iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** VIII **Month of publication:** August 2026

DOI: https://doi.org/10.22214/ijraset.2026.84595

Design, Modeling, and PID Control Implementation of an Inverted Pendulum Two-Wheeled Self-Balancing Mobile Robot

Mr. Manish Kumar[1], Mr. Aniket Kumar[2], Dr. Madan Kumar[3], Mrs Madhuri Kumari[4], Amit Kumar[5], Abhay Kumar[6], Anmol Soni[7], MD Imran[7], MD Arfat[8]
[1, 2, 3, 4]*Lecturer, Department of Mechanical Engineering and Civil Engineering, Government Polytechnic Muzaffarpur*
[4, 5, 6, 7, 8]*Diploma student, Department of Mechanical Engineering, Government Polytechnic Muzaffarpur*

Abstract—Two-wheeled self-balancing robots represent a classic benchmark for underactuated, non-linear dynamic feedback control systems based on the inverted pendulum model. This paper presents the end-to-end design, kinematic modeling, hardware assembly, and experimental calibration of a compact two-wheeled self-balancing robot platform. The proposed system utilizes an Arduino Nano ATmega328P microcontroller as the core computing unit, coupled with a 6-axis MEMS MPU6050 Inertial Measurement Unit (IMU) featuring an onboard Digital Motion Processor (DMP) for real-time attitude sensing. Actuation is achieved via two DC geared motors driven by an L298N dual H-bridge module. A proportional-integral-derivative (PID) control scheme is implemented to compute corrective torque vectors and balance the chassis dynamically along its pitch axis. Experimental evaluation demonstrates that the calibrated controller (Kp = 21, Ki = 140, Kd = 0.9) maintains stable upright posture with recovery times between 1.5 and 3.0 seconds following external impulsive disturbances. The system provides a scalable, cost-effective framework for educational mechatronics and mobile autonomous transport platforms.
Keywords: Self-Balancing Robot, Inverted Pendulum, PID Control, MPU6050, Arduino Nano, MEMS IMU, L298N Motor Driver, Mechatronics.

I. INTRODUCTION

Mobile robotics has seen accelerated advancements in recent years, shifting from rigid, stationary industrial manipulators toward autonomous mobile entities capable of navigating unstructured human environments. A persistent challenge in modern robotics control is the stabilization of underactuated systems—systems possessing fewer independent control actuators than degrees of freedom (DoF). A premier paradigm of such systems is the inverted pendulum, which serves as the physical and mathematical foundation for two-wheeled self-balancing robots.

Unlike multi-wheeled or tracked platforms that rely on statically stable configurations, a two-wheeled self-balancing robot is inherently unstable. Its center of mass lies vertically above its pivot axle, requiring continuous, dynamic torque adjustments to remain upright. Devices leveraging this dynamic balance operational principle range from personal transporters (e.g., Segway) to modern narrow-aisle warehouse logistical robots and agile humanoids.

This paper details the engineering design, hardware execution, mathematical formulation, and experimental parameter tuning of a lightweight, highly efficient two-wheeled self-balancing robot. Integrating mechanical structure design, Micro-Electro-Mechanical Systems (MEMS) sensor fusion, embedded software engineering, and classical closed-loop control theory, this study demonstrates a fully functional prototype executed under constrained resource environments.

II. RELATED WORK AND LITERATURE REVIEW

The inverted pendulum problem has served as a benchmark in control theory literature for decades. Early pioneering theoretical contributions by Åström and Furuta (1996) explored energy-based swing-up and stabilization control mechanisms for rotary inverted pendulums. The practical commercial utility of two-wheeled dynamically balanced transport was pioneered by Dean Kamen with the introduction of the Segway Human Transporter in 2001, demonstrating that high-frequency electronic feedback systems could effectively replace static mechanical stability.

Grasser et al. (2002) designed 'JOE', a mobile inverted pendulum, formulating detailed state-space decoupled equations for decoupled balance and trajectory tracking.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VIII Aug 2026- Available at www.ijraset.com

In modern open-source robotics, microcontrollers such as the ATmega328P coupled with MEMS-based Inertial Measurement Units (IMUs) like the InvenSense MPU6050 have democratized low-cost balance platform research. Classical Proportional-Integral-Derivative (PID) algorithms remain the prevailing standard for industrial dynamic stabilization due to their computational efficiency and robust real-time response.

III. SYSTEM ARCHITECTURE AND HARDWARE INTEGRATION

The hardware design balances physical structural rigidity, optimum mass distribution, rapid execution frequency, and power regulation. The mechatronic system comprises three primary subsystems: the processing and sensing unit, the actuation and drive stage, and the mechanical chassis.

A. Controller and Sensing Electronics

The core processing hub is an Arduino Nano board featuring an ATmega328P 8-bit AVR microcontroller operating at 16 MHz. Pitch angle detection is governed by an MPU6050 6-axis IMU combining a 3-axis MEMS accelerometer and a 3-axis MEMS gyroscope. To eliminate high-frequency mechanical vibration noise and gyroscopic drift, the onboard Digital Motion Processor (DMP) computes fused attitude orientation (quaternions) directly, communicating with the microcontroller via an Inter-Integrated Circuit (I2C) bus running at 400 kHz.

B. Actuation and Power Distribution

Motion drive is provided by two DC geared TT motors operating at a nominal voltage of 6V–9V, delivering up to 200 RPM with high torque gear reduction. The motors drive high-traction rubber-tyred wheels. Power amplification and direction inversion are managed via an L298N dual H-bridge motor driver capable of handling up to 2A continuous output per channel. System electrical energy is supplied by a 9V Li-Ion battery pack connected to the driver module, with regulated logic power fed back to the Arduino and IMU.

TABLE 1: Hardware Specifications and Parameter Configuration

Subsystem Component	Model / Specification	Key Operational Parameters
Microcontroller Unit	Arduino Nano (ATmega328P)	16 MHz clock, 32 KB Flash, 5V operating logic
Inertial Measurement Unit	MPU6050 (MEMS Gyro + Accel)	6-DoF, I2C interface, Onboard DMP, 400 kHz bus
Motor Driver IC	L298N Dual H-Bridge Module	2A per channel max, PWM speed control, 5V logic reg.
Drive Motors	Dual DC Geared TT Motors	Rated 3V-9V DC, ~200 RPM at load, gear-reduction torque
Tires / Wheels	Rubber Rubber-Tyre Wheels	High coefficient of friction on smooth interior surfaces
Power Source	9V Li-Ion Battery Pack	Independent motor rail feed with regulated logic stepping

IV. MATHEMATICAL MODELING AND PID CONTROL SCHEME

A. Inverted Pendulum Dynamics

The robot is modeled as an inverted pendulum mounted on a cart capable of 1D horizontal translational movement. Let m represent the mass of the chassis, L the distance from the wheel axis to the center of mass, J the moment of inertia, and theta the inclination angle from the vertical setpoint. The continuous-time differential equation governing rotational dynamics around the tilt pivot is expressed as:

$$J * (d^2\theta/dt^2) - m * g * L * \sin(\theta) = - u(t)$$

where g is acceleration due to gravity, and u(t) is the corrective control torque applied by the motor wheels against the ground surface. Under small-angle approximation ($\sin(\theta) \approx \theta$), the system linearizes around the unstable equilibrium point $\theta = 0$.

B. Closed-Loop PID Control Strategy

To stabilize the unstable equilibrium point, a discrete parallel Proportional-Integral-Derivative (PID) controller processes the orientation error e(t) = θ_setpoint - θ_actual. The mathematical representation of the generated control output u(t) is defined as:

$$u(t) = K_p \cdot e(t) + K_i \cdot \int e(\tau)\, d\tau + K_d \cdot \frac{de(t)}{dt}$$

Where Kp supplies immediate proportional restoring force, Ki eliminates steady-state tilt error accumulation caused by mechanical weight imbalance, and Kd dampens systemic oscillations by evaluating the rate of change of tilt angle.

TABLE 2: Microcontroller Hardware Pin Mapping

Peripheral Component	Peripheral Pin	Arduino Nano Interfacing Pin
MPU6050 IMU	SDA / SCL / INT	A4 (I2C SDA) / A5 (I2C SCL) / D2 (Hardware Interrupt)
MPU6050 Power	VCC / GND	5V / GND
L298N Control Inputs	IN1 / IN2 (Motor A)	Digital Pin D6 / Digital Pin D9
L298N Control Inputs	IN3 / IN4 (Motor B)	Digital Pin D10 / Digital Pin D11
L298N Speed Enables	ENA / ENB (PWM)	Digital Pin D5 (PWM) / Digital Pin D3 (PWM)
L298N Power Rail	12V Terminal / GND	9V Battery (+) / 9V Battery (-) & Common GND
L298N Logic Output	5V Regulated Out	Arduino Vin Pin

V. MECHANICAL DESIGN AND SYSTEM ASSEMBLY

The physical chassis utilizes a multi-tiered structure fabricated from lightweight foam boards reinforced with vertical blue acrylic structural columns. This multi-level architectural strategy raises the system's moment of inertia and elevates the center of mass, thereby increasing the natural time constant of falling and providing the controller ample time to execute corrective maneuvers.

The lower tier houses the heavy actuators (TT geared motors) and the L298N motor driver module, keeping the low-level drive electronics close to the wheel axle. The upper tier supports the micro-controller unit and the MPU6050 IMU. Firm mechanical mounting of the IMU was crucial to prevent parasitic chassis vibration from corrupting the accelerometer sensor readings.

VI. EXPERIMENTAL RESULTS AND DISCUSSION

A. Experimental PID Tuning Strategy

Control parameter tuning was conducted empirically using a systematic iterative heuristic process. First, Ki and Kd were set to zero while Kp was incrementally raised until stable self-sustained balance oscillations occurred. Next, Kd was increased to introduce derivative damping and suppress overshoot. Finally, Ki was introduced to eliminate steady-state tilt error offset caused by slight mass asymmetries in the chassis assembly.

TABLE 3: Experimental PID Calibration Parameters and Response Dynamics

Parameter	Initial Trial	Final Tuned Value	Physical Control Effect
Setpoint (θ_0)	180.0°	182.0°	Compensates for asymmetric mechanical center-of-gravity offset
Kp (Proportional)	5.0	21.0	Determines initial restoring torque magnitude proportional to tilt
Ki (Integral)	50.0	140.0	Eliminates low-frequency drift and persistent steady-state offset
Kd (Derivative)	0.1	0.9	Dampens high-frequency oscillations and prevents angular overshoot

B. Performance Verification under Disturbances

The physical prototype was subjected to rigorous performance testing across multiple surface mediums and under active pulse disturbances. On smooth tiled and carpeted flat surfaces, the robot successfully maintained balance within 2 to 3 seconds from power startup.

TABLE 4: Experimental Performance Evaluation Summary

Test Scenario	System Dynamic Response	Evaluation Outcome
Static Balance (Flat Surface)	Achieved steady vertical poise within 2.0s of initialization	PASS / STABLE
Forward Push Disturbance (5cm displacement)	Immediate forward acceleration; recovered setpoint in 1.8s	PASS / RECOVERED
Backward Push Disturbance (5cm displacement)	Immediate reverse acceleration; recovered setpoint in 1.5s	PASS / RECOVERED
Inclined Surface Test (3° ramp)	Maintained equilibrium with minor steady displacement shift	PARTIAL SUCCESS
Low Voltage Operation (<7.0V Rail)	Insufficient motor torque output leading to eventual tipping	FAIL (Under-voltage)

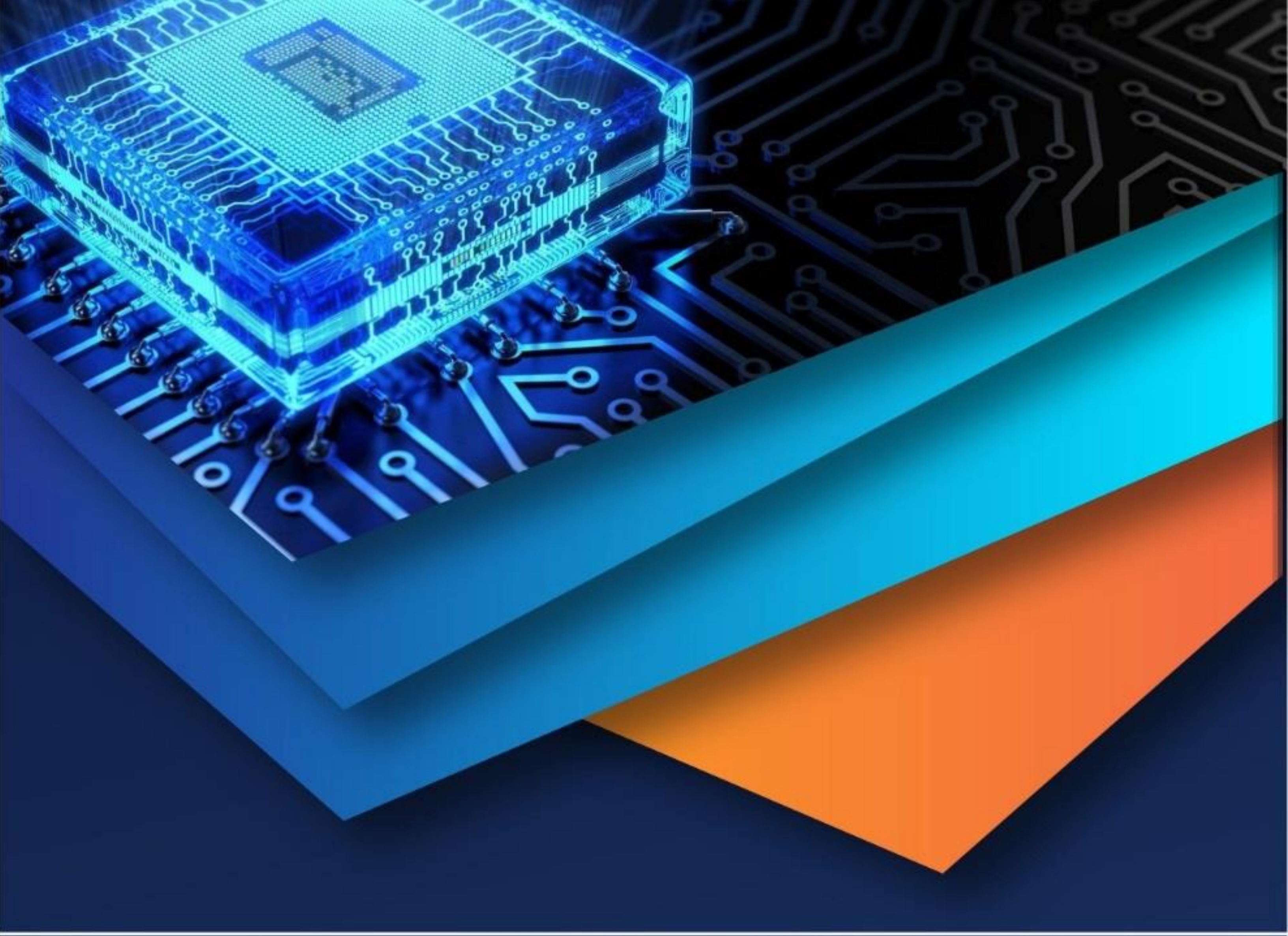
VII. CONCLUSION AND FUTURE WORK

This study presented the design, implementation, and empirical validation of an inverted pendulum self-balancing robot utilizing an Arduino Nano microcontroller, MPU6050 IMU, and PID closed-loop control algorithm. The experimental findings validate that optimal PID gain selection (Kp = 21, Ki = 140, Kd = 0.9) enables robust dynamic recovery from impulsive force perturbations.

Future work will focus on integrating wireless teleoperation (via Bluetooth HC-05 or Wi-Fi ESP8266), incorporating ultrasonic obstacle avoidance sensors, upgrading the structural chassis to 3D-printed ABS material, and exploring advanced non-linear control strategies such as Model Predictive Control (MPC) and Fuzzy Logic.

REFERENCES

[1] K. J. Åström and K. Furuta, "Swinging up a pendulum by energy control," Automatica, vol. 36, no. 2, pp. 287-295, 1996.
[2] F. Grasser, A. D'Arrigo, S. Colombi, and A. C. Rufer, "JOE: a mobile, inverted pendulum," IEEE Transactions on Industrial Electronics, vol. 49, no. 1, pp. 107-114, 2002.
[3] Arduino Project Hub, "Arduino Two Wheel Self Balancing Robot," 2025. [Online]. Available: https://projecthub.arduino.cc/mircemk/arduino-two-weel-self-balancing-robot
[4] California State University, Northridge, "Implementation of a Self-Balancing Robot," ScholarWorks, 2022. [Online]. Available: https://scholarworks.calstate.edu/downloads/08612n93k
[5] InvenSense / TDK, "MPU-6050 Product Specification Revision 3.4," TDK InvenSense, Sunnyvale, CA, 2013.
[6] B. Beauregard, "Improving the Beginner's PID," 2011. [Online]. Available: http://brettbeauregard.com/blog/2011/04/improving-the-beginners-pid-introduction/
[7] Atmel Corporation, "ATmega328P 8-bit AVR Microcontroller Datasheet," Microchip Technology Inc., 2016.
[8] STMicroelectronics, "L298N Dual Full-Bridge Driver Datasheet," Geneva, Switzerland, 2000.



INDEX COPERNICUS

TOGETHER WE REACH THE GOAL

10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY