).

Power supply: Connect the positive terminal (+) of the power supply to the servo power input, and the negative terminal (-) to the ground (GND) of the servo motors. Make sure the power supply voltage matches the servo motors' voltage requirements (usually +5V or +6V).

Signal connections: Each servo motor has three wires: red, black/brown, and white/yellow.Connect the red wire (power) of each servo to the positive terminal of the power supply.

- Connect the black/brown wire (ground) of each servo to the ground (GND) of the power supply.

- Connect the white/yellow wire (signal) of each servo to the respective signal pin on the servo controller.

CHAPTER-2 RELATED CONCEPTS

Kinematics:

Kinematics deals with the study of motion without considering forces and torques. In the context of quadrupedal spider robots, kinematics involves analyzing the movement and positions of the robot's legs and body. It helps determine the leg joint angles and coordination required to achieve desired locomotion patterns.

Control Systems: Control systems are crucial for maintaining stability, coordinating leg movements, and achieving desired behaviors in quadrupedal spider robots. Control algorithms use sensor feedback and reference inputs to adjust the robot's posture, leg positions, and overall locomotion. Control systems can be implemented using techniques such as proportional-integral-derivative (PID) control, model predictive control (MPC), or reinforcement learning.

Sensor Integration: Quadrupedal spider robots often incorporate various sensors to perceive their environment and gather feedback for control. Proximity sensors, such as ultrasonic or infrared sensors, can detect obstacles and aid in collision avoidance. Inertial measurement units (IMUs) provide information about the robot's orientation and acceleration. Vision systems, such as cameras or depth sensors, enable visual perception and object recognition.

One related concept to a spider robot is bio-inspired robotics. Bio-inspired robotics is a field of study that takes inspiration from biological systems, such as animals, insects, or plants, to design and develop robotic systems with enhanced capabilities.In the case of a spider robot, it is often inspired by the locomotion and movement patterns of real spiders. By mimicking the biomechanics and behaviors of spiders, engineers aim to create robots that can navigate complex terrains, exhibit agile movements, and demonstrate efficient locomotion.

Key concepts related to spider robots and bio-inspired robotics include:

a. Locomotion: Spider robots are designed to replicate the walking or crawling movements of spiders. Researchers study the leg arrangements, joint movements, and coordination mechanisms in spiders to develop algorithms and control strategies for the robot's leg movements.

b. Leg Kinematics: Understanding the kinematics of spider legs helps in designing the leg structures and joint configurations of spider robots. By replicating the leg morphology and degrees of freedom found in spiders, engineers aim to achieve similar dexterity and maneuverability in their robotic counterparts.

c. Gait Generation: Gait refers to the pattern of leg movements used for locomotion. Spider robots employ different gaits, such as tripod or alternating tripod, to traverse diverse terrains. Researchers analyze spider locomotion to determine the most efficient gaits and develop algorithms to generate appropriate leg movements for stable and agile locomotion.

d. Sensing and Perception: Spiders rely on sensory information from their environment to navigate and detect obstacles. Spider robots may incorporate sensors such as distance sensors, cameras, or tactile sensors to perceive their surroundings and adapt their movements accordingly.

e. Biomimicry: Biomimicry is the practice of imitating nature's designs and principles to solve engineering problems. In spider robot development, biomimicry involves replicating the morphology, locomotion, and behavior of spiders to achieve specific functionalities, such as traversing rough terrain or climbing vertical surfaces.

f. Control Systems: Spider robots employ sophisticated control systems to coordinate the movements of multiple legs and maintain stability.

These control systems utilize feedback from sensors and implement algorithms that mimic the coordination and adaptation mechanisms observed in spiders.

g. Applications: Spider robots find applications in various domains, including search and rescue operations, exploration of complex environments, inspection of confined spaces, and entertainment. Their ability to traverse challenging terrains and access areas inaccessible to wheeled or tracked robots makes them suitable for tasks where agility and maneuverability are critical.

By exploring these concepts and principles, researchers aim to create spider robots that exhibit efficient locomotion, adaptability to complex environments, and enhanced mobility, allowing them to perform tasks that traditional robot designs may struggle with.

CHAPTER-3 REQUIREMENTS

3.1 SOFTWARE REQUIREMENTS

1) Bluetooth RC Controller:

Bluetooth RC Controller app is an Android application that allows users to control robots or other devices that use Bluetooth communication. The app provides a user-friendly interface with a customizable joystick and button controls. It also supports multiple Bluetooth protocols, such as HC-05 and HC-06, and allows users to connect to and control multiple devices simultaneously. Bluetooth RC Controller app can be used for a variety of applications, including robotics, home automation, and remote control vehicle.

2) Arduino IDE:

The Arduino Software (IDE) is an open-source platform that simplifies the process of writing and uploading code to the board. It is compatible with Windows, Mac OS X, and Linux operating systems, and is written in Java programming language. The IDE is designed to provide a user-friendly interface for programming and is widely used for programming Arduino and other development boards. BLUETOOTH RC CONTROLLER APP: Bluetooth RC Controller app is an Android application that allows users to control.

3.2 HARDWARE REQUIREMENTS

1) Mobile:

Mobile it use for controlling our spider robot,by using Bluetooth RC Controller app.

3) Arduino Nano:

The servo motors, which control the movements of the robot legs, receive their final output from the Arduino Nano, while the input from the Bluetooth RC controller is received by the HC-05 module and forwarded to the Arduino Nano for appropriate output to the servo motors.

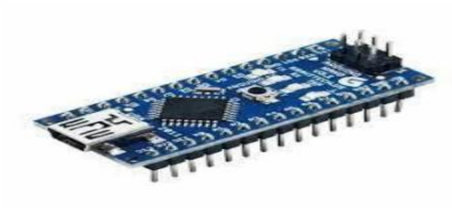

Figure (2): Arduino nano

4) Bluetooth Module HC – 05:

The HC-05 is a Bluetooth module commonly used for wireless communication in various electronic projects. It can be easily integrated into microcontroller projects and enables communication with other Bluetooth-enabled devices such as smartphones or tablets. The HC- 05 can operate in both master and slave modes and has a range of up to 10.

Figure (3): Bluetooth Module HC – 05

5) Nano 328P Expansion Adapter Breakout Board IO Shield:

The Nano 328P Expansion Adapter Breakout Board IO Shield can be used as an interface between the Arduino Nano board and other external components. It provides access to all the input and output pins of the Arduino Nano and allows for easy connection with other components such as sensors, motors, and displays. The breakout board also provides additional features such as voltage regulation, reverse polarity protection, and level shifting. It can be used in a variety of projects that require interfacing with external components and expanding the capabilities of the Arduino Nano board.

Figure (4): Nano 328P Expansion Adapter Breakout Board IO Shield

6) Servo Motor:

A servo motor is a small, efficient motor commonly used in various robotic and electronic projects for its ability to position its output shaft precisely. It consists of a DC motor, a potentiometer, and control circuitry enclosed in a compact housing. The motor's angular position is controlled by applying pulses of a certain width to its position signal terminal, which is typically done through a microcontroller or similar device. The pulse width ranges from approximately 0.5ms for 0-degree rotation to 2.2ms for 180-degree rotation, with a recommended frequency of around 50Hz to 60Hz. Servo motors are widely used due to their accuracy, compact size, and energy efficiency.

Figure (5): Servo Motor

6)DC-DC Buck Converter :

The buck converter produces an output voltage that is lower than its input voltage and is named so because the inductor generally opposes the input voltage, and in an ideal case, the output voltage is the product of the switching duty cycle and the supply voltage.

Figure (6): DC-DC Buck Converter

CHAPTER-4 IMPLEMENTATION DETAILS

4.1 Legs Description:-

Leg code is used for spider robots because it provides a systematic way to control the movements of their legs. Spider robots, also known as hexapods or multi-legged robots, typically have multiple legs that need to be coordinated to achieve stable locomotion and perform various tasks.

Leg code allows the robot to control the position, orientation, and movement of each leg individually or in coordination with other legs. It defines the gait patterns, which are sequences of leg movements, that enable the robot to walk, crawl, or perform other desired motions. The leg code specifies the joint angles, timing, and coordination between different legs to generate the desired leg trajectories and achieve stable locomotion.

By using leg code, the spider robot can adapt its leg movements to different terrains, obstacles, or tasks. The code can be designed to incorporate feedback from sensors, such as accelerometers or force sensors, to adjust leg movements based on the robot's interaction with the environment. Leg code provides a higher level of abstraction, allowing developers to focus on defining the desired leg behaviors rather than directly controlling individual motors or actuators.

Overall, leg code simplifies the programming and control of spider robots, enabling them to perform complex locomotion patterns and tasks with their multiple legs in a coordinated and efficient manner.

4.2 Remote Control Crawling Flowchart :

```
+ - - - - - - - - - - - - - - - - - - - - +
|                 S t a r t               |
+ - - - - - - - - - - - - - - - - - - - - +
                     |
                     |
```

```
                 v
+---------------------------+
| Initialize Bluetooth      |
| connection                |
+---------------------------+
                 |
                 |
                 v
+---------------------------+
| While Bluetooth           |
| connection is active      |
+---------------------------+
                 |
                 |
                 v
+---------------------------+
| Receive command from      |
| remote control            |
+---------------------------+
                 |
                 |
                 v
+---------------------------+
| If command is "Forward"   |
| Move all legs             |
| forward                   |
+---------------------------+
                 |
                 |
```

```
            v
+------------------------+
|  Else If command is    |
|    "Backward"          |
|    Move all legs       |
|      backward          |
+------------------------+
            |
            |
            v
+------------------------+
|  Else If command is    |
|    "Turn Left"         |
|  Turn left by adjusting|
|    leg movement        |
+------------------------+
            |
            |
            v
+------------------------+
|  Else If command is    |
|    "Turn Right"        |
| Turn right by adjusting|
|    leg movement        |
+------------------------+
            |
            |
            V
```

```
+--------------------------+
|   Else If command is     |
|       "Stop"             |
|    Stop all leg          |
|      movements           |
+--------------------------+
             |
             |
             v
+--------------------------+
|  Else If command is      |
|  "Custom Command 1"      |
| Perform custom action 1  |
+--------------------------+
             |
             |
             v
+--------------------------+
|   Else If command is     |
|  "Custom Command 2"      |
| Perform custom action 2  |
+--------------------------+
             |
             |
             V
```

```
+ - - - - - - - - - - - - - - - - - - - - - - - - +
|                  E l s e                        |
|    I n v a l i d   c o m m a n d ,              |
|    i g n o r e   o r   d i s p l a y            |
|     e r r o r   m e s s a g e                   |
+ - - - - - - - - - - - - - - - - - - - - - - - - +
                        |
                        |
                        V
+ - - - - - - - - - - - - - - - - - - - - +
|                 E n d                   |
+ - - - - - - - - - - - - - - - - - - - - +
```

A flowchart is a graphical representation of a process or algorithm that uses different symbols to illustrate the sequence of steps and decision points. Here's a description of a flowchart for a spider robot algorithm:

a) Start: The flowchart begins with the "Start" symbol, indicating the beginning of the algorithm.

b) Initialization: This step involves initializing the robot's hardware components and setting the initial positions and angles of the robot's legs. It is represented by a rectangular box.

c) Sensor Data Acquisition: The algorithm reads sensor data from various sensors on the spider robot, such as distance sensors, gyroscopes, or accelerometers. This information is used to gather data about the robot's environment. The symbol for sensor data acquisition is a parallelogram.

d) Task Planning: Based on the acquired sensor data, the algorithm analyzes the robot's current state and determines the desired task or behavior. It decides on the appropriate actions to achieve the task, considering the available sensor information. This step is represented by a diamond-shaped decision symbol.

e) Leg Movement Calculation: In this step, the algorithm calculates the required movement and angles for each leg to execute the desired actions. It considers factors such as stability, balance, and obstacle avoidance while calculating leg movements. The symbol for leg movement calculation is a rectangular box.

f) Servo Control: The algorithm sends control signals to the servo motors based on the calculated leg movements. It adjusts the servo positions and angles to move the robot's legs accordingly. The symbol for servo control is a parallelogram.

g) Loop Execution: This step represents the continuous repetition of steps 3 to 6 to enable real-time sensor data acquisition, task planning, leg movement calculation, and servo control. The algorithm continuously updates the robot's actions and movements based on the changing environment or task requirements. The loop execution is depicted by an arrow pointing back to the sensor data acquisition symbol.

h) Termination: The algorithm includes a termination condition to stop the flowchart execution when the desired task or behavior is achieved or when a specific condition is met. The termination condition is represented by a diamond-shaped decision symbol.

i) End: The flowchart ends with the "End" symbol, indicating the completion of the algorithm.

This flowchart provides a visual representation of the spider robot algorithm, illustrating the sequence of steps and decision points involved in controlling the robot's movements based on sensor data. The specific details within each step can vary depending on the robot's capabilities and the desired tasks or behaviors.

4.3 Result :-

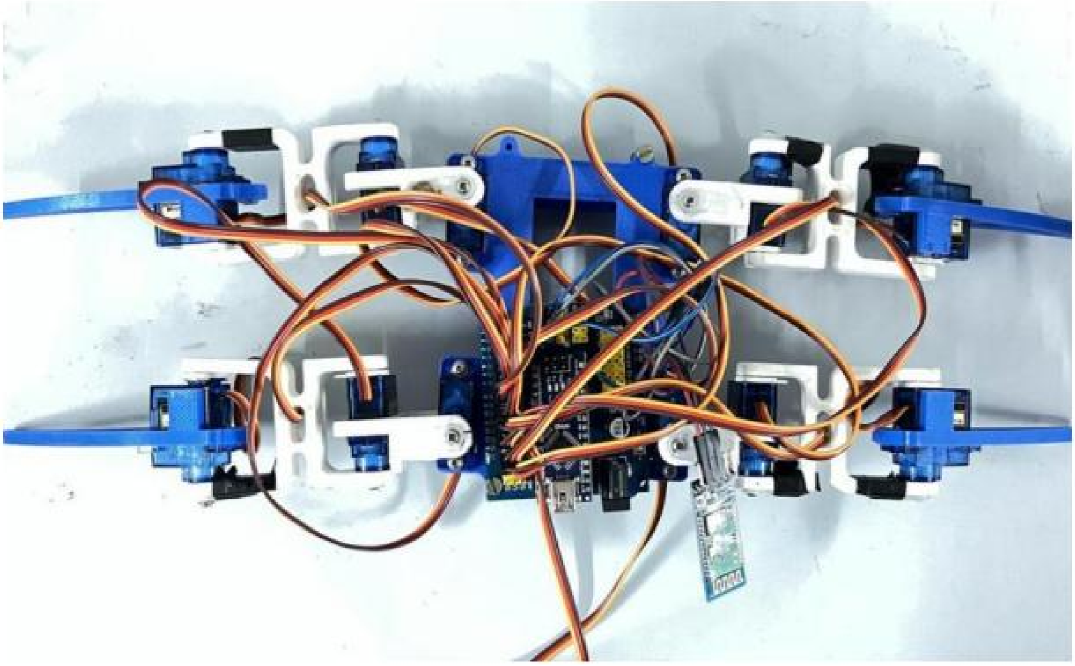

Figure (7):Quadrupedal Spider Robot

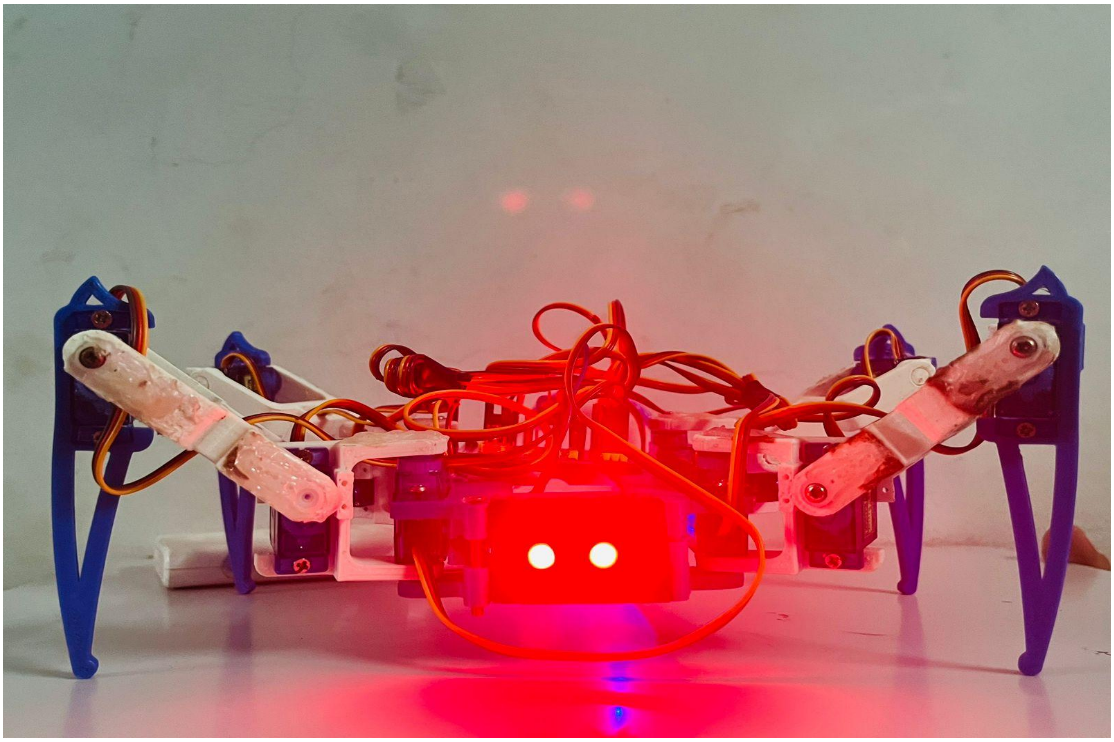

Figure (8): Front View of Spider Robot

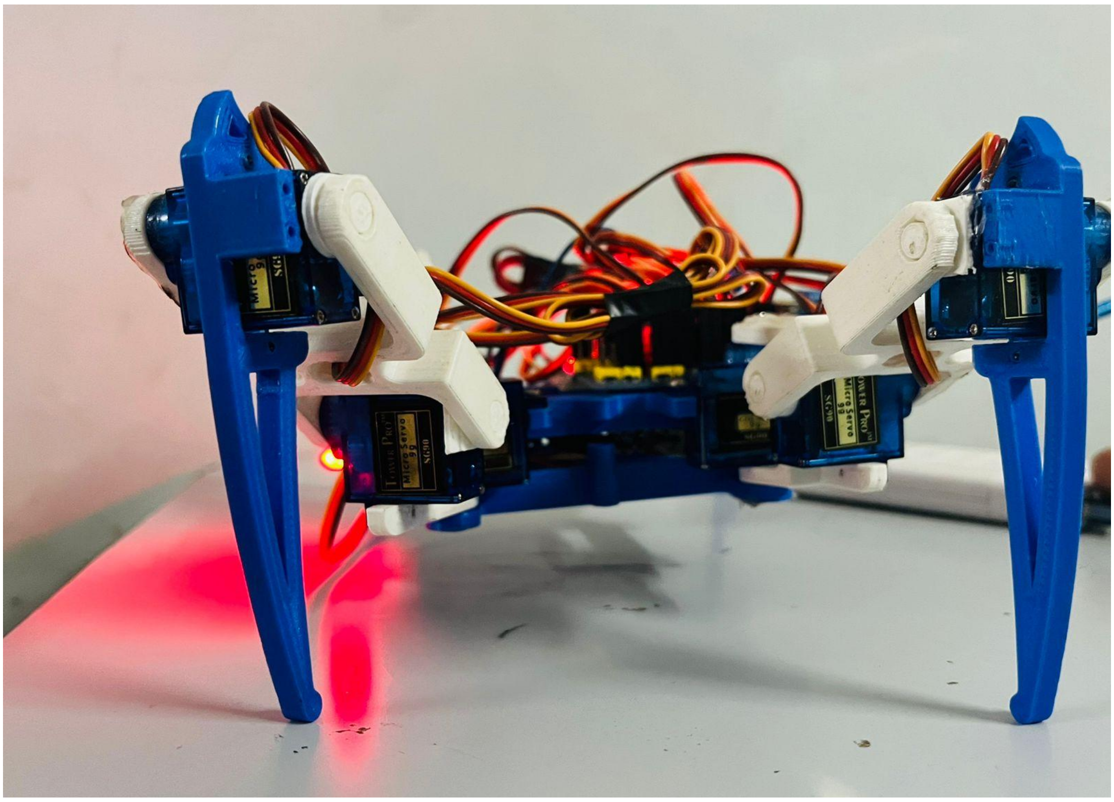

Figure (9): Side View of Spider Robot

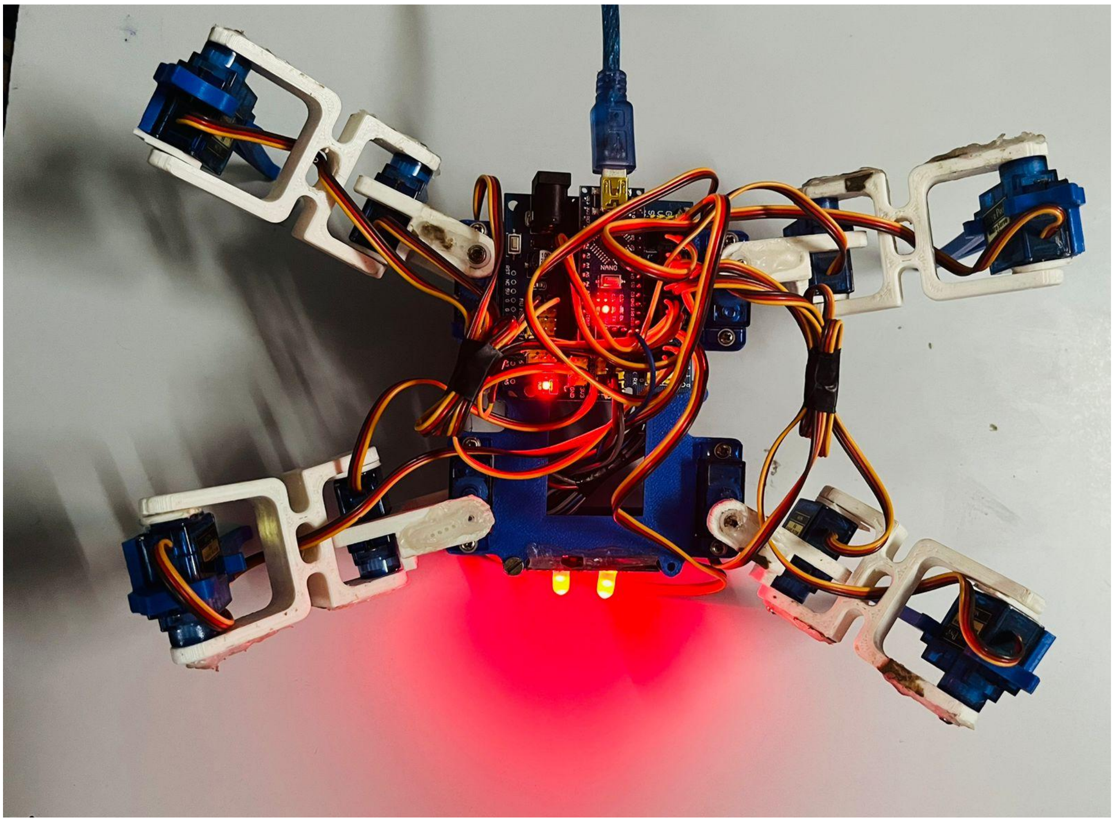

Figure (10): Top View of Spider Robot

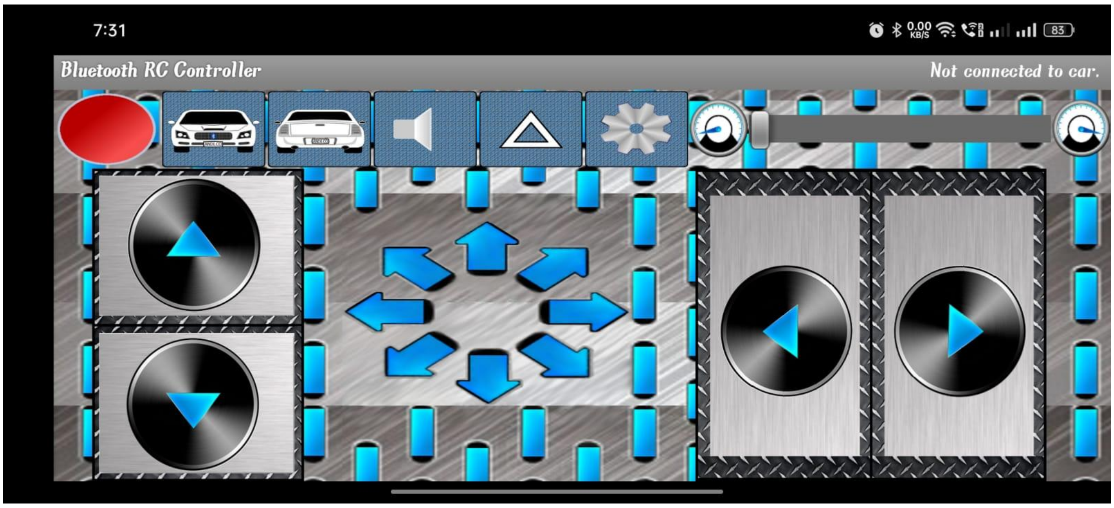

Figure (11): Mobile Remote Control

Conclusion

In this study, we presented the design and development of a quadrupedal spider robot using Arduino Nano and HC-05 modules. The robot is capable of moving in various positions and can be controlled using an Android smartphone. The buck converter provides a stable power supply to the system, and the robot moves smoothly without any issues. The quadrupedal spider robot has great potential in space exploration and rough terrain operations. Future studies can focus on developing more advanced spider robots capable of performing complex tasks.

BIBLIORGAPHY

[1] U. Saranli, M. Buehler, and D. E. Koditschek, "Rhex-A Simple and Highly Mobile Hexapod Robot," Int. J. Robot. Res., vol. 20, pp. 616-631, 2001.

[2] Kavlak, Koray. (2021). "Bluetooth Controlled Hexapod Robot Design," European Journal of Science and Technology. 10.31590/ejosat.960612

[3] Ritesh G.Waghe, Deepak BhoyIndl, Sagar Ghormade3, "A Real Time Design and Implementation of Walking Quadruped Robot for Environmental Monitoring," International Research Journal of Engineering And Technology (IRJET) vol:05,No:05 May 2018.

[4] Dipak Patil, Himali Patil, Abhijeet Patil, Sunil Kalal, "Camouflage Technique Based Multifunctional Army Robot," International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE) vol:4, February 2015

[5] Mr.V.Arun1 , Mr.S.V.S.Prasad2 , G.Sridhar Reddy3 , L.Ruthwik Reddy4 , M.Venkatesh5 , M.Sai Pavan Kumar6 , "Arduino Quadruped Robot," IOSR Journal of Electronics and Communication Engineering (IOSR-JECE) e-ISSN: 2278-2834,p- ISSN: 2278-8735.Volume 11, Issue 3, Ver. III (May-Jun .2016).

[6] *Santiago Noriega Alvarez, Maria Camila Rojas, Hemando Leon-Rodriguez, "Design and Development of Quadruped Spider Robot," Universidad EI Bosque, Bogota, Colombia*

[7] https://www.instructables.com/ARDUINO-SPIDER-ROBOT-QUADRUPED

[8] https://youtu.be/fnMmnd9k6q8

[9] https://github.com/Arijit1080/Spidy-The-Spider-Robot

[10] https://ijcrt.org/papers/IJCRTO020003.pdf