# ME351: Lab Report

## Experiment 3: Balancing of Ball on a Beam System

**Group 6** | Week 6


Harshil Safi | 20110073

Hrishikesh CP | 20110078

Kamal Vaishnav | 20110089

Kareena Beniwal | 20110095


*Under the Guidance of*
*Prof. Jayaprakash KR*

## 1. Objective

The objective of this experiment is to design and implement a PID control algorithm using an Arduino and motor controller, to stabilize a ball at a desired position on a beam, with the help of an ultrasonic sensor. The expected learning outcome is to gain an understanding of the principles of PID control, and to develop practical skills in designing and implementing control algorithms using physical components such as motors and sensors. The experiment will involve building a physical model of a beam, hinged at its center and controlled by a DC motor, using an Orange 12V 200 RPM Johnson Geared DC Motor and a 13A DC motor controller. The beam will be stabilized using an ultrasonic sensor(HC-SR04) and an optocoupler isolation module (TLP281 4-channel). The power supply used will be a 12V, 5A power supply, and the microcontroller used will be an Arduino Uno R3.

## 2. Experimental design and Fabrication details

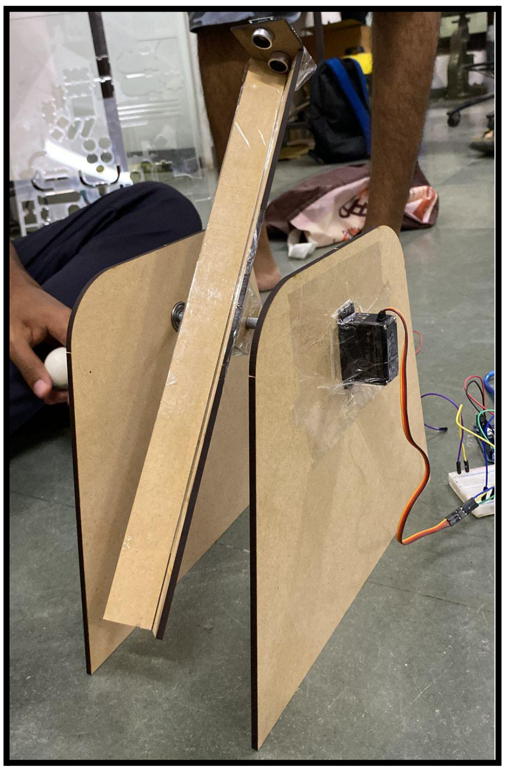
The PID controller is a feedback control algorithm that continuously measures the error between the desired setpoint and the actual process variable and applies a control signal to correct the error. The PID controller uses three terms to calculate the control signal: proportional (P), integral (I), and derivative (D) terms.

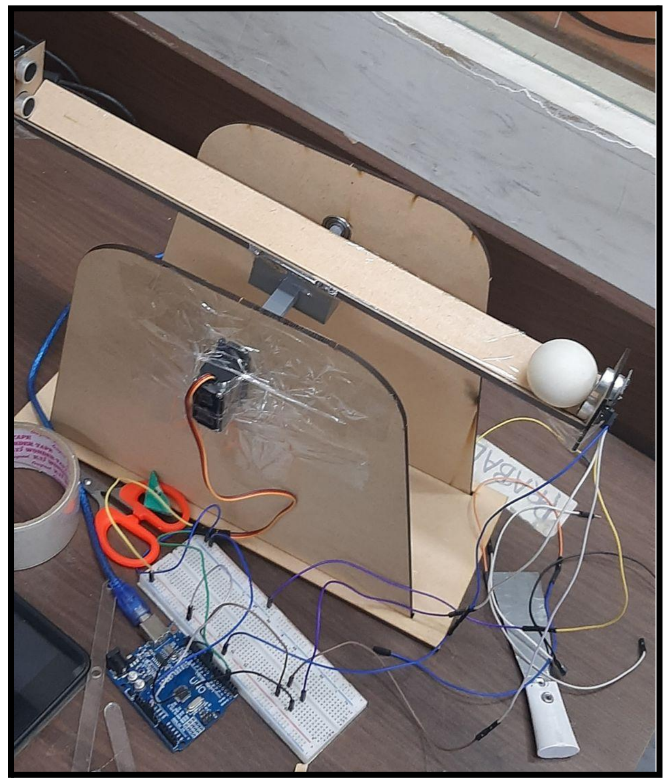
The P term responds to the current error between the setpoint and the actual process variable. The I term integrates the error over time to handle steady-state errors, while the D term responds to the rate of change of the error to handle overshoot and undershoot.

The performance of a PID controller can be evaluated based on several criteria such as steady-state error, rise time, settling time, overshoot, and stability. These criteria are typically specified in the control requirements and can be used to tune the PID controller parameters to meet the desired performance.

In balancing a ball on a beam, the PID controller can be used to maintain the position of the ball at a desired location by continuously measuring the position of the ball and applying a control signal to the motor to adjust the beam's angle. The ultrasonic or IR sensor can be used to measure the position of the ball, and the motor controller can be used to control the motor's speed and direction.

The model mainly consists of parts laser cut from MDF sheets for the required dimensions. The path for the ball consists of strips of mdf sheet. The strips are arranged at right angles and sensors are attached to both ends.





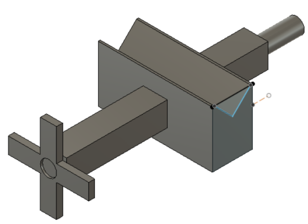


Fig: Cad model of the shaft

A cylindrical design was not done for the shaft as constant jects provided was causing the shaft to break.

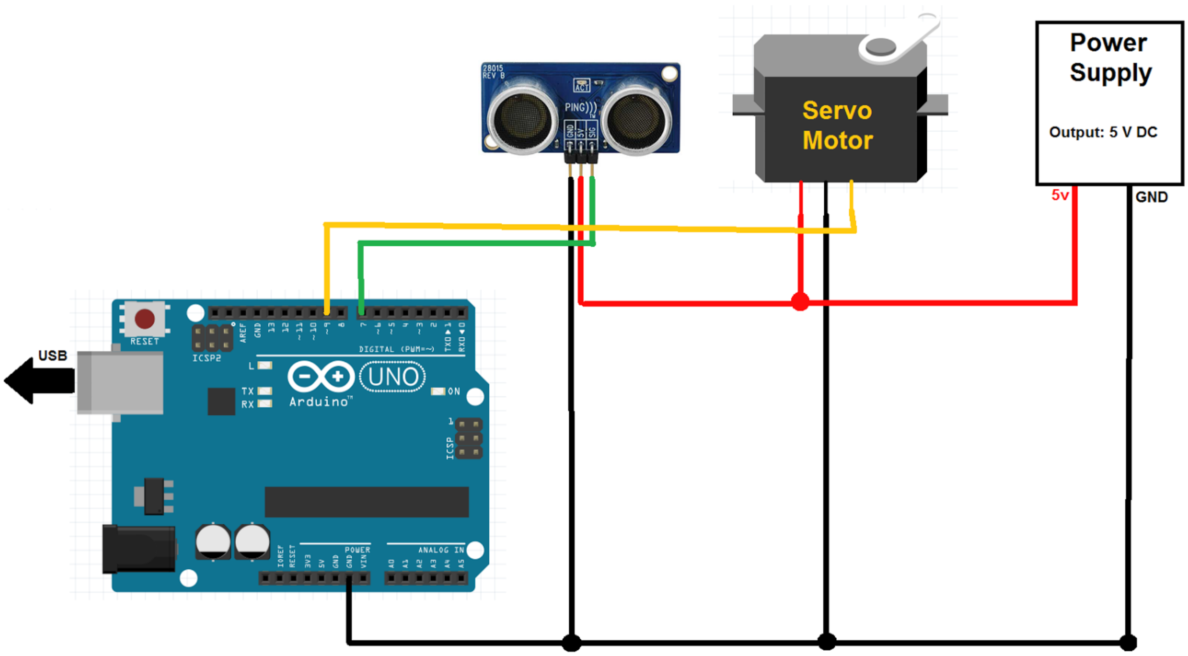



Fig: Connections

## *Arduino Code for the experiment:*

```
#include <Servo.h>
#include <Ultrasonic.h>


// Define constants
const int setpoint = 15; // desired distance for the ball from the sensor
const int servo_center =90; // servo center position
const double kp = 0.5; // proportional gain
const double ki = 0.00001; // integral gain
const double kd = 100; // derivative gain
int range=30;
```

```
// Define variables
int distance = 0;


double error, last_error, total_error, P, I, D, pid_value,tn,tp,deltaT;
float pos;

// Create objects
Servo myservo;
Ultrasonic ultrasonic1(9, 10);

void setup() {
  myservo.attach(3); // attach servo to pin 9
  myservo.write(servo_center);
  delay(10);// set initial position to the center
  Serial.begin(9600);
  tp=micros();

}


void loop() {

  float l=ultrasonic1.read();

// Serial.print(l);
  //Serial.print("  ");
 //Serial.print(r);


  distance=l;


  if((distance<40)&&(distance>0)){
  error = setpoint -distance;
  tn=micros();


  deltaT=(tn-tp)/1000;

  tp=tn;
  P = kp * error; // calculate proportional term
  total_error += (error*deltaT); // update integral term
  I = ki * total_error; // calculate integral term
  D = kd * ((error - last_error)/deltaT); // calculate derivative term
  last_error = error; // update last error
  pid_value = P + I + D;
```

```
 Serial.println(distance);
 if(abs(error)>=3)
 pos=servo_center-pid_value;

 if(pos>servo_center+range)
 pos=servo_center+range;
 else if(pos<servo_center-range)
 pos=servo_center-range;

 //Serial.println(distance);

  myservo.write((int)pos); // move servo towards minimum position


  delay(10);
 }
}
```

## *Explanation of the above code:*

The above Arduino code is written for a Ball Balancing experiment that uses a beam to balance a ball. The objective of the experiment is to control the position of the ball on the beam using a servo motor based on the feedback from an Ultrasonic sensor.

The program first defines some constants, such as the desired setpoint (the desired distance of the ball from the sensor), the proportional, integral, and derivative gains, and the servo center position.

The program then creates objects for the Servo motor and the Ultrasonic sensor and initializes them in the setup() function. The initial position of the servo motor is set to the center, and the serial communication is started.

In the loop() function, the program reads the distance from the Ultrasonic sensor and calculates the error between the setpoint and the current distance. The program then calculates the P (proportional), I (integral), and D (derivative) terms of the PID controller using the error and the gains. The PID output is calculated as the sum of these terms.

The program then checks whether the absolute error is greater than or equal to three. If it is, the program calculates the new position of the servo motor based on the PID output. The position is limited to a range of +/-30 degrees from the center position to prevent the servo from over-rotating.

From the code also we can infer that we have put a minimum error of 3 cm in position for the PID control to activate and hence the system may have an absolute error less than 3cm.

## 3. Mathematical and Theoretical Analysis:

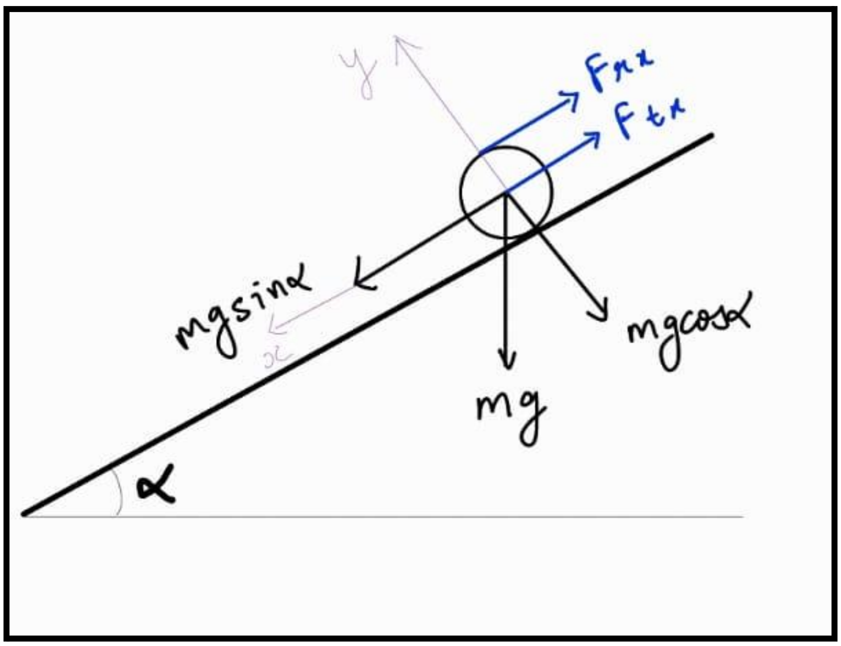



Fig: FBD of the ball

The parameters of the ball and beam are defined as follows:

α: Beam angle coordinate

L: Beam Length

m: Mass of the ball

R: Radius of the ball

J : Ball's moment of inertia

G : Gravitational acceleration

x : Position of the ball

Neglecting frictional forces, the two forces influencing the motion of the ball are:
- $F_{tx}$ Force due to translational motion
- $F_{rx}$ Force due to ball rotation

Translational equations:

$$\ddot{x} = \frac{d^2 x}{dt^2} \qquad F_{tx} = m \cdot \ddot{x}$$

Torque due to the ball rotation is:

$$T_r = F_{rx} \cdot R = J \cdot \frac{dw_b}{dt} = J \cdot \frac{d\left(\frac{v_b}{R}\right)}{dt} = J \cdot \frac{d^2\left(\frac{x}{R}\right)}{dt^2} = \frac{J}{R} \cdot \ddot{x}$$

$$F_{rx} = \frac{J}{R^2} \cdot \ddot{x}$$

The moment inertia of the ball is (sphere):

$$J = \frac{2}{5} \cdot m \cdot R^2$$

$$F_{rx} = \frac{2}{5} \cdot m \cdot \ddot{x}$$

Newton's second law along the inclination:

$$F_{rx} + F_{tx} = m \cdot g \cdot \sin\alpha$$

$$\frac{2}{5} \cdot m \cdot \ddot{x} + m \cdot \ddot{x} = m \cdot g \cdot \sin\alpha$$

Finally we have:

$$\ddot{x} = \frac{5}{7} \cdot g \cdot \sin\alpha$$

## 4. Results and discussions

Through trial and error we were able to balance the ball consistently,with the PID values as follows

$K_p$=0.5

$K_d$=100

$K_i$=0.00001(close to zero)

We were able to balance the ball at a distance of 15 cm from one end of a 40cm long path position profile of the ball.

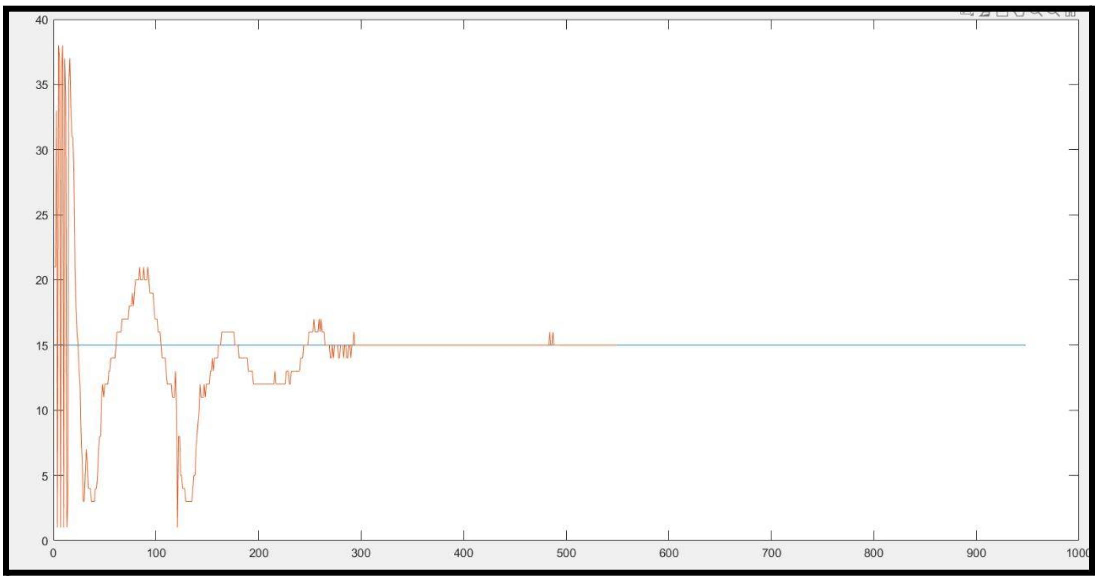



Fig: Position of ball vs Time

Based on the results obtained from the experiment, it can be concluded that the PID control algorithm implemented using the Arduino and motor controller successfully stabilized the ball at the desired position on the beam. The ultrasonic sensor provided accurate and reliable data to the control algorithm, allowing it to adjust the motor speed to maintain the ball's position.

The PID control algorithm demonstrated the ability to quickly respond to changes in the ball's position and maintain stability with minimal overshoot and settling time. This indicates that the algorithm can be used effectively in various real-world applications where precise control of a system is required.

Overall, the experiment was successful in achieving the objective of designing and implementing a PID control algorithm using an Arduino and motor controller to stabilize a ball at a desired position on a beam, with the help of an ultrasonic sensor. Further improvements could be made to the control algorithm to increase its robustness and performance in more complex systems.

## 5. Source of discrepancy/mismatch, if any

There can be several possible discrepancies in the experiment, such as:

***Sensor Accuracy:*** The accuracy of the ultrasonic sensor or Sharp IR sensor used to detect the position of the ball on the beam could be a potential source of error. These sensors have limitations in their measurement range and resolution, which could lead to inaccuracies in the control algorithm.

***Sensor Calibration:*** The ultrasonic sensor or Sharp IR sensor used to detect the position of the ball on the beam must be calibrated properly to provide accurate measurements. Any inaccuracies in the calibration process can result in discrepancies in the experimental results.

***Motor Characteristics:*** The characteristics of the DC motor used, such as its torque-speed curve, efficiency, and power rating, may not be accurately known or may not match the specifications provided by the manufacturer. This can result in discrepancies in the experimental results and may require adjustments to the control algorithm.

***Mechanical System Dynamics:*** The physical system used in the experiment, including the beam, the ball, and the motor, may have dynamics that are difficult to model accurately. This can result in discrepancies in the experimental results and may require adjustments to the control algorithm.

***Controller Tuning:*** The PID controller used in the experiment must be tuned properly to ensure stable and accurate control. Improper tuning of the controller can result in overshoot, oscillation, or instability in the system.

***Noise:*** Environmental factors such as electrical noise, mechanical vibrations, or air currents can also affect the accuracy of the measurements and control, leading to discrepancies in the experimental results.

## 6. Scope for improvement

There are several areas where improvements can be made in the experiment, such as:

***Sensor Selection:*** The accuracy and resolution of the sensors used in the experiment can be improved by selecting sensors with higher precision and accuracy. For example, using a higher-end laser displacement sensor instead of an ultrasonic or Sharp IR sensor can improve the measurement accuracy.

***Sensor Calibration:*** To ensure accurate measurements, the ultrasonic or Sharp IR sensor should be calibrated precisely. Calibration should be performed regularly to ensure accurate measurements.

***Motor Selection:*** Using a higher-quality DC motor with a better torque-speed curve and power rating can improve the accuracy of the experimental results. A motor with better efficiency can also lead to energy savings.

***Mechanical System Design:*** The design of the mechanical system can be improved by using high-precision manufacturing techniques and better materials to ensure consistent performance. The beam length can be increased beyond 1m, which can help in better testing and experimentation.

***Controller Design:*** The PID controller can be designed using more advanced techniques such as model-based control or adaptive control, which can lead to more accurate and stable control. The controller can also be implemented using a digital signal processor (DSP) instead of an Arduino for better performance.

***Power Supply:*** Using a higher-quality and more stable power supply can improve the stability of the system and prevent fluctuations in voltage or current.

***Noise Reduction:*** Measures such as shielding the sensors and motor, or using filters to remove electrical noise, can reduce the impact of external noise on the experimental results.

## 7. Acknowledgements

We would like to express our sincerest gratitude to Mr. Jayprakash KR, Assistant Professor of Mechanical Engineering at IIT Gandhinagar, for his invaluable guidance and support throughout the completion of our experiment on 'Balancing of Ball on a Beam System'. We also extend our appreciation to our TAs NagaVishnu, Vaibhav Tandel, Harsh Gupta, and Ishan for their dedicated support and motivation.

## 8. References

- https://www.youtube.com/watch?v=VGTy1MdWkhk
- https://ctms.engin.umich.edu/CTMS/index.php?example=BallBeam&section=SystemModeling