**Project Spiderbot**

**Introduction**

The SpiderBot project aims to create an advanced robotic platform capable of agile exploration and surveillance in dynamic and challenging environments. Drawing inspiration from the remarkable abilities of spiders, SpiderBot will incorporate cutting-edge locomotion mechanisms, sensory perception, and autonomous decision-making capabilities. This project description outlines the primary goals, features, and potential applications of SpiderBot. Through rigorous development, testing, and validation, SpiderBot will provide a reliable and adaptable solution for a wide range of real-world applications, contributing to advancements in robotics and enhancing human capabilities.

**Why a Hexapod?**

 Six legs yield extra stability and maneuverability, especially when climbing something like rubble. With 4 or less legs, the robot needs to be very careful about how it positions its center of gravity and which legs are off the ground, so as not to fall over. With 6 or more legs, the robot can lift multiple legs up without tipping over, and does not need to exert much effort try not to fall. It also means, from a robotics perspective, the algorithms for locomotion do not necessarily need to be closed loop, and do not necessarily need to take into consideration the momentum or center of gravity of the robot. Therefore, controlling such a robot, even though it has more limbs, is actually easy. More than six consumes a lot of power and moreover the gait patterns are complex .The more the legs the more complex it becomes

**Uses**

Following are few places where spiderbot can used:

* Search and Rescue: Spiderbots can navigate through complex and challenging environments, such as collapsed buildings or disaster zones, to search for survivors or assess the situation. Their ability to climb over debris and uneven surfaces makes them valuable for search and rescue missions.
* Inspection and Maintenance: Spiderbots can access difficult-to-reach areas, such as tall structures, bridges, or pipelines, to perform inspection and maintenance tasks. Their climbing abilities make them ideal for assessing the integrity of infrastructure.
* Surveillance and Security: Spiderbots can be deployed for surveillance and security purposes. They can navigate indoor and outdoor spaces, providing real-time video or sensor data to monitor and secure locations.
* Medical Applications: Spiderbots may be utilized in medical applications, such as minimally invasive surgery. Their compact size and ability to move around obstacles can be beneficial in accessing hard-to-reach areas within the human body.
* Mapping and Environmental Monitoring: Spiderbots can be equipped with various sensors to map terrain, monitor environmental parameters, or collect data in remote locations.
* Military and Defense: In military applications, spiderbots can be used for reconnaissance and surveillance missions. Their ability to move quietly and traverse challenging landscapes can be advantageous in certain scenarios.
* Space Exploration: Spider-like robots have been proposed for use in space missions, both for planetary exploration and satellite maintenance.

**Mechanical design**

All the legs are 3D printed and the base is made up of 5.5mm of plywood. The bot has a covering of Acrylic.

**Dimensions**

Base-Hexagon of side 140 MM

Leg-6 legs. Each leg having 3 joints. The three joints provide 3 DOF mimicking a human Leg of which dimensions are

HIP 124 mm

LEG 174 MM

Hip to Base 66 mm

**Electronics**

**List of Components used**

Single pole MCB, 10A buck converter, 10amg wires, 35kg-cm servo (RKI 1202), 16kg-cm servo (RKI 1206), Lippo battery 4200mah, Raspberry pi 4,2 Arduino Mega

**PCB design**

Trace widths used: 188.89mils, 13.79mils, 6mils Components: Arduino, Male-Male Headers, XT-60 connector Current rating: 20A Copper weight: 2oz

**Software**

We have used ROS (Robot Operating system) framework. The programming language used is Python

Rosserial is used for communication between the raspi and arduino

**Rosserial**

Rosserial is a protocol to send data through a serial interface. In a client-server rosserial implementation, a rosserial-server is a computer running ROS and a rosserial-client is the microprocessor that receives sensors’ data and transports it to the server in the form of ROS messages. rosserial-server in this implementation is a publishing node while rosserial-client is a subscriber node, although this can sometimes be the other way round.

**Requirements**

* *ROS Environment*: First, you need to have a working ROS environment on your computer. This involves installing ROS and setting up a ROS workspace with your packages.
* *Install Rosserial*: Install the Rosserial package on your ROS computer by running the appropriate command

sudo apt-get install ros-<distro>-rosserial-arduino

sudo apt-get install ros-<distro>-rosserial-python

* Install Rosserial on Arduino
* *Configure Rosserial:* In the Arduino code, include the necessary Rosserial headers, and configure the communication parameters, such as the serial port and the baud rate
* *Run Rosserial*: Run the Rosserial node on your ROS computer. This node handles the communication between the computer and the microcontroller
* *Upload Arduino Code*: Upload the Arduino code with the Rosserial integration to your Arduino board.

**Future goals**

Implementing CPG, improving the mechanical design to make the bot lighter Using RL to developing non slipping motion. Last but not the least, making the robot autonomous. Upgrading the circuitry of the robot. It includes replacing current electrical devices with better alternatives.egArduino ,raspi with jetson nano and STM board respectively. Using motors which give feedback as it will be needed for implementing CPG/using sensors at legs which detect elevation and depth