**Tuesdays 1:00PM or 3:00PM -** Weekly Meeting

To do:

* Rationale where necessary(Simulation/Analysis if we have any)
* Performance Requirements

**System Requirements (Dhruv, Auda and Gurkarmjit)**

* Table of revisions
* Purpose - IN PROGRESS
* Scope - IN PROGRESS
* Operation - IN PROGRESS
  + normal scenario - IN PROGRESS
  + negative scenario - nah???????
* system diagram - ??
* context diagram - DONE
* system level variables - ??
  + monitor and control variables - DONE
* **Requirements (Joseph and Ahmed)**
  + functional and nonfunctional
  + functional diagram

On the agenda:

-dividing work for requirements - DONE

-figure out implementation/design of the project

Project Plan - Gantt chart and Activities that need to be performed over the course of the project (Dhruv and Sodie)

CAD Diagrams (Sodie feat. Auda and Joseph)

-determine parts to purchase - Canada

Servos

Surface - Metal/Acrylic or something else like wood

3D print legs

Arduino small or big boy

Servos (10pack):

<https://www.amazon.ca/Kuman-Micro-Helicopter-Airplane-Controls/dp/B01N997RVE/ref=sr_1_12?ie=UTF8&qid=1508269392&sr=8-12&keywords=servo>

Raspberry Pi camera: <https://www.amazon.ca/Raspberry-Pi-Camera-Module-Megapixel/dp/B01ER2SKFS/ref=sr_1_3?s=electronics&ie=UTF8&qid=1508268324&sr=1-3&keywords=raspberry+pi+camera>

Camera cable extension (61cm): <https://www.amazon.ca/Flex-Cable-Raspberry-Pi-Camera/dp/B00M4DAQH8/ref=sr_1_2?ie=UTF8&qid=1508267854&sr=8-2&keywords=raspberry+pi+flex+cable>

-figure out timelines and when man's are all free - DONE



MECHTRON 4TB6 -

System Requirements

Dhruv Aggarwal

Gurkarmjit Kooner

Joseph Moolaseril

Auda Rab

Ahmed Belal

# 

# 

# **Purpose**

The purpose of the project is to build an autonomous system which will interact with a ball on a platform. The system will be capable of adjusting the platform to balance the small ball in the center of the platform, and will also be capable of moving the platform up and down to bounce the ball vertically. This project will involve the implementation of control systems and object tracking.

# **Scope**

This project is based on a popular activity of ping pong bouncing and balancing. The system will be able to perform these standards activities at a skill level of an average human which was validated against each member of the project development team. This system is a stand-alone product with multiple modes of operation. THIS IS SHIT!!! I agree

The intent of this project is to have a development team of 5 Mechatronics engineering students apply PID theory to autonomously balance and bounce a ping pong ball on a platform. The system will be able to track a ball and react to its trajectory in real-time. Modes of operation will be selected by the user and the system will perform according the the mode selected.

In scope

1. HMI to display information mode switching, current operation mode and timer’s report.
2. Centring the ball via the two modes of operation which are balancing and bouncing.
3. Tracking the ball during execution.
4. The introduction of the ball to the system will automatically put the system in run state.
5. Self-adjustment of the platform to keep the ball within the environment.
6. Meeting the defined height for the system in bounce mode.
7. System will be shutdown following a user input.
8. When the ball is removed, the platform will return to home position.
9. The system will be self contained.

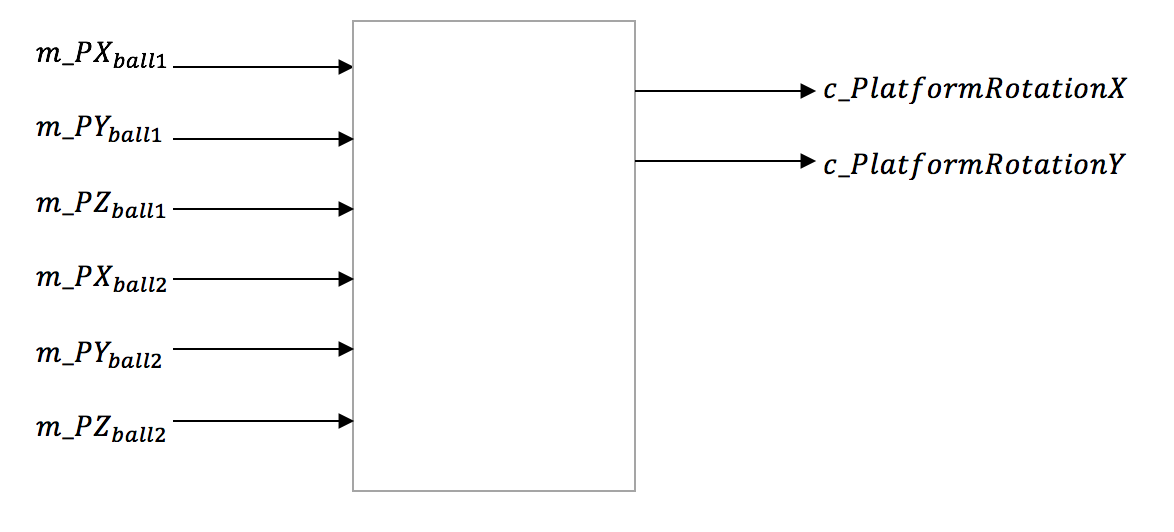
Out of scope

1. The system will auto-shutdown.
2. Allowing user to select a height the ball will be bounced at.
3. Adjusting cameras to different heights

# **Standard Notation**

|  |  |
| --- | --- |
| Notation | Description |
| Ball | The ball |
| Platform | The platform |
| Bounce | The bouncing Mode of operation |
| Balance | The balancing mode of operation |
| Adjust | The movement of the platform |
| Environment | The volume of space that the ball can be tracked |

# **Context Diagram**



|  |  |  |
| --- | --- | --- |
| Variable | Unit | Description |
|  | cm | Previous position of the ball in the x-axis |
|  | cm | Previous position of the ball in the y-axis |
|  | cm | Previous position of the ball in the z-axis |
|  | cm | Current position of the ball in the x-axis |
|  | cm | Current position of the ball in the y-axis |
|  | cm | Current position of the ball in the z-axis |

|  |  |  |
| --- | --- | --- |
| Variable | Unit | Description |
|  | cm | Angle of the platform tilt in the x axis |
|  | cm | Angle of the platform tilt in the y axis |

|  |  |  |
| --- | --- | --- |
| Variable | Unit | Description |
| k\_MaxBallHeight | cm | The maximum height the ball will bounce to |
| k\_PlatformLength | cm | The length of the platform |
| k\_PlatformWidth | cm | The width of the platform |
| k\_BallSize | cm | The diameter of the ball |

# **Behaviour Overview**

This concept was formulated from group discussion and once formalized, pitched to Dr. Wassyng. After discussion, this project was approved for the Computing and Software Department Capstone Project. Over the course of this document the system will be referred to as an autonomous system, the object being adjusted will be referred to as a ball, the ball will be balanced on a platform. The actions carried out by the autonomous system will be referred to as balancing and bouncing.

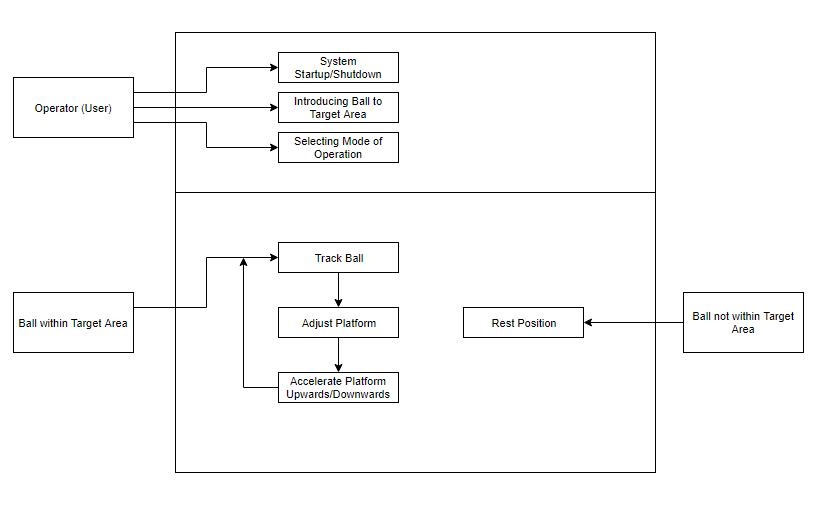
# 

# 

# **Required Behaviour Description**

# 

|  |  |
| --- | --- |
| **State** | **Description** |
| Detect Ball | In this state the system’s downward-facing camera will be searching for the target ball within the target area. |
| Rest Position | When the system cannot detect the ball within the target area, it will return to a flat resting position until a ball is detected. |
| Track the Ball | When a ball is detected within the target area, the system will track its position and velocity in order to determine its trajectory. |
| Adjust Platform | Based on the current position and extrapolated trajectory of the ball, the system will adjust the platform’s orientation to contain the ball within the target area. |
| Bouncing the Ball | When the system is in the user-selected ‘Bounce’ mode, the system’s goal will be to center the ball on the platform while keeping it in the air. |
| Accelerate platform upward upon impact | If the system is in ‘Bounce’ mode and the ball’s bounce height is lower than the expected height, the system will accelerate the platform upwards upon impact with the ball in order to increase the ball’s vertical velocity. |
| Balancing the Ball | When the system is in the user-selected ‘Balance’ mode, the system’s goal will be to center the ball on the platform. |
| Accelerate platform downward upon impact | If the system is in ‘Balance’ mode and the ball is still bouncing off of the platform upon impact, the system will accelerate the platform downward upon impact with the ball in order to dampen the ball’s vertical velocity. The goal of this action is to diminish the vertical velocity of the ball to zero (bringing the ball to rest on the platform’s surface). |



* The rectangles within the square represent the system's reactions to differing external factors.
* The top rectangle represents user-determined states:
  + The user is responsible for powering up and shutting down the system.
  + The user is also responsible for placing the ball into the target area of the system. The system will not have an auto-start feature.
  + The modes of operation for the system will be determined by user-controlled inputs.
* The bottom rectangle represents the system’s reactions to external factors not controlled by the user
* If the ball is within the target area, the system will autonomously track the ball, adjust the platform, and accelerate the platform upwards/downwards.
* If the ball is not within the target area the system will return to a flat resting position.

# **Normal Operation**

## Description

In a normal scenario, the system must be able to detect the location and velocity of the ball in the X, Y, and Z axes, and strategically maneuver the platform to balance or bounce the ball based on user selected modes.

## Normal Use Cases

Ball is detected above platform

When a ball is tossed, rolled, or dropped onto the robotic platform or the space directly above it, the robot will begin to track its position and velocity in the X, Y, and Z axes. It will then begin to adjust the platform based on the predicted path of the ball in an effort to move the ball towards the center of the platform upon impact.

Ball makes contact with the platform in the ‘balance’ mode

When the system is in the user-selected ‘balance’ mode, the platform will accelerate downwards just before contact with the ball in order to slow down the ball’s vertical velocity. Once the ball loses all vertical velocity and is simply rolling around on the platform, the system will focus purely on adjusting the platform.

Ball makes contact with the platform in the ‘bounce’ mode

When the system is in the user-selected ‘bounce’ mode, the platform will accelerate upwards just before contact with the ball in order to maintain the ball’s vertical velocity.

## Undesired Event Handling

### Ball cannot be detected above platform

If the ball cannot be detected above the platform, the platform will return to a flat resting position until it can detect a ball again.

# **Requirements**

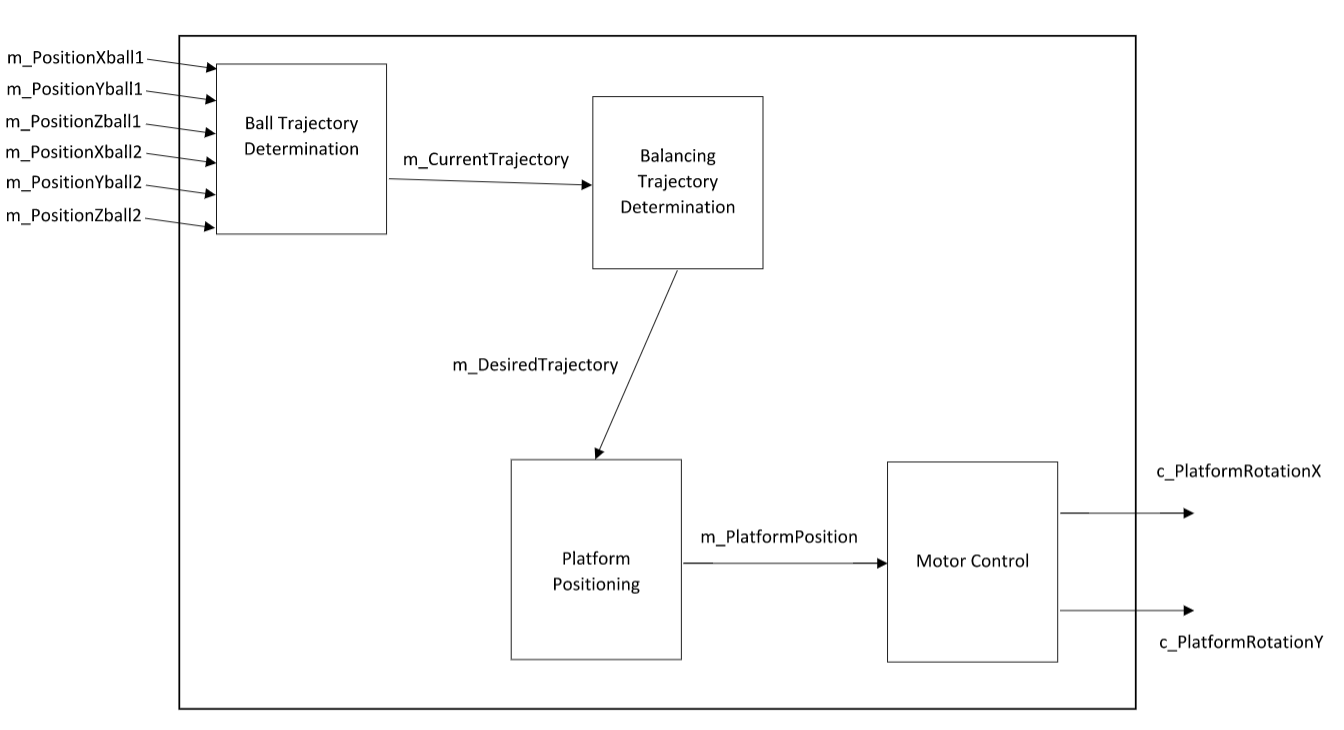
## Functional Requirements

1. The device shall have two modes of operation, balance and bounce.
2. When in balance mode, the device shall balance a ball regardless of the ball’s starting position.
3. When in bounce mode, the device shall bounce a ball and maintain a desired height.
4. The device shall switch between modes based on user input.
5. The device shall indicate which mode it is in.
6. The device shall bring a bouncing ball to rest in its centre when going from bounce to balance mode.
7. The device shall track the planar movement of the ball.
8. The device shall measure the height of the ball from the platform.
9. The device must react quickly to the ball’s movement.
10. The device must not allow the ball to fall off the platform.
11. The device will automatically start the system once the ball is introduced.

## Non-Functional requirements

1. The device must be able to bounce a ball 30 cm from the platform.
2. The device must be portable.
3. The device must be self-contained.
4. The platform will be made from wood
5. The device shall not exceed the allotted budget
6. The device must be easy to use.

## Functional Diagram

­­

List of Requirements that are likely to change

* Maximum height of the ball bouncing

The maximum height of the ball bouncing is a requirement that can potentially change due to the system’s capability that will be defined during testing.

* Number of modes of operation

The number of modes of operation can change as outlined in the project goals document as there are additional features/operations that can be integrated into the system however, this depends on the time left till our final deliverable.

List of Requirements that are not going to change

* The device shall have two modes of operation, balance and bounce.
* When in balance mode, the device shall balance a ball regardless of the ball’s starting position.
* The device shall track the planar movement of the ball.
* The device shall measure the height of the ball from the platform.
* The device must react quickly to the ball’s movement.
* The device must not allow the ball to fall off the platform.
* When in bounce mode, the device shall bounce a ball.
* The device shall switch between modes based on user input.
* The device shall indicate which mode it is in.
* The device shall bring a bouncing ball to rest in its centre when going from bounce to centre mode.

The above functional requirements will not change as these are the fundamental requirements for the system to behave in it’s normal operation modes.