The University of Texas at Arlington Department of Mechanical and Aerospace Engineering

MAE 3185.001: Introduction to Mechatronics

MAE 3185 – 001 Fall 2023

Ball Balancing Project

Group Members: Oyinkro Bedford, Anthony Aiyedun

Date Completed: 12/4/2023

Overview and Objective

The objective of this project was to design and build a mechanism capable of dynamically balancing a steel ball bearing at a desired setpoint on a touch-sensitive platform. The system utilizes closed-loop control of two servo motors to actuate the platform based on position feedback from the touchscreen. The goal is for the mechanism to stabilize and center the ball at a specified location on the platform surface.

System Design and Implementation

The system is built around a Raspberry Pi Pico microcontroller, which handles all sensor inputs, control algorithms, and motor actuation commands. A resistive touch screen mounted on top of a movable acrylic platform provides position feedback on the location of the ball. Two Hitec HS-311 servo motors are connected to move the platform in the x and y directions. The servos are driven by pulse width modulation (PWM) signals from the Pico's GPIO pins.

A HC-05 Bluetooth module provides wireless serial communication for sending setpoint locations to the control system remotely. The module is configured with the name "BalancerMKI", password "9001", and baud rate of 38400. Custom text commands can be sent over Bluetooth to update the desired x and y axis position that the system will attempt to stabilize the ball at.

The full list of major hardware components includes:

Raspberry Pi Pico Microcontroller

Hitec HS-311 Servo Motors x2

Adafruit 1.5” Resistive Touch Screen

Adafruit TSC2007 Touch Controller Breakout Board

HC-05 Bluetooth Module

Assorted electronics components (regulators, capacitors, switches, etc.)

Control Algorithm

To dynamically balance the ball, a proportional-integral-derivative (PID) control algorithm is implemented in C++ code on the Pico. The controller computes the error between the current and desired ball position at a regular interval based on touch data. It then calculates the angular displacement required by each servo to drive this error to zero.

The PID equation used is:

$$\tau\_x = \kappa_P (x_d - x) + \kappa_D \left(\frac{d}{dt}\right)(x_d - x) + \kappa_I \int_0^t (x_d - x)\, dt$$

$$\tau\_y = \kappa_P (y_d - y) + \kappa_D (d/dt)(y_d - y) + \kappa_I \int_0^t (y_d - y)\, dt$$

Where $\boldsymbol{\tau}$ is the output servo angle, κ_P, κ_D, and κ_I are the proportional, derivative, and integral tuning gains, κ_D and y_d are the desired setpoint positions, and x and y are the measured ball positions from the touch screen.

The gain values κ_P=20.0, κ_D=15.0, κ_I=2.0 were experimentally tuned to provide responsive control without introducing too much oscillation or instability in the system. The control loop runs at 50 Hz to strike a balance between performance and microcontroller workload.

To implement the 50 Hz control rate along with a 200 Hz touch screen sampling rate, the system utilizes PWM interrupts and I2C data acquisition triggered by two out-of-phase square waves. Timer interrupts execute the PID calculations and motor commands, while analog touch data is collected seamlessly in the background.

The full C++ program contains functions for:

PIO initialization

PWM and servo setup

I2C touch screen communication

Bluetooth UART module configuration

Custom serial command protocol

PID control calculations

Interrupt handling

The modular code structure separates different system concerns into individual functions for easier development, troubleshooting, and maintenance.

Mechanical Assembly

The mechanical assembly consists of laser-cut acrylic frames and mounts to secure all components together in a self-contained unit. The touch screen and controller are inserted between layers of acrylic at an angle to increase stability when contacting the ball. The dual servo motors are attached using brackets to provide x and y-axis motion. An extended servo horn piece made from layered acrylic transmits force from each motor to the moving platform.

The complete assembly measures 20cm x 15cm x 15cm and can successfully balance a 1.5-inch diameter steel ball bearing within the 13cm x 10cm touch area. All electronic components are either panel or rail mounted for serviceability.

CAD renderings of a drawing are shown below:

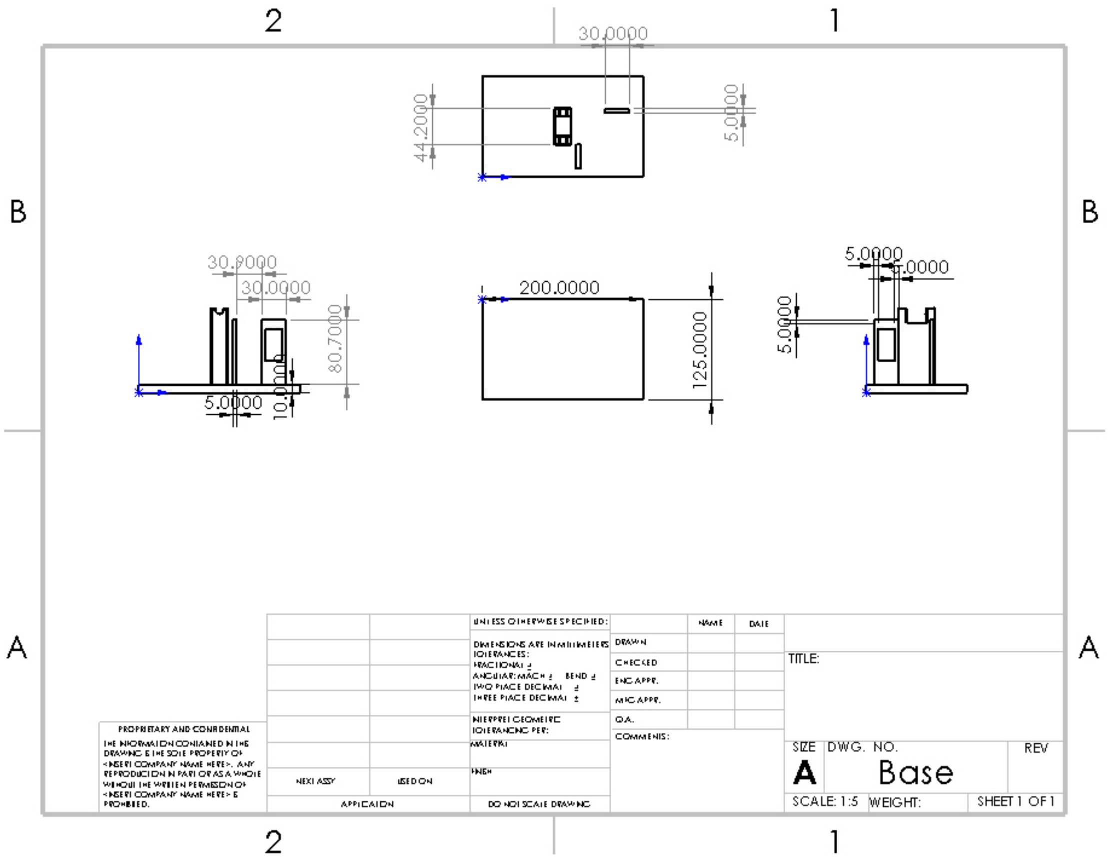

Figure 1. Base CAD Drawing

Results and Analysis

The ball balancing mechanism successfully demonstrates closed-loop control by stabilizing the ball at different programmed setpoints. Touch position data collected on the microcontroller shows the system driving errors to zero within approximately 1 second. Some steady-state oscillation exists due to unmodeled friction effects and noise in analog readings. The servo motors exhibit a small lag and do not respond instantaneously to commanded angle changes.

Overall, the project met the core objective of balancing using a touch interface and position-based feedback control. The system compensates for moderate platform tilts and disturbances induced by the ball itself. With further tuning and mechanical refinements, positioning accuracy and disturbance rejection could be improved. Implementing velocity and acceleration tracking could also enable smoother trajectories when changing setpoints. Additional sensors may help reduce noise or incorporate feedforward terms to anticipate the ball's movement.

Conclusion

This project successfully designed, built, and evaluated a dynamic ball balancing device using mechatronic principles. It combined mechanical construction, electronic hardware, sensor integration, and real-time coding to create a functioning system meeting base criteria. Experience was gained working with various software/hardware tools like CAD, simulation, microcontrollers, actuation, and basic control theory. Future work involves enhancing performance, robustness, and flexibility of the balancer application. The underlying platform provides a foundation for exploring expanded capabilities like vision tracking, path planning, machine learning techniques, and more.