# **Ball and Beam Control System Project**

## **Overview**

You are an engineer working for a cutting-edge robotics company. Your team has been tasked with developing a precise control system for a new generation of robotic arms. To demonstrate the principles involved, you'll create a ball and beam control system as a proof of concept. This system will showcase your ability to balance an object (the ball) at any desired position along a beam, simulating the precision required for robotic arm movements.

### **Learning Objectives**

* Understand and apply principles of control systems
* Develop mathematical models for physical systems
* Implement PID control algorithms
* Simulate dynamic systems using software tools
* Design and build a physical control system
* Integrate hardware and software components
* Test and optimize system performance

## **Project Physics and Differential Equations**

### **Key Physics Concepts**

* Newton's laws of motion
* Torque and moment of inertia
* Friction and rolling resistance

### **Relevant Equations**

* [List of relevant physics equations]

### **External Resources**

* [List of recommended textbooks, online courses, or tutorials]

### **Deriving the Differential Equation**

* Guide students through the process of deriving the differential equation for the ball and beam system
* Provide hints and checkpoints for verification

### **State Space Representation and Transfer Function**

* Instructions for deriving the state space representation
* Steps to obtain the transfer function from the differential equation

## **Simulation Software Selection**

For this project, we will use MATLAB for simulation.

### **MATLAB Simulation Steps**

1. Create a new MATLAB script
2. Define system parameters
3. Implement the state space model
4. Create a function for the PID controller
5. Set up the simulation loop
6. Implement real-time plotting
7. Add user interface for PID parameter input
8. Display system performance metrics (overshoot, settling time, etc.)

## **Physical System Considerations**

### **Fixed Components**

* Beam material: Aluminum
* Beam stands: 3D printed brackets
* Hinges: Ball bearings
* Platform: Wooden base
* Microcontroller: Arduino Uno
* Testing materials: Steel ball bearings of various sizes
* Power supply: 9V battery
* Circuit board: Solderless breadboard

### **Design Decisions**

Students must choose from the following options:

1. Actuator type:
   * Servo motor
   * Stepper motor
   * DC motor with encoder
2. Position sensor:
   * Digital ultrasonic sensor
   * Analog infrared sensor
   * Proprietary time-of-flight sensor
3. Circuit components:
   * List of recommended active and passive components
4. Communication protocol:
   * I2C
   * SPI
   * UART

## **System Build**

1. Assemble the mechanical components
2. Set up the breadboard circuit
3. Connect the microcontroller, actuator, and sensor
4. Prepare the power supply

## **Code Design**

### **PID Library**

* Provide function headers for PID implementation
* Students must complete the implementation

### **Motor Control**

* Option to use existing library or create a custom one
* Provide resources for both approaches

### **Sensor Interface**

* Option to use existing library or create a custom one
* Provide resources for both approaches

### **Main Control Loop**

* Provide a template with key function calls
* Students must implement the core logic

## **Integration and Testing**

1. Upload code to the microcontroller
2. Perform initial system tests
3. Tune PID parameters
4. Conduct performance tests:
   * Response time
   * Stability at various positions
   * Overshoot measurements
5. Record video demonstration
6. Capture screen recording of system response

## **Documentation and Submission**

1. Compile results and analysis
2. Update project repository on GitHub
3. Create project documentation:
   * System design overview
   * Mathematical model and derivations
   * Simulation results and analysis
   * Hardware implementation details
   * Software architecture and key algorithms
   * Test results and performance metrics
   * Challenges encountered and solutions
   * Future improvements and applications
4. Submit all required materials as specified in the submission instructions

## **Command Line Options**

[Not applicable for this project]

## **Input File Formats**

[Not applicable for this project]

## **Output File Format**

[Not applicable for this project]

## **Test Files**

[Not applicable for this project]

## **Error Checking**

* Implement error handling for sensor readings
* Add input validation for user-entered PID parameters
* Include fail safes for motor control to prevent damage

## **Assumptions**

* The beam is perfectly straight and uniform
* The ball is perfectly spherical with uniform density
* Air resistance is negligible

## **Libraries and Restrictions**

* Students may use standard Arduino libraries
* Use of third-party PID or motor control libraries is not allowed unless specified

## **Coding Style Guidelines**

* Use meaningful variable and function names
* Include comments explaining non-obvious code sections
* Follow consistent indentation and formatting
* Avoid global variables where possible
* Implement modular code structure with clear separation of concerns