

Design and Development of Spherical Robot Using Pendulum Mechanism

K.Krishnamurthy

Department of Mechatronics Engineering
Kongu Engineering College, Perundurai – 638 060

B.Meenakshipriya

(Corresponding Author)

Department of Mechatronics Engineering
Kongu Engineering College, Perundurai – 638 060

K.Indhu Shree

Department of Mechatronics Engineering
Kongu Engineering College, Perundurai – 638 060

R.Vaishanth

Department of Mechatronics Engineering
Kongu Engineering College, Perundurai – 638 060

V.Sandeep

Department of Mechatronics Engineering
Kongu Engineering College, Perundurai – 638 060

V.Pramoth Vijay

Department of Mechatronics Engineering
Kongu Engineering College, Perundurai – 638 060

Abstract: This work reveals to design a spherical robot to traverse through rough and undulating terrain which is rather difficult to achieve using wheels. The spherical robot typically has an external spherical shell that acts as an enclosure to the inner drive mechanism, for active steering and motion control. The purpose of the Inner drive mechanism is to shift the center of mass of the robot in the desired direction which creates the rolling motion of the spherical shell in the specified direction. A 3D model is designed using SOLIDWORKS by considering all the dimensional constraints. The robot parts were modeled according to the dimensions of the mechanical and electrical components used. The robot inner parts are 3D printed using ABS (Acrylonitrile butadiene styrene) material and symmetrically distributed inside the spherical shell to eliminate unbalanced oscillation.

1. Introduction

Future technology is increasingly dependent on mobile robots. They have versatile capability of operating in hostile environments, carry out routine tasks like monitoring safety or guarding houses, and are some examples to be mentioned. Mobile robots do require a high level intelligence and adaptively through senses because the mission cannot be exactly preplanned in the working environment. They mostly comprise of complex systems consisting subsystems, each for handling motion, sensory, piloting, navigation and communication. Mobile robots are available in various physical forms ranging from

smallest robot in the scale of centimeter, weighing couple of grams to the biggest one, huge working machines weighting several tons. It might be necessary for the mobile robot to make physical contact with the environment by using a tool or manipulator, but in many cases only sensors are required like in monitoring or surveillance tasks.

Mobile robots may have different number of wheels according to the need of the application. For increased stability of mobile robots, add more number of wheels to it, to keep it stable even in undulating surfaces and prevent it from tipping over. But increasing the number of wheels constraints the mobility of the robot. Because the nonholonomic constraints on their wheel mechanisms prevents sideways movements without preliminary maneuvering. These vehicles are quite restricted in their motion [5] And also as the contact surface with the ground increases, there is overall reduction in efficiency of the robot and more energy is consumed. But even adding more wheels will keep the robot balanced for only a range of orientation of the underlying surface. If the inclination or undulations are greater than the range that the robot can tolerate, it will often result in tipping over of the robot. All of these problems come down to the size and the shape of the wheel, which is always a constraint in conventional mobile robots.

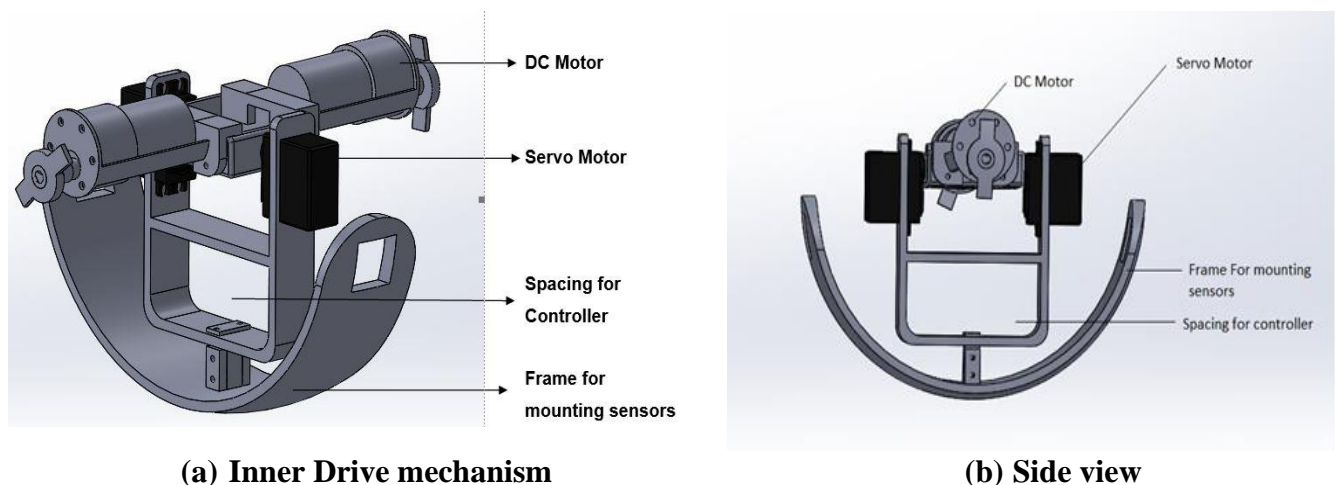
In recent years, development of another kind of mobile robot, a spherical rolling robot, has attracted the interest of many researchers. A spherical rolling robot would be practically useful because it can achieve omnidirectional motion and may have a high capability to move on a rough terrain. [6] Most of the robots resolve this issue by adding more number of wheels to the robot platform or changing the angle and orientation of the wheel setup. But this comes with a huge compromise of mobility. Higher number of wheels may be a hindrance to the mobility and allows only limited range of motion. Secondly, the process of navigating through rough terrain may be a problem when operating in remote locations. An array of sensors needed to be mounted onto the robot platform to collect various geographical data to allow motion through unknown terrain. The controller must entirely rely on the sensor feed for controlling the robot. Spherical rolling mechanisms exhibit a number of advantages with respect to wheeled and legged mechanisms. All mechanical and electrical components including the actuation mechanism are safely located inside a spherical shell rolling itself over the ground surface. As the motion of a sphere rolling without slipping over a surface is governed by nonholonomic constraints, spherical rolling robots are classified as nonholonomic mechanical systems. [4] Nonholonomic systems may or may not have a momentum conservation. The nonholonomic constraint complicates the reduction procedure. The work in [7] shows that for an invariant constraint, and through a specific connection and momentum map, we obtain a reduction process similar to the unconstrained case. The reduction of nonholonomic systems is based on factoring the dependence of symmetry. [2]

Mobility is a very crucial feature of mobile robots. The Omni-directional motion is favored, for many applications. The purpose of this work is to introduce a new mobile robot and analyze its construction and motion properties. The spherical structure is statically stable and due to the spherical shape, the robot recovers from the collisions with unknown obstacles. Any sensor can be mounted inside the spherical shell and the robot can be effectively used. But a downside of this robot is it's impossible to install a manipulator but different sensors for environment perception and navigation are easy to install. Mobility of the robot is based on disturbing the system equilibrium by unbalancing the Inside Drive Unit IDU. IDU is moved by a motor driven wheel that can be turned to cause the heading of the motion to change.

The spherical exo-skeleton provides the robot with maximal stability in the absence of orientation preference, and also the ability to roll in all directions rather than one, as provided by the wheel. The outer perimeter of the robot exo-skeleton is relatively large in dimension, thus providing the ability to roll over rough terrain with relative ease. Through proper material selection, the exoskeleton can be fitted with a variety of sensors and other hardware for sensing the environment. The internal motion mechanism will provide the robot with rapid maneuverability by enabling it to quickly accelerate and decelerate, or move with constant velocity. The propulsion mechanism will also allow the robot to climb slopes of considerable inclination and traverse terrain with significant undulations.

2. Modelling of Spherical Robot

The 3D model of the spherical robot is developed using SOLIDWORKS as shown in Figure 1. It is a solid modeler and utilizes a parametric feature-based approach to make models and assemblies. Building a model in SOLIDWORKS usually starts with a 2D sketch (although 3D sketched are available for power users). Relations are used to define attributes like tangency, parallelism, perpendicularity and concentricity. The dimensions within the sketch are often controlled independently or by relationships to other parameters inside or outside of the sketch. In an assembly, the analog to sketch relations are mates. Just as sketch relations define conditions like tangency, parallelism and concentricity with reference to sketch geometry, assembly mates define equivalent relations with reference to the individual parts or components, allowing the easy construction of assemblies.



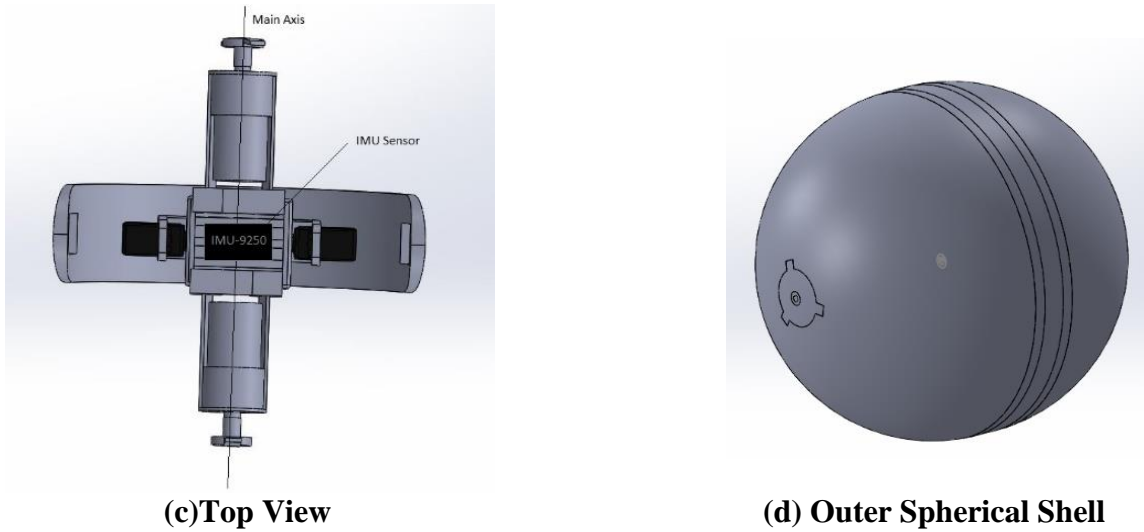


Fig 1. 3D Model of Spherical Robot

3. Materials and Methods

3.1. Motion Dynamics of the system

Mobility of the robot is based on disturbing the system equilibrium by unbalancing the Inside Drive Unit IDU [3] by shifting the mass of the system by an angle theta in the same vertical plane as shown in Figure 2. Due to this apparent shift in Center of Mass (CoM), the outer spherical shell tends to rotate in the same direction. This causes the robot to roll over. The system tends to get back to the original position to maintain the balance but to keep the robot moving in the desired direction. The mass is continuously shifted, thereby causing the robot to move by rolling in the specified direction. The bottom frame of the robot is designed to mount camera, other optional sensors and batteries. The batteries make up for the most of the mass of the system. The robot motion is achieved with the help of a pair of DC motors and a pair of Servo motors. The forward/backward motion is achieved by actuating the dc motors. The DC motors are controlled using the L298N motor driver. DC motors are given 12v DC power supply from batteries.

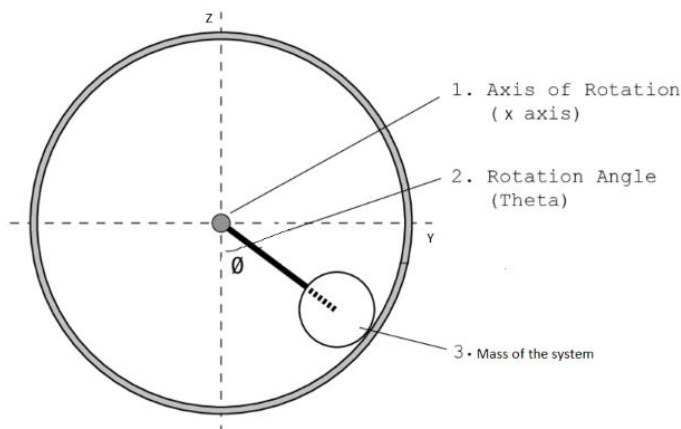
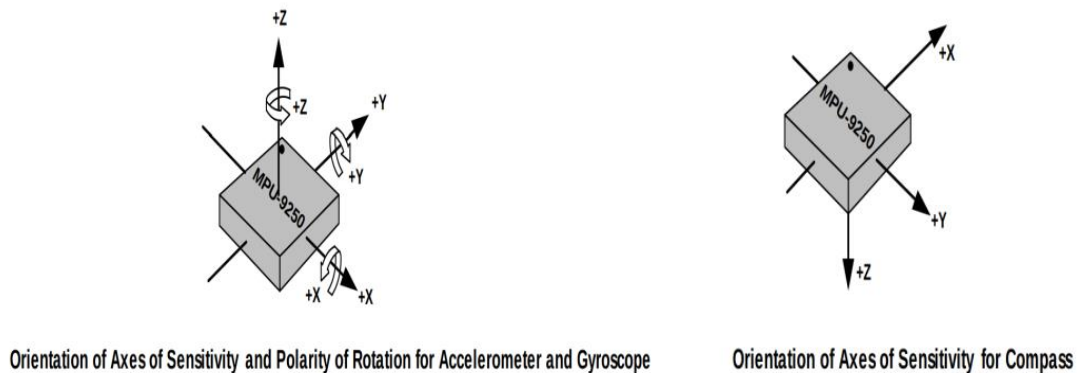


Fig 2. Motion Dynamics of the System**3.2.Orientation of the robot**

The Orientation of the robot is measured by using an Inertial Measurement Unit Sensor (IMU) as shown in Figure 3. An MPU-9250 sensor is used which consist of 3 Axis Accelerometer and 3 Axis gyroscope. The Accelerometer provides the acceleration data of the robot, whereas the Gyroscope provides the rotation about each of X, Y, Z axes. The Sensor is placed at the middle of the main frame axis of the robot and is enclosed inside a 3D printed part to provide protection and prevent the misalignment of the sensor. It is very crucial to know the orientation of the robot to control its direction of motion. Rotation about X axis of the sensor provides the pitch control and rotation about Y axis provides the Yaw control of the robot.

**Fig 3. Orientation of the Robot****3.3.Steering of Robot**

The steering of the robot is achieved by shifting COG towards the sides as shown in Figure 4. This is performed by actuating the servo motors. Actuating the servo motors tends to rotate the hanging frame about the Y axis of the robot. The initial position of the hanging frame may vary depending upon the current orientation of the robot with respect to the underlying surface. The maximum rotations about Y axis is 60 deg both clockwise and counter clockwise direction from the neutral position of the robot i.e. with a range of motion of -60 to 60 deg from neutral position. But this range of motion may vary for different inclinations of the traversing surface. This is dynamically changed by analyzing the data from the MPU-9250 Gyroscope sensor. Steering the robot by an angle of 90 deg is not possible. The robot takes a longer curved path to completely turn towards the specified direction. During the whole turning process the vertical frame is maintained with the same side tilt until the complete turn is accomplished. The robot is steered with the help of a pair of MG995 servo motors. The servo motors are controlled using Adafruit 16 channel servo driver. They are supplied with 9V power supply.

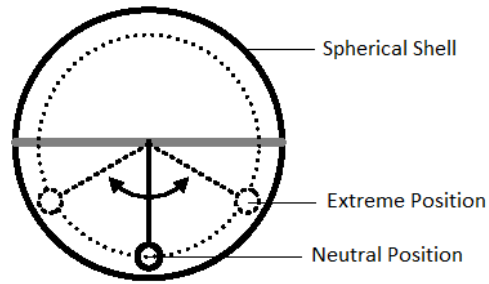


Fig 4. Steering of Robot

3.4 .Control using Remote Devices

The spherical robot can be remotely controlled using any other device fitted with a display and a Wi-Fi connection with the local server of the robot. As the robot switches ON, the Raspberry Pi runs a web server in its memory for interacting with its control system. Controlling the spherical robot is as easy as connecting the on board Wi-Fi of the device with the local server of the robot and entering the URL in any native browser. The control keys and the specific instructions are displayed as a webpage on the device that is accessing the web server of the robot. The web page displays a virtual controller to control the robot as shown in Figure 5.

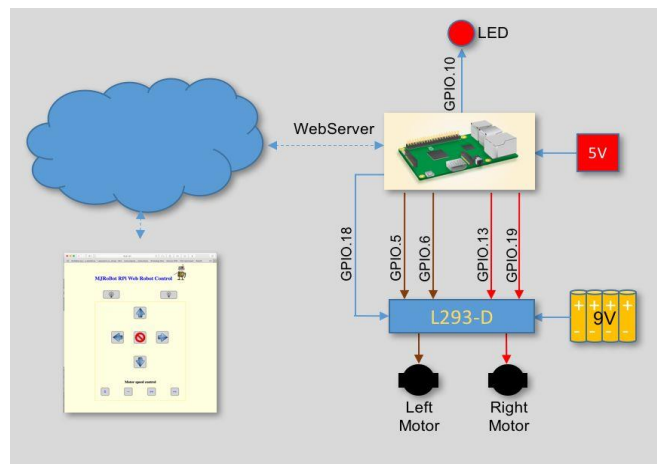


Fig 5. Deploying web server in Raspberry pi 3

4. Fabrication of Spherical Robot

4.1 Components and its Specifications

The components required for the fabrication of project are explained below in detail.

4.1.1 Raspberry Pi 3B+

The Raspberry Pi 3B+ shown in figure 6 is a tiny credit card sized computer with versatile capability. It features a powerful CPU and 1 Gigabyte of ram to satisfy all the computational needs. Due

to its versatile capability, it can be used as a controller for any projects. To facilitate the simple interfacing of external hardware with the Raspberry Pi board, it is provided with 40 GPIO pins each having its own purpose. It is also provided with an array of ports that include USB ports for communicating with external hardware. A mouse and a keyboard can be plugged in on the go for controlling and interacting with the operating system of the raspberry PI. It has a dedicated micro USB slot used as power supply. For this work, the operating system (Raspbian OS) is installed on a 16Gb SD card and inserted in the micro SD slot provided on the controller board.

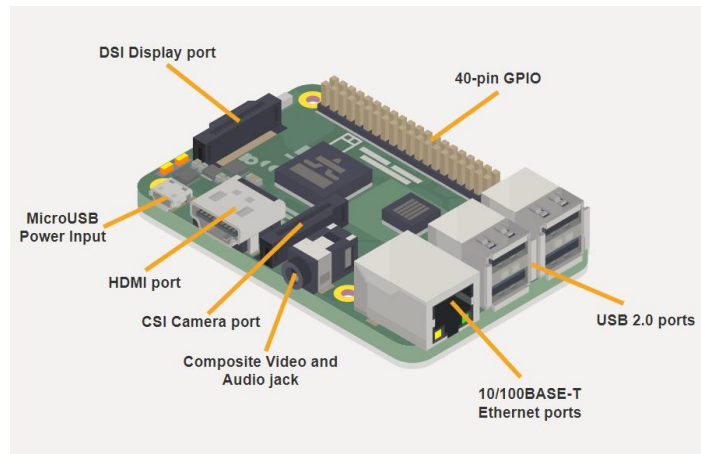


Fig 6. Raspberry PI 3 Model B+

4.1.2 MPU-9250

The MPU-9250 is a 9-axis Motion Tracking device that mixes a 3-axis gyroscope, 3-axis accelerometer, 3-axis magnetometer and a Digital Motion Processor (DMP) all in a little 3x3x1mm package available as a pin-compatible upgrade from the MPU-6515. With its dedicated I2C sensor bus, the MPU-9250 directly provides complete 9-axis Motion Fusion output. The MPU-9250 Motion Tracking device, with its 9-axis integration, on-chip Motion Fusion, and run-time calibration firmware, enables manufacturers to eliminate the costly and sophisticated selection, qualification, and system level integration of discrete devices, guaranteeing optimal motion performance for consumers. Gyroscope can measure the lean and lateral orientation of the bot whereas accelerometer can only measure the linear motion of the bot. MPU-9250 is additionally designed to interface with multiple non-inertial digital sensors, like pressure sensors, on its auxiliary I2C port. The MPU-9250 is enclosed inside a 3D printed part to avoid damage and placed at the center of the bot to stop the misalignment of the sensor. The pin diagram of MPU-9250 is shown in figure 7.

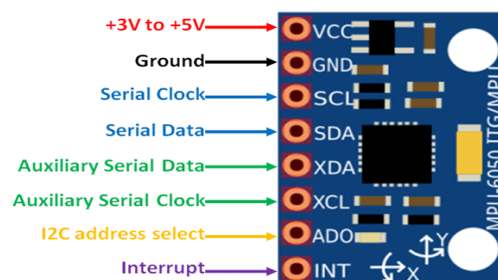


Fig 7. Pin Diagram of MPU-9250

4.1.3 Servo Motor

MG995 servo motor shown in figure 8 provides a quick control response and constant torque throughout the servo range. It's a closed-loop system where it uses positive feedback system to regulate motion and final position of the shaft. Here the device is controlled by a feedback signal generated by comparing output and reference input signal. Here, reference input signal is compared to reference output and therefore the third signal is produced by feedback system. And this third signal acts as input to regulate device. This signal is present as long as feedback signal is generated or there's difference between reference input and reference output. So, the most task of servomechanism is to take care of output of a system at desired value at presence of noises. These servos control the swaying and steering of bot within the specified direction.



Fig 8. MG995 Servo Motor

Servo motors are often driven using the PWM outputs available on most embedded MCUs. But while the Pi does have native HW support for PWM, there's just one PWM channel available to users at GPIO18. That sort of limits your options if you would like to drive more than one servo or if you furthermore may want to dim an LED or do some kind of other PWM goodness also. But Pi does have HW I2C available, which we will use to speak with a PWM driver just like the PCA9685, used on Adafruit's 16-channel 12-bit PWM/Servo Driver shown in figure 9. Using this breakout, you'll easily approach to 16 servo motors on your Raspberry Pi employing a Python library. The Adafruit servo driver is that the best solution for controlling quite one servo.

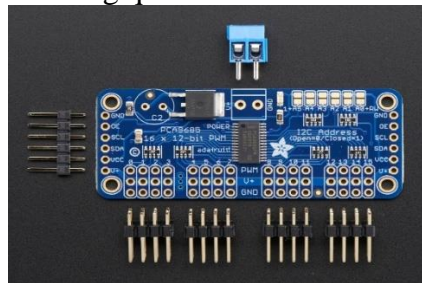


Fig 9. Adafruit Servo Motor Driver Pinout

4.1.4. DC MOTOR

Orange PM33 12v 100RPM DC motor shown in figure 10 is a simple 100 RPM 7.36N-cm side shaft DC gear motor. This is a new compact size motor at a low cost. The supply voltage range is 10-12V with the polarity markers at the base of the motor. The overall body of the motor is made up of metal. The motor has a D type shaft with a shaft length of 21mm and a diameter of 6mm. The shaft type is D and hence the coupling arrangement can be made easily. The DC motors are coupled on opposite sides of the bot to ensure the centre of mass and to control the backward or forward motion of the bot.



Fig 10. PM 33 DC Motor

4.1.5 L298N DRIVER

L298N 2A Based Motor Driver shown in figure 11 may be a higher power motor driver perfect for driving DC motors and Stepper motors. It uses the favored L298 motor driver IC and has an onboard 5V regulator which it can supply to an external circuit. It can control up to 4 DC motors, or 2 DC motors with directional and speed control.

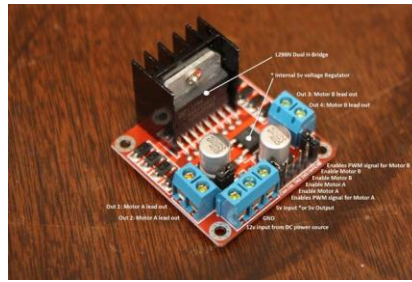


Fig 11. L298N Motor Driver

4.2. Design of Interfacing Circuit

The required components are interfaced with raspberry pi. Initially, each device is individually connected and tested for errors. The easiest way to hook the servo breakout up to your Pi is using a breadboard and connecting it to the Pi using I2C as shown in figure 12. Switching directions on the servo can cause tons of noise on the supply, and therefore the servo(s) will cause the voltage to fluctuate significantly, which may be a bad situation for the Pi. It's highly recommended to use an external 5V supply with servo motors to avoid problems caused by voltage drops on the Pi's 5V line.

Raspberry Pi is configured to connect with the IMU sensor using I2C communication as shown in figure 13. L298 is known as a dual bidirectional motor driver which is predicated on dual H-Bridge Motor driver IC. This circuit shown in figure 14 allows you to control two DC motors independently in either direction. It also provides an onboard 5V regulator that you simply can use to power your 5V circuits very conveniently.

- Connect IN1 on the L298 to the Raspberry Pi's pin number 26.
- Connect IN2 on the L298 to Raspberry Pi's pin number 20.

- Connect the ENA and 12-volt pin to a 9-volt battery.
- Make sure the grounds of the battery, Raspberry Pi, and L298 are common.

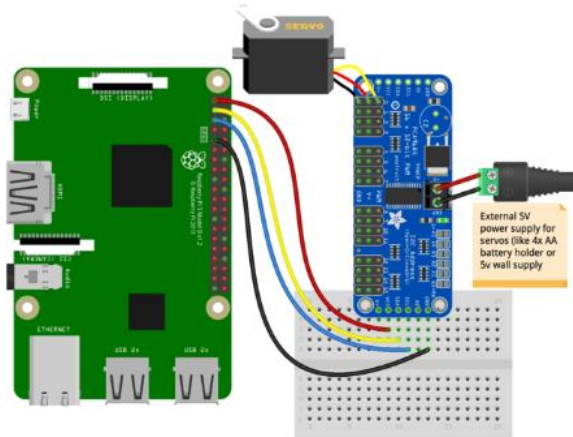


Fig 12. Raspberry PI with Adafruit Servo Driver

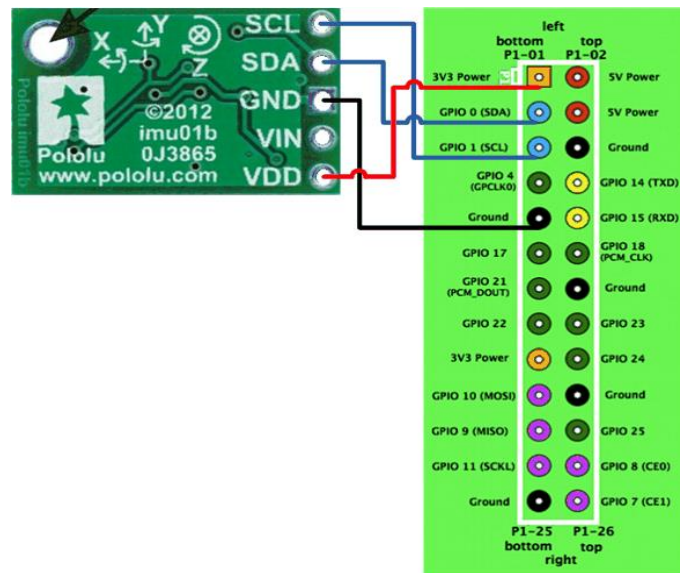


Fig 13. MPU-9250 with Raspberry Pi

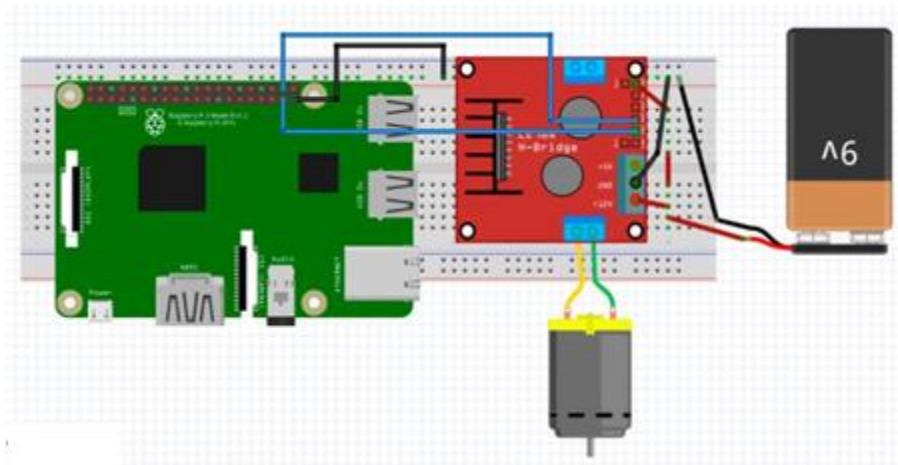


Fig 14. L298N Motor Driver with Raspberry Pi

4.3. Spherical Robot

The fabricated model of spherical mobile robot is shown in figure 15. It is developed with an objective of having improved mobility and maneuverability than existing wheeled and track-based mobile robots. It consists of an external spherical skeleton, which encloses an inner motion mechanism and an intelligent control system. The ball is so assembled that every component has a corresponding identical component placed diametrically opposite on the sphere [1]. The spherical exo-skeleton provides the robot with maximal stability in the absence of orientation preference, and also the ability to roll in all directions rather than one, which is not possible using the wheel. The outer perimeter of the robot exo-skeleton is relatively large in dimension, thus providing the ability to roll over rough terrain with relative ease.

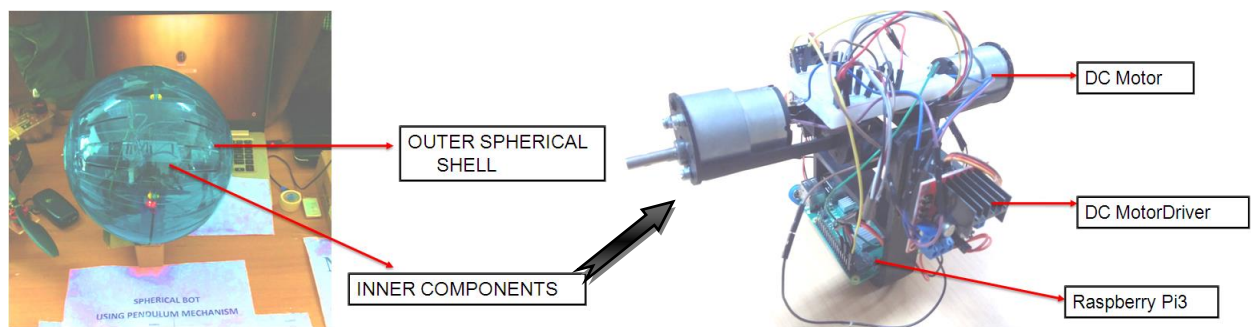


Fig 15. Fabricated Model of Spherical Robot

The motion of the robot is achieved by shifting the Center of Gravity (COG) of the robot. The COG can be shifted by actuating a pair of dc motors and a pair of Servo motors. The forward /backward motion can be achieved by shifting the COG in forward/backward direction. The 12V DC motor is actuated using a L298N motor driver. The Side swaying and steering of the robot can be controlled with the help of a pair of MG995 Servo motors. These servo motors are controlled using Adafruit 16channel Servo driver. An IMU sensor (MPU-9250) is used for measuring the rotation of the robot. This rotation

data is used for countering the side swaying of the robot. Using this data, the servo is actuated in the opposite direction.

5. Conclusions

The motivation behind this work is to design and develop a spherical mobile robot that has improved mobility and maneuverability than existing wheeled and track-based mobile robots. Incorporating a control system, interfacing various sensors with the controller and create an active balancing system to counter any unbalanced oscillations are significant of the proposed robot. Also assemble all the components to form the IDU and to ensure uniform symmetrical distribution of mass. Interfacing the robot with a remote device and remotely actuating using a wireless connection. The conventional mobile robot has lot of useful features but has many constraints that limit its mobility. To overcome such problems, the proposed Spherical Robot was developed to enable easy locomotion on undulating surfaces without the fear of the mobile robot tipping over. Also providing enhanced mobility for versatile applications and maintains the robot system in equilibrium at different orientation preferences. The capabilities of the prototype can be enhanced by

- Incorporating a camera to assist remote controlling of the robot
- Embedding sensors in the exo-skeleton to sense the data about the environment
- A reaction wheel mechanism can be incorporated to balance the side swaying of the robot as it provides a fast response time.
- Machine vision system can be integrated into the raspberry Pi for autonomous control of the robot

References

- [1] Bhattacharya, S. and Agrawal, S.K., 2000. Spherical rolling robot: A design and motion planning studies. *IEEE Transactions on Robotics and Automation*, 16(6), pp.835-839..
- [2] Gajbhiye, S. and Banavar, R.N., 2016. Geometric modeling and local controllability of a spherical mobile robot actuated by an internal pendulum. *International Journal of Robust and Nonlinear Control*, 26(11), pp.2436-2454
- [3] Halme, A., Schonberg, T. and Wang, Y., 1996, March. Motion control of a spherical mobile robot. In *Proceedings of 4th IEEE International Workshop on Advanced Motion Control-AMC'96-MIE* (Vol. 1, pp. 259-264). IEEE
- [4] Kayacan, E., Bayraktaroglu, Z.Y. and Saeys, W., 2012. Modeling and control of a spherical rolling robot: a decoupled dynamics approach. *Robotica*, 30(4), pp.671-680.
- [5] Mojab, P., 2002, May. Introducing August: a novel strategy for an omnidirectional spherical rolling robot. In *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292)* (Vol. 4, pp. 3527-3533). IEEE..
- [6] Otani, T., Urakubo, T., Maekawa, S., Tamaki, H. and Tada, Y., 2006, March. Position and attitude control of a spherical rolling robot equipped with a gyro. In *9th IEEE International Workshop on Advanced Motion Control, 2006.* (pp. 416-421). IEEE.
- [7] Ostrowski, J., Burdick, J., Lewis, A.D. and Murray, R.M., 1995, May. The mechanics of undulatory locomotion: The mixed kinematic and dynamic case. In *Proceedings of 1995 IEEE International Conference on Robotics and Automation* (Vol. 2, pp. 1945-1951). IEEE.