



Design and Implementation of PID Control System for Beam Balancing using BLDC Motors

¹Roopmeet Kaur *  and ²Hardik Dhiman

¹Academy of Innovative Semiconductor and Sustainable Manufacturing, National Cheng Kung University, Tainan, Taiwan

²Department of Computer Science and Information Engineering, National Cheng Kung University, Tainan, Taiwan

¹roopmeetkhurana03@gmail.com

²hardik.dhiman29@gmail.com

Abstract: The beam balancing system is a classical example of a nonlinear and inherently unstable control problem widely used in control engineering research. This paper presents the design and implementation of a beam balancing system using a Brushless DC (BLDC) motor integrated with a Proportional–Integral–Derivative (PID) controller. The system aims to maintain the beam at a desired angle irrespective of external disturbances and varying load conditions. An MPU6050 sensor is used to measure the beam angle, and real-time feedback is processed through the Vega Aries development board to generate appropriate control signals. The PID controller dynamically adjusts the motor speed via an Electronic Speed Controller (ESC) to achieve system stability with minimal overshoot and steady-state error. Experimental results demonstrate that the proposed system effectively stabilizes the beam with improved response time and robustness. This implementation highlights the applicability of PID control in real-world unstable systems and provides a foundation for advanced control strategies. The developed prototype offers a low-cost and efficient platform for understanding practical control system implementation. The proposed setup also demonstrates reliable operation under repeated testing conditions. Performance analysis confirms that the controller is capable of reducing oscillations and maintaining desired beam position accurately. This work can further be extended by integrating intelligent control techniques such as fuzzy logic, adaptive PID, or optimization-based controllers for enhanced performance.

Keywords: Vega, BLDC Motor, MPU 6050, Proportional-Integral-Derivative (PID), ESC.

1 Introduction

The system of a balancing beam is a common feedback non-linear unstable system having a wide range of applications in electronics, aerospace, and robotics control engineering [1]. A proportional-integral-derivative (PID) controller is used in this project that uses feedback to regulate the process, where the proportional component produces an output that is proportional to the difference between the desired angle and the actual

angle [2]. The integral component produces an output that is proportional to the accumulated error over time. This helps to eliminate steady-state errors in the system. The derivative component produces an output that is proportional to the rate of change of the error, where the difference between the angle measured and the desired angle will be the error measured in the beam. This helps to reduce overshoot and improve the response time of the system. The PID controller is designed to adjust the motor's speed to maintain the balance of the beam even in harsh atmospheric conditions [3]. There are various types of PID control systems used for balancing a beam at a certain defined position. This paper presents the design and implementation of a balancing beam system with the use of the Vega Aries V3 board and BLDC motors. The system used for the beam balancing is a 2-degree-of-freedom, nonlinear, unstable system that has wide applications in engineering [4],[5]. The goal is to design a balanced beam system that can move freely with respect to the pivot point on the beam without losing momentum, regardless of the weight and angle of the beam.

2 Methodology

The balance beam system is implemented with the help of various components, including BLDC motors, Vega Aries V3 board, MPU 6050 sensor, and electronic speed controllers. The system uses brushless DC motors with flaps to balance the beam and calculate the angle using an inertial measurement unit (IMU) module. A proportional-integral-derivative (PID) controller is designed to adjust the speed of the motors to maintain the balance of the beam [6]. The system's primary objective is to achieve stable and precise balance of the beam, regardless of its weight and angle, using a dedicated PID control loop. The value that is controlled is the inclination angle of the beam, which is desired to be zero, meaning that the beam is perfectly horizontal [7]. The error is the difference between the actual angle and the desired angle of the beam ($\text{Error} = \text{Actual Angle} - \text{Desired Angle}$) [8], [9]. The electrical circuit of the system consists of an ARIES v3.0 board, a brushless DC motor, an accelerometer and a gyroscope sensor. The system is controlled using an ARIES v3.0 kit, and the code for the PID controller is developed using the Arduino IDE. The system is tested by varying the weight and angle of the beam. The system is also able to maintain the balance of the beam even when external disturbances are applied to it. Two degrees of freedom used in this system are: BLDC motor rotating on the beam and the rotation through its center of axes. This aims at achieving the desired value for the angle of a beam by controlling the speed of the BLDC motors connected at the ends of the beam. Angle measurements take place with the help of the MPU 6050 sensor. The control voltage is sent to a motor controller through a power supply of 12V which further generates a torque to drive the beam to the desired angle. With the help of this method, the problem of the instability of the beam is solved with the help of a PID controller.

Technical Approach:

- i. **Sensor Data Acquisition:** The MPU6050 sensor will continuously measure the beam's inclination angle and provide precise angle feedback to the control system.

- ii. **PID Control Loop:** The VEGA Aries 3.0 board will employ a PID control algorithm to process the angle feedback and generate corresponding control signals for the BLDC motors.
- iii. **Motor Control:** The control signals will be sent to the ESCs (Electronic Speed Controllers) connected to the BLDC motors, enabling precise control of motor speed and direction.
- iv. **Beam Stabilization:** The PID control loop will continuously adjust the motor speeds based on the angle feedback, striving to maintain the beam in a stable and balanced position.
- v. **Power Supply:** The PDB board powers the system with a 12V battery, delivering 12V to each ESC.

3 Design and Assembly

The very first step is to determine the specifications of the wooden beam, such as length, width, and material strength, and select suitable BLDC motors, propellers, ESCs, and an MPU6050 sensor module based on the system requirements. Further, we need to attach the BLDC motors with propellers to the ends of the wooden beam securely and connect the ESCs to the motors, ensuring the correct clockwise and anti-clockwise connections for balance. Mount the Aries V3 board and MPU6050 sensor at the center of the beam, ensuring stable positioning. Now, connect the ESCs to the PDB board, and wire the PDB board to the 12V battery, providing power to the system. Further, Power up the system and verify the connections and functionality of each component., Calibrate the MPU6050 sensor to ensure accurate angle measurements and verify that the ESCs are responding correctly to control signals from the Aries V3 board. Now, fine-tune the PID controller parameters for optimal performance and stability. For the software part, develop the necessary software code for the Aries V3 board to implement the PID control algorithm. Read the angle measurements from the MPU6050 sensor and calculate the error ($e(t)$) with respect to the desired angle and utilize the PID controller to generate control signals based on the error, adjusting the motor speeds accordingly. Continuously monitor the angle measurements and adjust the control signals to maintain beam balance. For Testing and Optimization, gradually increase the beam weight and vary the angle to simulate different scenarios. Observe the system's response and verify its ability to balance the beam at any point. Introduce external disturbances, such as slight pushes or vibrations, to assess the system's robustness [10]. Now, Optimize the PID controller parameters if necessary to enhance system performance and stability [11]. In order to evaluate the Performance, measure and analyze the system's accuracy in maintaining beam balance under different conditions, and evaluate the system's response time and ability to recover from disturbances. Compare the actual beam angle with the desired angle to assess the overall performance. Further, document and analyze the test results to identify any areas for further improvement.

4 System Practical Model

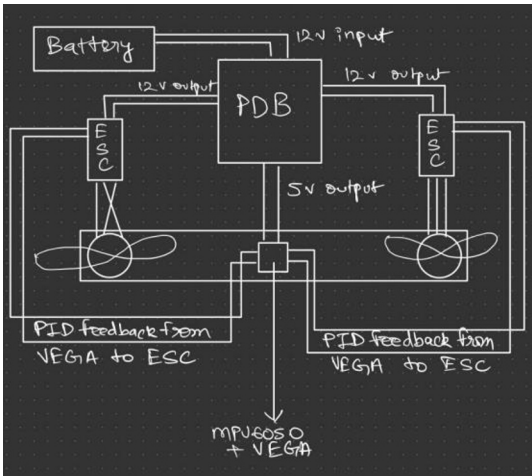


Fig. 1 Circuit Model

The circuit model and diagrammatic representation is shown in figure 1 and figure 2, where a wooden beam is taken, and at its ends, two BLDC motors are attached along with the propellers. Two ESCs are used for controlling the speed of motors. ESC has three wires, three on one side which are connected to the motor (The Right ESC, the connection made with the motor is clockwise whereas, and in the Left ESC, the connection is anti-clockwise to maintain the balance) and three on the other side, out of which the positive and negative wire is fabricated on the PDB board on the respective positive and negative terminals for both the ESCs, the other left signal wire is connected to the Aries V3 - board which is placed at the middle of the beam along with the MPU 6050 sensor. The PDB board is kept below the beam on which a 12V battery is placed. The PDB sends 12V to each ESC and 5V to the Vega-board.

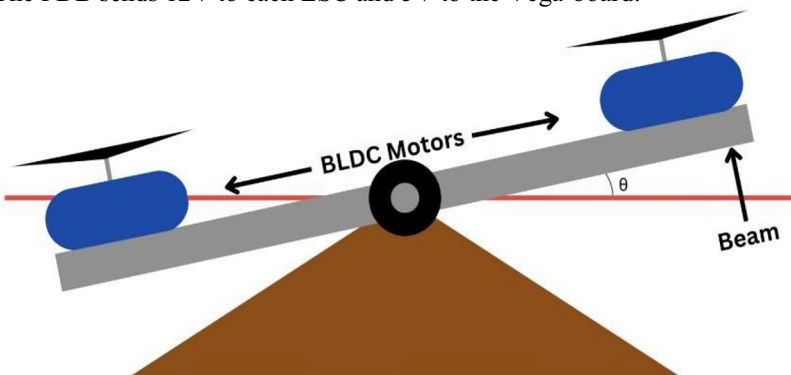


Fig. 2. Diagrammatic representation of the system

A wooden beam serves as the foundation for the balancing system, with two BLDC motors and propellers attached to its ends. To control the motor speed, two ESCs are utilized, each connected to the respective motor. The ESCs have three wires, with three connections made to the motors. For the right ESC, the motor connection is clockwise, while for the left ESC, the connection is anti-clockwise to maintain balance. On the other side of the ESCs, the positive and negative wires are connected to the PDB board, which acts as a fabrication point. The remaining signal wires are linked to the Aries V3 board for both the ESCs positioned at the beam's center, alongside the MPU 6050 sensor. Positioned beneath the beam, the PDB board powers the system with a 12V battery, delivering 12V to each ESC and 5V to the Vega board and the whole circuit model is shown in figure 3.

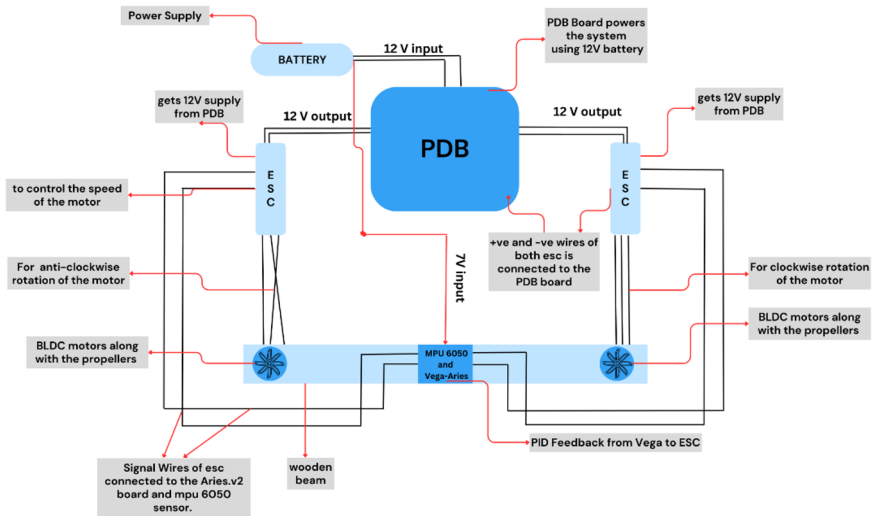


Fig. 3. Full labeled Circuit Model

5 Design and Assembly

The accelerometer is responsible for measuring the acceleration for 3 axes: X, Y and Z. The movement of IMU is upwards, the Z axis, forward and backward, the Y axis, and side to side, the X axis. The acceleration of terrestrial gravity is also detected by the IMU, having values in "g" units. In order to calculate the inclination angle with respect to the X axis or Y axis, the terrestrial gravity is used to identify the accelerometer readings. Now, further considering that IMU is perpendicular to the ground, which means the Z axis $1g = 9.8 \text{ m/s}^2$, and the two X and Y axes are 0. Now, assuming that we tilted the IMU 90 degrees such that the X-axis is perpendicular to the ground, hence marking the acceleration of $1gm/s^2$ gravity [12].

Hence, the roll angle is given by the formulae:

$$X_a = (\text{float})Ax/16384.0 - 0.04 \text{ (Accelerometer x-axis data in g unit)}$$

$Y_a = (\text{float})A_y/16384.0$ (Accelerometer y-axis data in g unit)
 $Z_a = (\text{float})A_z/16384.0 + 0.04$ (Accelerometer z-axis data in g unit)
 $\text{Roll Angle} = \tan(Y_a/\sqrt{X_a*X_a + Z_a*Z_a}) * 1/(3.142/180)$

Formulae used for calculating error and total PID value:

```

error = Roll Angle – desired_angle ;
pid_p = Kp * error ;
if (-3 < error < 3) {
pid_i = pid_i + (Ki * error); }
pid_d = Kd*(( error – previous_error) / elapsedTime) ;
PID = pid_p + pid_i + pid_d ;

```

Limiting PID values to avoid oversteer:

```

if (PID < -64) {
PID = -64 }
if (PID > 64) {
PID = 64 }

```

For Calculating PWM signals of both motors and putting a limit if values exceeds max:

```

pwmLeft = throttle + PID;
pwmRight = throttle – PID;
if (pwmLeft < 0) {
pwmLeft = 0; }
if (pwmLeft > 128) {
pwmLeft = 128; }
if (pwmRight < 0){
pwmRight = 0; }
if (pwmRight > 128) {
pwmRight = 128; }

```

The tuned PID values are:

```

double Kp = 0.500;
double Ki = 0.020;
double Kd = 0.001;

```

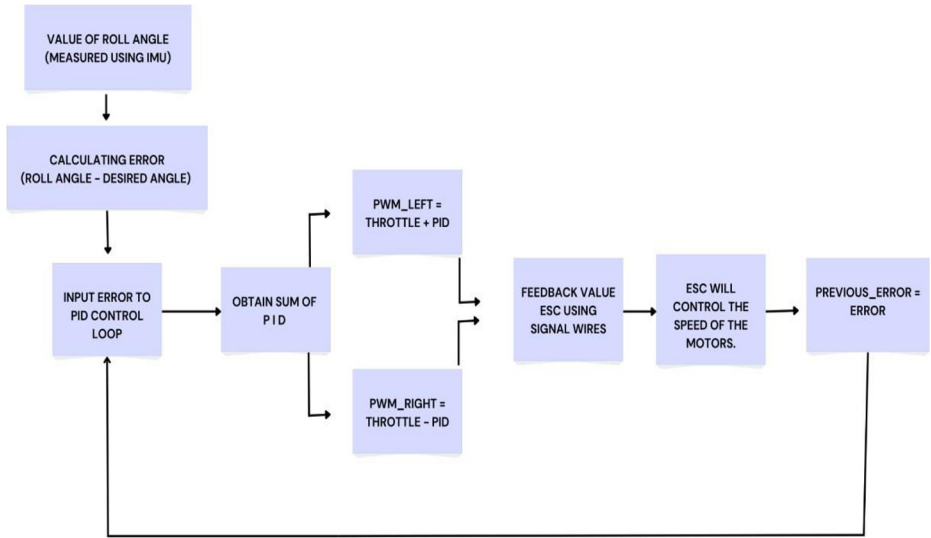


Fig. 4. Flowchart indicating the implementation of PID using roll data from IMU

6 Results

This paper presents a novel investigation into the performance difference between utilizing the Vega Aries board and a competitor board like Arduino UNO for implementing a complex task, specifically the design and handling of a PID Control System for a beam balancing application. The final working model is shown in figure 5 and a modified version of this project can also be used to tune PID for all three axes of a drone. This can be again made using a larger stand with a cage-like structure that would remove the need to fly it again and again in the air, and also make it easier to access and tune the PID values.

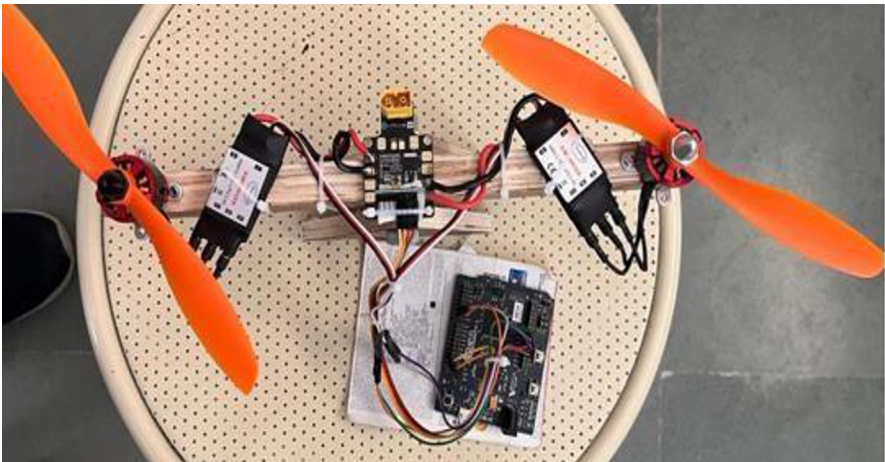


Fig. 5. Final Working Model

7 Conclusion

This paper presented the successful design and implementation of a beam balancing system using a BLDC motor controlled by a PID algorithm. The system effectively demonstrates the ability to stabilize a nonlinear and inherently unstable setup using feedback control mechanisms. By integrating the MPU6050 sensor for accurate angle measurement and the Vega Aries board for real-time processing, the system achieves reliable performance under varying conditions. The PID controller proved effective in minimizing steady-state error, reducing overshoot, and improving response time. Although the system performs well, further improvements can be made by implementing advanced control strategies such as fuzzy logic or adaptive control to enhance robustness and adaptability. Overall, this work validates the effectiveness of PID-based control in practical engineering applications and serves as a foundation for future research in intelligent control systems.

References

- [1] S. Kumar Krishnamoorthy, N. Das, P. Gudimetla and K. Emami, "Enhanced Speed Control for BLDC Motors Using WOA-Integrated PID Controller Optimization," in *IEEE Access*, vol. 12, pp. 162465-162475, 2024, doi: 10.1109/ACCESS.2024.3480349.
- [2] M. H. Mahmood, O. F. Ali, M. J. Mnati, B. M. Sabbar, and A. Van Den Bossche, "Intelligent PID parameter tuning for BLDC motors by using genetic algorithms," *Journal Européen des Systèmes Automatisés*, vol. 58, no. 7, pp. 1463–1476, Jul. 2025, doi: 10.18280/jesa.580714.
- [3] M. Mahmud, S. M. A. Motakabber, A. H. M. Z. Alam, A. N. Nordin, and A. K. M. A. Habib, "Modeling and performance analysis of an adaptive PID speed controller for the BLDC motor," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 11, no. 7, 2020.
- [4] S. Usha, J. Mani Dubey, R. R. Rayar, and M. V. Suganyadevi, "Performance enhancement of BLDC motor using PID controller," *International Journal of Power Electronics and Drive Systems (IJPEDS)*, vol. 12, no. 3, pp. 1335–1344, Sep. 2021, doi: 10.11591/ijpeds.v12.i3.pp1335-1344.

- [5] A. Prakash, D. V. Bhaskar, A. Ranjan, P. Kumar and U. R. Muduli, "Advanced Control and Modeling for Enhanced BLDC Motor Efficiency and Performance," *2024 IEEE International Conference on Power Electronics, Drives and Energy Systems (PEDES)*, Mangalore, India, 2024, pp. 1-6, doi: 10.1109/PEDES61459.2024.10961315.
- [6] R. Kristiyono and W. Wiyono, "Autotuning Fuzzy PID Controller for Speed Control of BLDC Motor", *J Robot Control (JRC)*, vol. 2, no. 5, pp. 400–407, Sep. 2021.
- [7] D. Joshi, D. Deb and A. K. Giri, "Metaheuristic Adaptive Input Output Feedback Linearization Control for BLDC Motor Drive," in *IEEE Transactions on Consumer Electronics*, vol. 71, no. 2, pp. 6120-6130, May 2025, doi: 10.1109/TCE.2025.3553388.
- [8] F. G. Borges, M. Guerreiro, P. E. S. Monteiro, F. C. Janzen, F. C. Corrêa, S. L. Stevan Jr., H. V. Siqueira, and M. dos S. Kaster, "Metaheuristics-Based Optimization of a Robust GAPID Adaptive Control Applied to a DC Motor-Driven Rotating Beam with Variable Load," *Sensors*, vol. 22, no. 16, p. 6094, 2022, doi: 10.3390/s22166094.
- [9] A. Hasni, H. El Fadil, A. Lassioui, M. El Ancary, Y. El Asri and M. Koundi, "Enhanced Speed Regulation of BLDC Motors via Dual PID Control and Cascaded Buck-Boost Converter," *2025 International Conference on Electrical Systems & Automation (ICESA)*, Troyes, France, 2025, pp. 1-6, doi: 10.1109/ICESA66763.2025.11281228.
- [10] S. Gupta, K. Ray, and S. Kaiser, "Self-Balancing Mobile Robot with Bluetooth Control: Design, Implementation, and Performance Analysis," *Automation*, vol. 6, no. 3, p. 42, 2025, doi: 10.3390/automation6030042.
- [11] Z. Jiang, H. Xue, H. Yue, X. Bao, J. Zhu, X. Wang, and L. Zhang, "A Review of Artificial Intelligence-Driven Active Vibration and Noise Control," *Machines*, vol. 13, no. 10, p. 946, 2025, doi: 10.3390/machines13100946.

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (<http://creativecommons.org/licenses/by-nc/4.0/>), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

