DESIGN AND IMPLEMENTATION OF AN EFFICIENT AUTOMATIC STABILIZED VEHICLE SYSTEM: A COMPREHENSIVE STUDY

V. Shashank, B. Tech (Mechatronics), Delhi Skill and Entrepreneurship University Okhla-II campus, New Delhi, India.
***Dr. D. Rakesh,** Ex-HOD (Mechatronics), Delhi Skill and Entrepreneurship University Okhla-II campus, New Delhi, India.
Dr. G. Charu, Assistant Professor, Delhi Skill and Entrepreneurship University Okhla-II campus, New Delhi, India.
Dr. G. Arun, Assistant Professor, Delhi Skill and Entrepreneurship University Okhla-II campus, New Delhi, India.
D. Lokesh, Software Engineer, Bosch Global Software Technologies, Pune, India.
Corresponding author, E-mail id: rkd.dite@gmail.com, rkdhammi7164@gmail.com

Abstract

This paper presents the development of a system capable of balancing a robot on two wheels. The system consists of a platform mounted on two wheels connected by a single axle, with an additional platform above it. The primary challenge is to maintain the stability of the upper platform using gyroscope and acceleration sensors, ensuring it remains horizontal. The implementation involves both hardware and software components, with a mechanical model based on a pendulum system. A PID controller is employed to find the stable inverted position, with Arduino serving as the microcontroller to send signals to motors for maintaining balance.

Keywords:
PID controller, Gyroscope, PID controller, Segway.

I. Introduction

The inverted pendulum is indeed a classic problem with a wide range of applications across various fields:
Segway: Perhaps one of the most prominent applications of the inverted pendulum concept is seen in the Segway personal transporter. By employing a sophisticated combination of sensors, motors, and control algorithms, the Segway maintains balance dynamically, enabling users to control the device through intuitive weight shifts [1]. Robotics: Inverted pendulums serve as valuable tools in robotics for the development and testing of control algorithms. They offer a simplified yet effective model for studying balance and stabilization challenges encountered in diverse robotic applications, including humanoid robots, bipedal walking robots, and drones. Through experimentation with different control strategies, engineers can refine their understanding of system dynamics and enhance the stability of robotic platforms [2]. Rehabilitation Robotics: In the realm of rehabilitation robotics, inverted pendulum-based devices play a crucial role in aiding patients in their journey to recover balance and enhance motor skills. These devices provide controlled movements and present stability challenges that are instrumental in the rehabilitation process. By engaging patients in targeted exercises, these systems facilitate progress and improve overall mobility and coordination [3]. The inverted pendulum serves as an invaluable educational tool in control system courses, where it is used to introduce fundamental concepts such as feedback control, stability, and system dynamics. Through hands-on experimentation with different control strategies, students gain practical insights into the behavior of dynamic systems and learn to design effective control algorithms. This experiential learning approach fosters a deeper understanding of control theory principles and prepares students for real-world engineering challenges [4]. Overall, the versatility and applicability of the inverted pendulum concept underscores its significance across a wide range of disciplines, from transportation and robotics to healthcare and education. By leveraging this fundamental principle, engineers and researchers continue

to innovate and advance technology in pursuit of more efficient and capable systems. Using open-source code and adapting it to suit hardware-requirements is a common practice in robotics development, as it allows for faster prototyping and leveraging the expertise of the broader community. The Arduino Nano microcontroller board is a popular choice for balancing robot projects due to its compact size, affordability, and ease of use. Limiting the intended operation and testing of the robot to indoor environments with level surfaces simplifies the control problem by reducing the complexity of the environment. It allows for focused development and testing, ensuring that the robot can reliably maintain balance and navigate within controlled conditions. Assuming the pendulum possesses a single degree of freedom further simplifies the control task by reducing the number of variables that need to be considered. This assumption enables control in one direction for both the tilt angle and the robot's position, streamlining the control algorithms and making them easier to implement and tune. By adopting these simplifications and constraints, developers can concentrate their efforts on refining the control algorithms and optimizing the robot's performance within the specified operating conditions. As the project progresses, there may be opportunities to expand the capabilities of the robot and adapt it for operation in more complex environments. However, starting with a focused scope allows for incremental development and gradual integration of additional features and functionalities.

II. Related Work

Hellman et al. highlighted in their research that robots of this nature can be conceptualized based on the physical principles of an inverted pendulum [5]. Achieving stability in such systems requires active control mechanisms. By employing the Arduino Uno microcontroller along with reliable angular and positional data, stability can be attained through controller implementation. One notable controller for this purpose is the LQR (Linear Quadratic Regulator), known for its modern and efficient performance. As a state-space feedback controller, the accuracy of the model is crucial since the output signal is dependent on it. However, during the validation process, a PID (Proportional-Integral-Derivative) regulator was utilized. The results indicated that the model's reliability was not yet satisfactory. For future testing and improvements, it would be advantageous to consider using an encoder to precisely track the servo's angular position or potentially explore the utilization of a stepper motor. These enhancements could contribute to refining the system's performance and increasing its reliability for practical applications. Shams et al. proposed a design for a two-wheeled balancing robot with a single axle connecting the wheels and two platforms mounted on it [6]. The upper platform is not inherently stable, and the goal was to balance it horizontally using distance sensors. The microcontroller, in this case an Arduino, would detect any inclination of the platform and send signals to the motors to adjust accordingly. Initially, the focus was on achieving balance on the two wheels. If the platform inclined, the microcontroller would command the motors to move forward or backward to counteract the tilt and maintain horizontal alignment. However, some experiments were left incomplete, presenting a significant drawback to the project's overall progress. Without conclusive results, drawing appropriate conclusions became challenging. Despite efforts, oscillation issues persisted within the system, indicating the need for further refinement and future work to achieve stability. Addressing these challenges and completing the pending experiments will be essential to advancing towards a stable solution and obtaining meaningful conclusions. Chandrasekar et al. designed a robot based on the Arduino microcontroller [7]. This robot achieves balanced and controlled movement initially and can subsequently avoid obstacles in three directions. The controller was developed progressively, emphasizing the importance of balancing and obstacle avoidance for enhancing the controller's performance. N. Meheswara Venkata Sai et al. [8] conducted research focusing on the design of a precise sensor filtering mechanism using a complementary filter to determine the pitch angle. This filter effectively eliminated gyroscopic drift and sensor noise. The control loop implementation on the Arduino Uno performed well, running at 530 Hz (± 20 Hz). Software PWM, executed through interrupt service routines of the Arduino controller, facilitated PWM control with a period of 16 milliseconds and duty cycle adjustment. The control loop, along with the PID controller, functioned as a closedloop feedback mechanism to balance the robot, maintaining a maximum oscillation amplitude of 1 cm (± 0.3 cm) and a balance time of approximately 15 seconds. The robot underwent testing in various scenarios, successfully balancing with different types of objects such as water and medicines. Designed for portability, the robot serves as a versatile servant robot. Stabilization provided by the reaction wheel is limited by the torque from the reaction wheel motor. Future plans involve utilizing a rotating disc and its gyroscopic precession for enhanced balancing, offering a more stable design with higher restoring torque. Attention to detail in the alignment of rotary axes, brackets, flexible couplings, and mounting is crucial. Furthermore, the incorporation of a fuzzy logic controller could enhance control flexibility and accuracy, expanding the capabilities of the system.

By studying the above papers, we tried to develop and implement a more efficient design of a two-wheel automatic stable robot which is cost-effective, oscillation-free and less prone to error.

III.Methodology

To enhance a balancing robot's performance:

3.1 CAD Model for Parameter Acquisition:

Importance: Accurate parameters (e.g., centre of mass) ensure simulation fidelity.

Process: Use CAD software to create a detailed model, matching physical dimensions and material properties.

Verification: Validate CAD model against physical measurements for consistency.

3.2Evaluation of Sensor Precision:

Importance: Sensor accuracy directly affects control algorithm performance.

Evaluation: Conduct calibration and characterization tests to assess accuracy, resolution, noise, and drift.

Verification: Validate sensor performance against reference measurements in real-world conditions.

By ensuring precision in both system modelling and sensor data, experiments in simulated environments can closely mirror real-world behaviour. Implementing a PID controller enables direct comparison between simulated and physical responses, aiding in performance assessment.

IV. System Design-An Overview

The construction of Automatic Stable Vehicles (ASVs) draws upon principles similar to those used in balancing inverted pendulums. Balancing an inverted pendulum is a classic problem in dynamics and control theory, often used as a benchmark for testing various control algorithms like PID controllers, neural networks, fuzzy control, and genetic algorithms. This problem can be extended to include multiple links, allowing for controlled motion of the cart while maintaining pendulum balance, or even balancing a cart-pendulum system on a see-saw. Moreover, the concept of the inverted pendulum finds relevance in rocket or missile guidance systems, where aerodynamic instability arises due to the center of gravity being situated behind the center of drag, presenting an analogous challenge. Notably, self-balancing transportation devices like the Segway PT leverage technology that addresses this problem. This paper details the design and implementation of an ASV, which inherently tackles an unstable system. It adopts the fundamental model of an inverted pendulum balanced on two wheels. The report outlines the derivation of the system model and establishes the framework for the robot's control system. Additionally, it showcases the complete implementation of a control system that stabilizes the robot, with the construction based on the utilization of an Arduino Microcontroller [6].

V.Working of System (Figure 1)

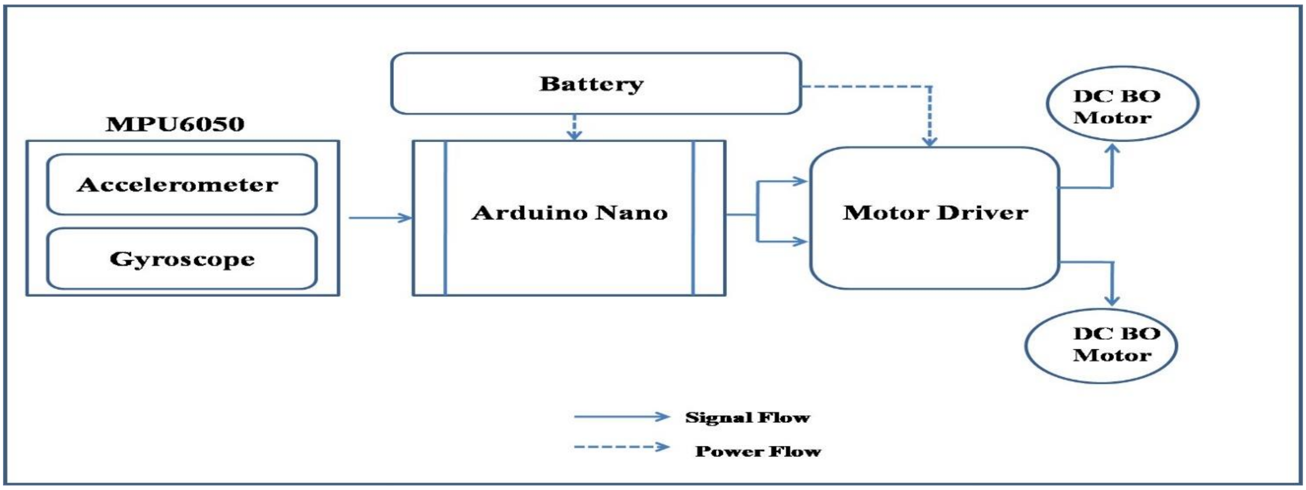


Figure 1: Block diagram of Automatic Stable Vehicle

The sensor system sends analog input voltages to the Arduino, ranging from 0V to 5V. The Arduino reads these analog inputs and converts them to digital using its built-in Analog-to-Digital Converter (ADC). Subsequently, it calculates the difference between these values (sensor1 value - sensor2 value) and uses this difference, along with the previous difference from the last loop execution, in a PID control mechanism to determine the PWM output that should be fed to the motor driver. To enhance the effect of the differential component (D) in the PID control, a time measuring function (milliseconds) was incorporated into the code. This function measures time in milliseconds, allowing for regular measurement of the difference between sensor values after fixed intervals (e.g., 10 milliseconds). This ensures that the change in difference (i.e., difference - previous difference) is substantial. The PID control utilized in the Arduino code focuses on the proportional (P) and differential (D) components, with the integral (I) component omitted as it was found that P+D was sufficient to balance the robot effectively. The motor driver, powered by a 12V power supply, receives PWM signals ranging from 0V to 5V from the Arduino. The motor driver, specifically the L298N Driver, scales this PWM signal to the required range of 0V to 12V for the motors. This scaling is necessary as the motors require a 12V power supply and high current, which cannot be provided directly from the microcontroller's output ports. When the robot tilts, causing a decrease in the height of sensor 1, the output PWM is adjusted accordingly to drive the motors in a direction that accelerates the robot towards the sensor 1 side. This acceleration induces a torque, opposing the torque due to gravity, and effectively brings the robot platform back to a horizontal level. Overall, this control mechanism allows the robot to balance itself by adjusting the motor's speed based on sensor feedback, effectively countering any deviations from the desired position.

VI.System Components

6.1 Mechanical Components

6.1.1 Base Plates

The design of the Automatic Stable Vehicle (ASV) incorporates three base plates, each serving a specific purpose:

- DC BO Motor Clamp Plate: Secures two BO motors in place, ensuring alignment and stability for proper vehicle functioning.
- Electronics Mounting Frame Plate: Provides a frame for mounting essential electronic components like Arduino Nano, Bluetooth module, sensors, and motor drivers. Facilitates efficient arrangement and connectivity.
- Weight Demonstration Top Plate: Designed to add weight for demonstration purposes, allowing users to observe stability and performance under different load conditions. Enables easy adjustment of the vehicle's centre of gravity.

All plates are made of durable acrylic sheets, offering structural support, transparency for visual inspection, and ease of fabrication. The incorporation of these base plates enhances functionality, stability, and versatility of the ASV.

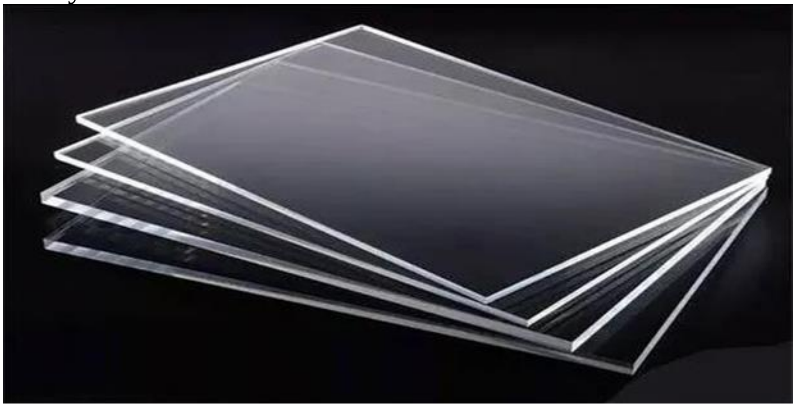


Figure 2: Acrylic sheets (Base plates) [10]

6.1.2 Wheels

The wheels for the Automatic Stable Vehicle (ASV) consist of two main components: rims and tires, each serving a specific function:

Rim: Constructed from hard plastic for high yield strength and durability. Reduces overall weight, enhancing acceleration and manoeuvrability. Provides structural integrity to withstand operational stresses. Tire: Made of rubber for high friction and excellent traction. Ensures stability and control on different surfaces. Enhances manoeuvrability and steering precision. Assembly: Rubber tires securely glued to plastic rims for a strong connection. Prevents slippage or detachment during operation. Optimizes performance with a balance of strength, weight, and traction. Overall, the wheel design prioritizes lightweight construction, durability, and traction to improve the ASV's acceleration, stability, and manoeuvring capabilities across various conditions.

Figure 3: Wheels [11]

6.1.3 Standoff (Figure 4) [12]

A standoff is a threaded separator used to keep two objects spaced apart in various projects. It resembles a complicated screw and can be made from metal, plastic, or other materials. Standoffs come in different styles, some threaded for screws and others with adhesive backing. They are used to prevent objects from bumping into each other and can also serve as spacers on shelves or countertops [13].

Figure 4: Standoff [12]

6.1.4 Assembly of Vehicle

Four standoff rods and nuts are utilized to stack the base plates vertically and secure them tightly. These standoffs also clamp the motors in place. The wheels are joined by a radius difference between the motor shaft and the inner hole on the rim. A 3D CAD model of the assembly depicts this setup.

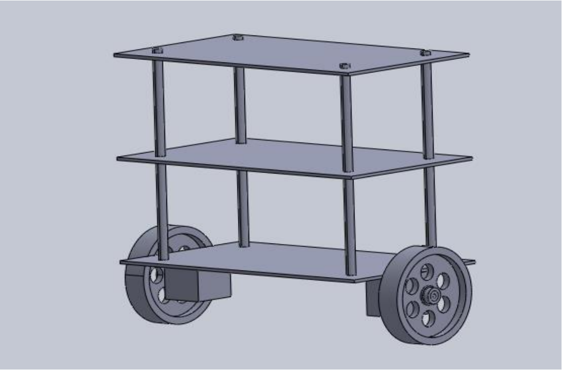


Figure 5: 3-D View of ASV frame designed in SolidWorks

6.2 Electronic Components

6.2.1 Arduino Nano

The Arduino Nano is a small, complete, and breadboard-friendly board based on the ATmega328 (Arduino Nano 3.x). It has more or less the same functionality of the Arduino Duemilanove, but in a different package. It lacks only a DC power jack, and works with a Mini-B USB cable instead of a standard one [13].

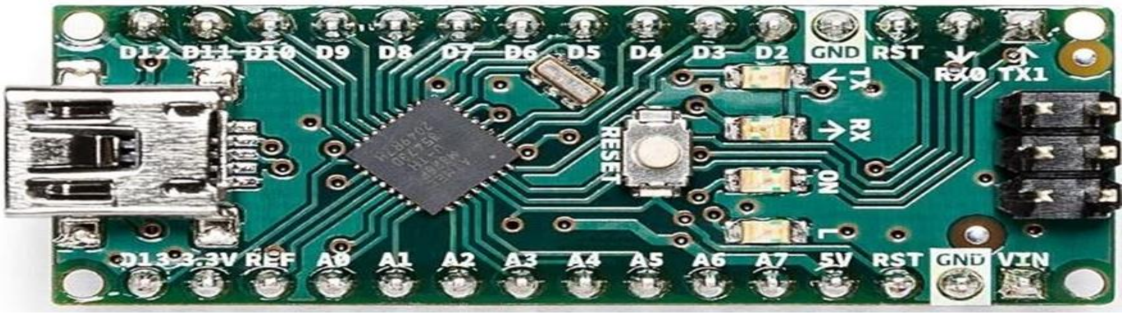


Figure 6: Arduino Nano [13]

6.2.2 Motor Driver

The L298N Motor Driver module includes an L298 Motor Driver IC, a 78M05 Voltage Regulator, resistors, capacitors, a Power LED, and a 5V jumper in one package. The 78M05 Voltage Regulator is activated only when the jumper is connected. If the power supply is 12V or lower, the internal circuitry is powered by the regulator, allowing the 5V pin to power the microcontroller. However, if the power supply exceeds 12V, the jumper should be removed, and a separate 5V source should be used. The ENA and ENB pins control motor speed, while the IN1 & IN2 and IN3 & IN4 pins control motor direction [14].

Figure 6: Motor Driver [14]

6.2.3 MPU 6050 sensor

The module features the MPU6050, a compact 6-axis Motion Tracking chip, combining a 3-axis gyroscope, 3-axis accelerometer, and Digital Motion Processor (DMP) in a small 4mm x 4mm package. It measures angular momentum, static acceleration due to gravity, and dynamic acceleration from motion or vibration. With an onboard LD3985 3.3V regulator, it's compatible with 5V logic microcontrollers like Arduino. The MPU6050 consumes less than 3.6mA during measurements and only 5µA when idle, making it suitable for battery-powered devices [15]

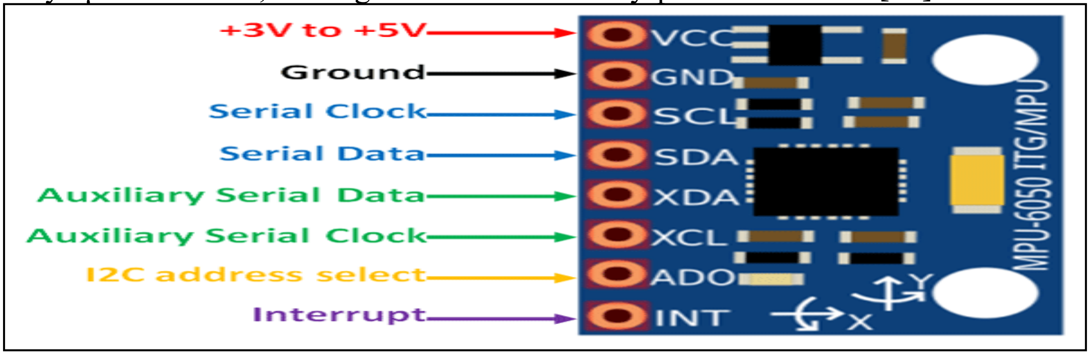


Figure 7: Pin diagram of MPU 6050 sensor [15]

6.2.4 Battery Operated Motor (BO Motor)

The 300 RPM BO Motor Plastic Gear Motor from the BO series offers high torque and RPM performance, even at low voltages. Its small shaft and compatible wheels make it versatile for various applications or robots. With mounting holes on the body and lightweight design, it's easy to integrate into circuits. Compatible with multiple wheel sizes, it's suitable for different setups. It's lightweight, absorbs shocks and vibrations, and operates with minimal lubrication due to its inherent properties, including low friction [16].

Figure 8: BO Motor [16]

VII. Implementation of Motor Driver and BO Motor with Arduino Nano

To wire an L298N motor driver and BO motor with Arduino Nano, follow these steps:

Gather Components: We'll need an Arduino Nano, L298N motor driver module, BO motor, breadboard, jumper wires, and a power supply.

7.1 Power Supply Connection:

- Connect the power supply to the buck converter, stepping down the 12V supply to 9V, and then provide it to the L298N motor driver.

7.2Arduino Nano to L298N Motor Driver Connection:

- Link the 5V pin on the Arduino Nano to the 5V pin on the L298N module.
- Connect the GND pin on the Arduino Nano to the GND pin on the L298N module.

7.3Motor Connection to L298N Motor Driver:

- Connect one terminal of the BO motor to the OUT1 terminal on the L298N module.
- Attach the other terminal of the BO motor to the OUT2 terminal on the L298N module.

7.4Controlling Motor Direction with L298N:

- Choose two digital pins on the Arduino Nano (e.g., Pin 2 and Pin 3) and connect them to the IN1 and IN2 pins on the L298N module.
- In your Arduino code, set the logic levels on these pins to control the motor's direction. For example, setting IN1 to HIGH and IN2 to LOW rotates the motor in one direction.

7.5 Controlling Motor Speed with L298N:

- Choose a digital pin on the Arduino Nano (e.g., Pin 9) and connect it to the ENA pin on the L298N module.
- In your Arduino code, use the analogWrite() function on this pin to control the motor's speed. Adjust the PWM value between 0 and 255 for varying speeds.

VIII. Implementation of MPU6050 sensor with Arduino Nano

The MPU6050 sensor is placed on the top acrylic sheet to detect the robot's orientation more accurately, aiding in effective balance. To wire it to the Arduino Nano: a) Connect VCC to Nano's 5V pin. b) Connect GND to Nano's GND pin. c) Connect SDA to Nano's A4 pin. d) Connect SCL to Nano's A5 pin.

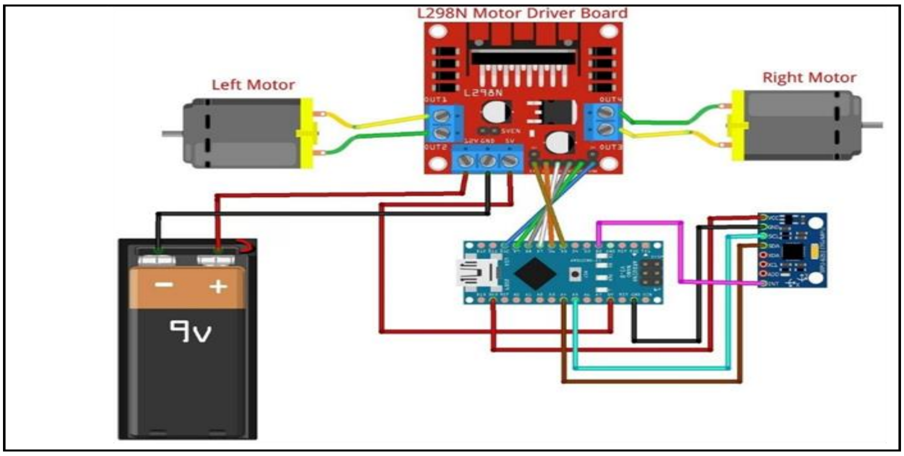


Figure 9: Overall Circuit diagram

IX.Balancing Test and Tuning

Inverted pendulum vehicles, like automatic stable vehicles, balance similarly to balancing a stick on your finger. Wheel rotation counterbalances falling. We adjust PID values until the robot stabilizes and regains balance. The code simplifies this with an Arduino PID library. We manually input P, I, D values and test stability by uploading the code to Arduino. Start with Ki and Kd as 0, then set Kp. After several attempts, adjust Kd to minimize oscillations without altering Kp, then fine-tune Ki manually.

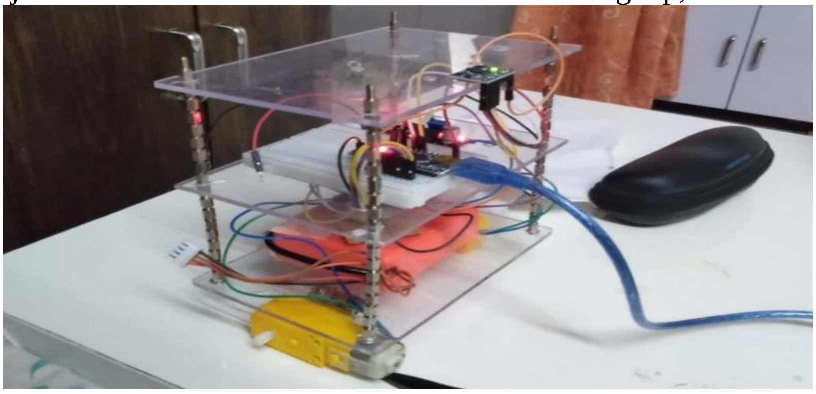
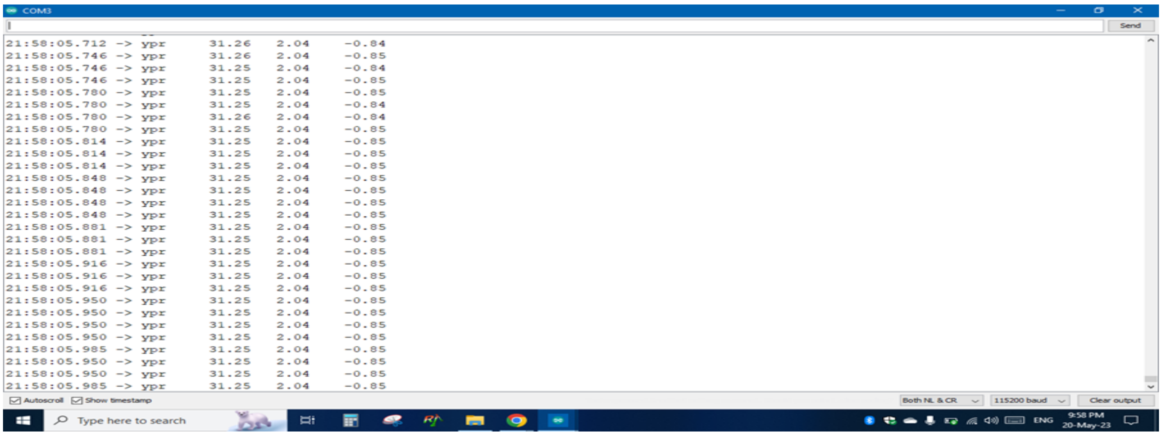


Figure 10: Balance Testing and Tuning



```
21:58:05.712 -> ypr	31.26	2.04	-0.84
21:58:05.746 -> ypr	31.26	2.04	-0.85
21:58:05.746 -> ypr	31.25	2.04	-0.84
21:58:05.746 -> ypr	31.25	2.04	-0.85
21:58:05.780 -> ypr	31.25	2.04	-0.85
21:58:05.780 -> ypr	31.25	2.04	-0.84
21:58:05.780 -> ypr	31.26	2.04	-0.84
21:58:05.780 -> ypr	31.25	2.04	-0.85
21:58:05.814 -> ypr	31.25	2.04	-0.85
21:58:05.814 -> ypr	31.25	2.04	-0.85
21:58:05.814 -> ypr	31.25	2.04	-0.85
21:58:05.848 -> ypr	31.25	2.04	-0.85
21:58:05.848 -> ypr	31.25	2.04	-0.85
21:58:05.848 -> ypr	31.25	2.04	-0.85
21:58:05.848 -> ypr	31.25	2.04	-0.85
21:58:05.881 -> ypr	31.25	2.04	-0.85
21:58:05.881 -> ypr	31.25	2.04	-0.85
21:58:05.881 -> ypr	31.25	2.04	-0.85
21:58:05.916 -> ypr	31.25	2.04	-0.85
21:58:05.916 -> ypr	31.25	2.04	-0.85
21:58:05.916 -> ypr	31.25	2.04	-0.85
21:58:05.950 -> ypr	31.25	2.04	-0.85
21:58:05.950 -> ypr	31.25	2.04	-0.85
21:58:05.950 -> ypr	31.25	2.04	-0.85
21:58:05.950 -> ypr	31.25	2.04	-0.85
21:58:05.985 -> ypr	31.25	2.04	-0.85
21:58:05.950 -> ypr	31.25	2.04	-0.85
21:58:05.950 -> ypr	31.25	2.04	-0.85
21:58:05.985 -> ypr	31.25	2.04	-0.85
```

Figure 11: Ideal rest point roll, pitch and yaw values

X. Result and Conclusion

We have successfully developed an efficient design for a two-wheel automatic stable vehicle—a two-wheel robot capable of self-balancing. This design is cost-effective, oscillation-free, and less prone to errors. It utilizes the inverted pendulum concept, like a Segway, to maintain balance when pushed. We have presented detailed information on component requirements, functions, and connections. Potential future enhancements could involve incorporating quadrature optical encoders for precise motor speed readings and potentiometers for fine-tuning the PID control system constants, thereby further improving stability.

XI.Future Scope

The future of Automatic Stable Vehicles is promising, with ongoing research and development. Expected advancements include:

- Increased autonomy, enabling navigation without human input.
- Improved efficiency through advanced materials and technologies.
- Enhanced safety with advanced sensors and control systems.
- Greater connectivity, allowing communication with other vehicles and infrastructure.
- Personalization features tailored to individual preferences.
- Expanded usage in various sectors like agriculture, military, and logistics. Overall, the future holds greater autonomy, efficiency, safety, connectivity, personalization, and usage for Automatic Stable Vehicles.

Acknowledgement: We acknowledge the work of Abhishek Kumar, Aman, Purushottam Jha and Ghulam Rabbani who also contributed to this comprehensive study and implementation of this project.

References

[1]J. H. Lienhard and J. H. Lienhard, "Segway Human Transporter," in Mechanical Engineering Review Manual: A Complete Review Course for the PEExamination for Mechanical Engineers, 12th ed., Professional Publications, Inc.,2006, pp. 7-41.

[2]T. B. Lauwers, et al., "Experimental Validation of the Co-design of Actuator and Controller for a Bipedal Walking Robot," in IEEE International Conference on Robotics and Automation (ICRA), 2004, vol. 3, pp. 2964-2969.

[3]T. C. W. Land, et al., "The Balance Trainer: A Computer-Aided RehabilitationDevice for Balance and Weight Bearing Measurement," in Journal of Medical Engineering & Technology, vol. 31, no. 3, pp. 200-206, 2007.

[4]S. H. Zak, "Inverted Pendulum Example," in Systems and Control, Oxford University Press, 2003, pp. 51-53.

[5]H. Hellman, et al., “Two-Wheeled Self-Balancing Robot”, Available: https://www.diva-portal.org/smash/get/diva2:916184/FULLTEXT01.pdf
[6]Gothe S. Shams, et al., “Self-Balancing Robot”, Available: https://core.ac.uk/download/pdf/55305134.pdf
[7]G. Chandrasekar, et al., “Self-Balancing Robot”, Available: https://ijarsct.co.in/Paper2144.pdf, International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
[8]N. Meheswara Venkata Sai, et al., “Design and Development of Self Balancing-Robot”,Available: http://indusedu.org/pdfs/IJREISS/IJREISS_3094_97729.pdf , International Journal of Research in Engineering, IT and Social Science, ISSN 2250- Volume 09, Special Issue 2, May 2019, Page 20-32.
[9]S.Patil et al., “Self-Balancing Robot”,Available: https://www.ijert.org/research/self-balancing-robot-IJERTCONV9IS03053.pdf, International Journal of Engineering Research & Technology (IJERT) ISSN:2278-0181Anitha, P.; Chakravarthy, T. Agricultural Crop Yield Prediction using Artificial Neural Network with Feed Forward Algorithm. Int. J. Comput. Sci. Eng. 2018, 6, 178–181.
[10]“Acrylic-Sheets”,[Online],Available: https://www.indiamart.com/proddetail/acrylic-transparent-sheet- 20299793712.html
[11]“Wheels”,[Online],Available: https://robocraze.com/products/bo-motor-wheels-4-pcs
[12]“Standoff”,[Online],Available :https://www.foamcoreprint.com/blog/what-are-standoffs
[13]“Standoff”,[Online],Available: https://www.amazon.in/Standoff-Securely- Accurately-Motherboard-Furniture/dp/B08QSB267C
[14]“Arduino-Nano”,[Online],Available:https://store.arduino.cc/products/arduino-nano.
[15]“L298N-Motor-Driver”,[Online],Available: https://components101.com/modules/l293n-motor-driver-moduleReddy, T.; Dutta, M. Impact of Agricultural Inputs on Agricultural GDP in Indian Economy. Theor. Econ. Lett. 2018, 8, 1840–1853.
[16] “Pin-diagram-MPU6050-Sensor”,[Online], Available: https://components101.com/sensors/mpu6050-module