EP1010 Botball

#160

Meetings

Team

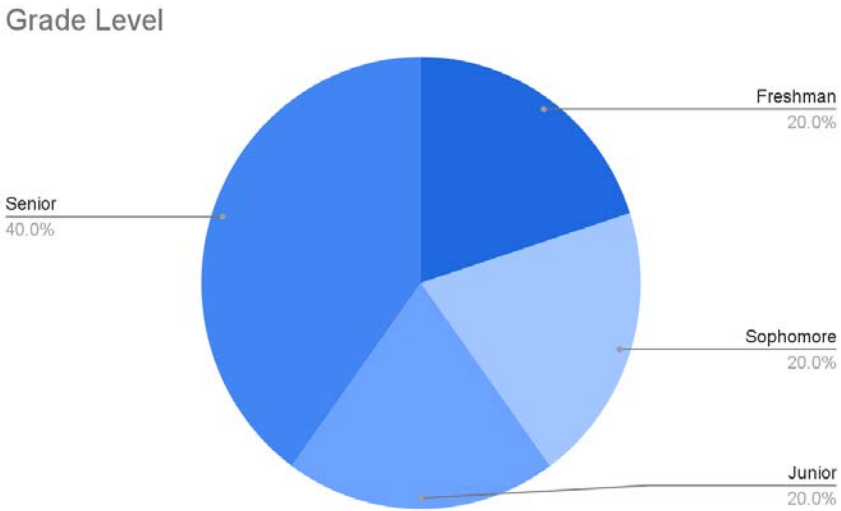
Non-school based (extracurricular)

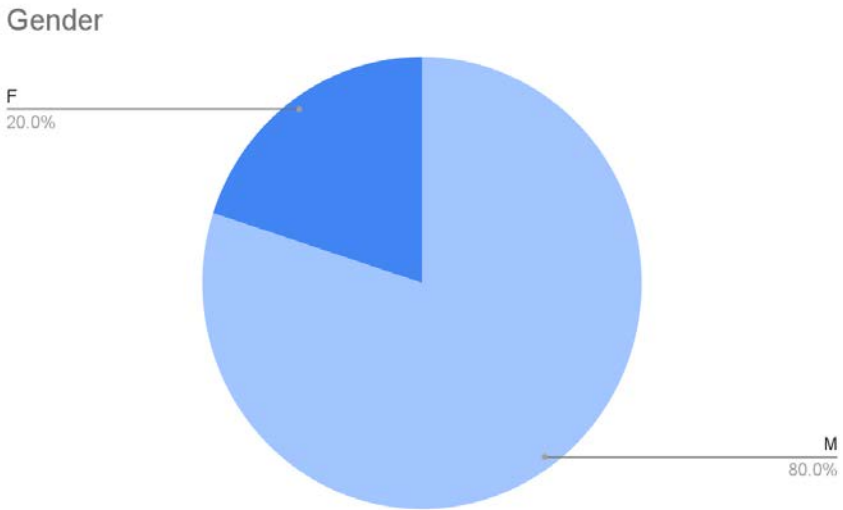
Address

36 Maryland Ave, Rockville, MD 20850

Mon	Tue	Wed	Thu	Fri	Sat	Sun
6-9 pm	—	6-9 pm	—	—	10-1 pm	—

Demographics





Team Organization

Builders

Build mechanisms and manage robots

Team Captain

Delegates work and strategizes

Coders

Code/test robots

Learning Goals

Develop and implement effective designs

Devise an overall strategy for the game

Build prototypes and improve upon them

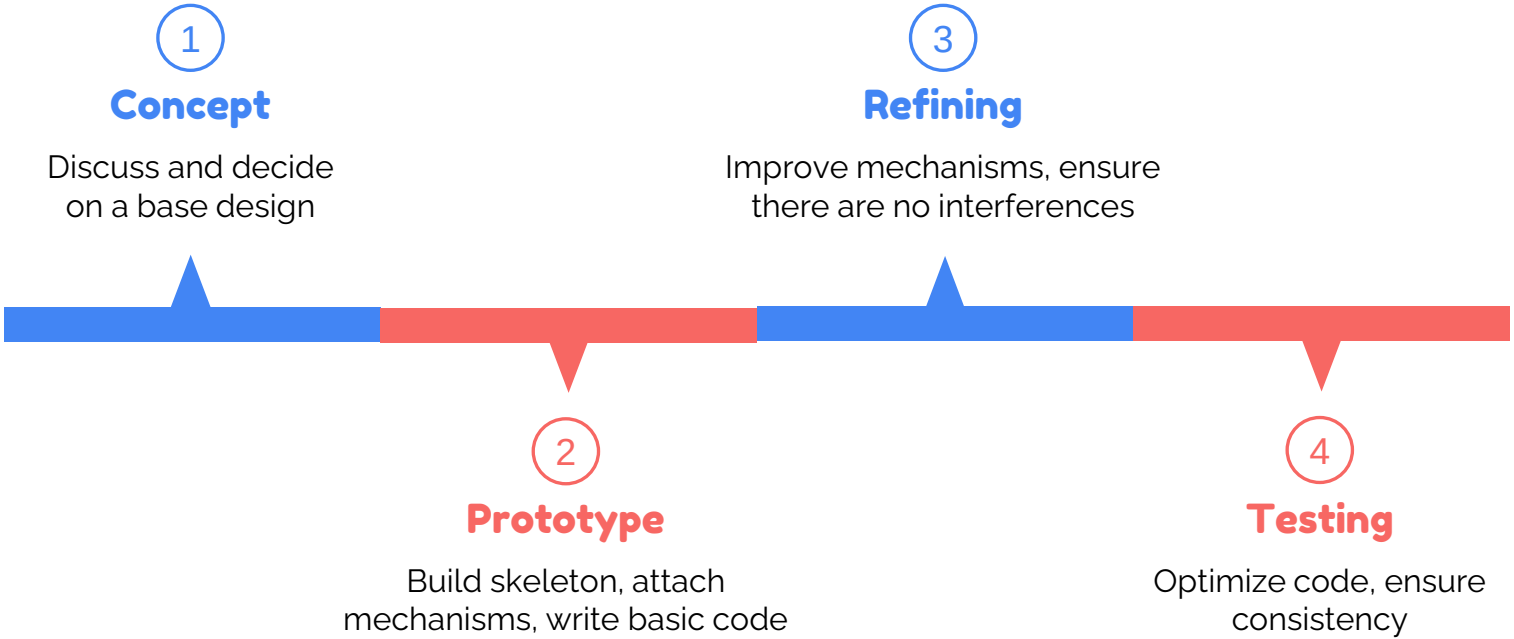
Cultivate ability to work as a team

Resolving Conflicts

Ex. When we have conflicting designs/strategies

Discussion & Majority Rule

Project Iteration



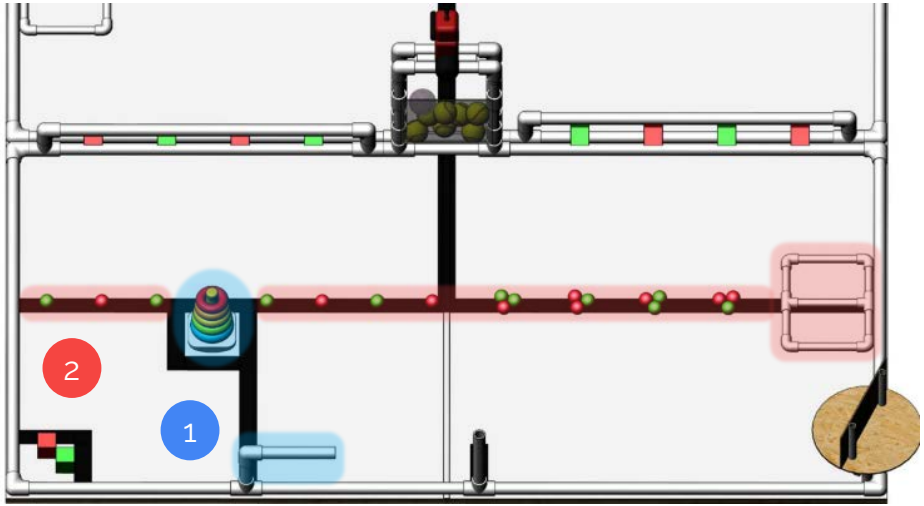
Initial Game Strategy

Roomba Bot

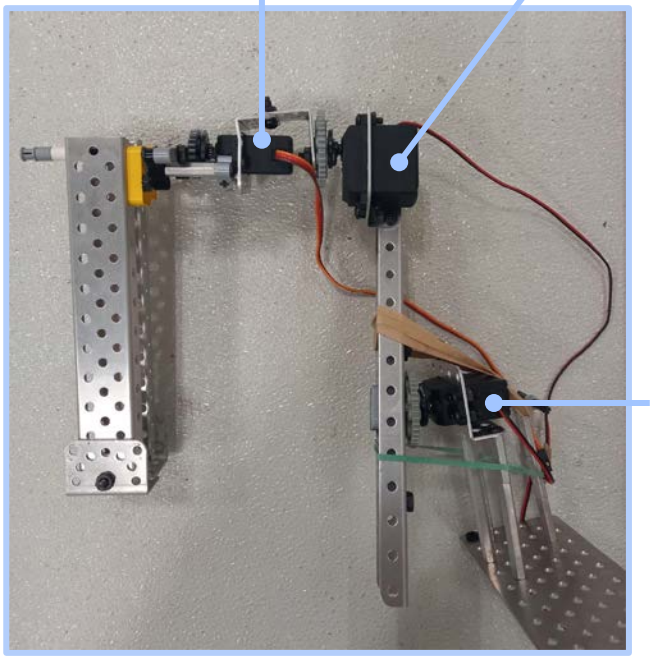
Get 3 rings onto horizontal electrophoresis

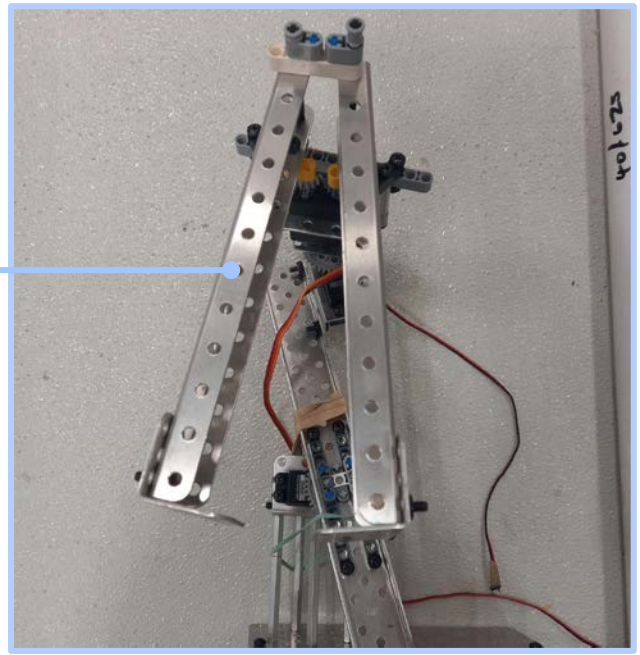
Bulldozer Bot

Push/scoop poms into the transporter and drag everything to starting box



Initial Design





Initial Pseudcode

Roomba Bot

Initialize
- Close claws
- Enable servos

Wait for light
Open claw
Drive forward
Close claw
Rotate claw
Rotate 90° to the right
Drive forward
Rotate to the right (to get rings on)
Stop

Bulldozer Bot

Initialize
- Enable servos
- Move servo to ground

Wait for light
Drive forward and find black line
Rotate 90° to the right
Line follow
Servo control (raise arm)
Servo control (lower arm)
Rotate 180°
Line follow
Rotate 90° to the left
Drive forward
Stop

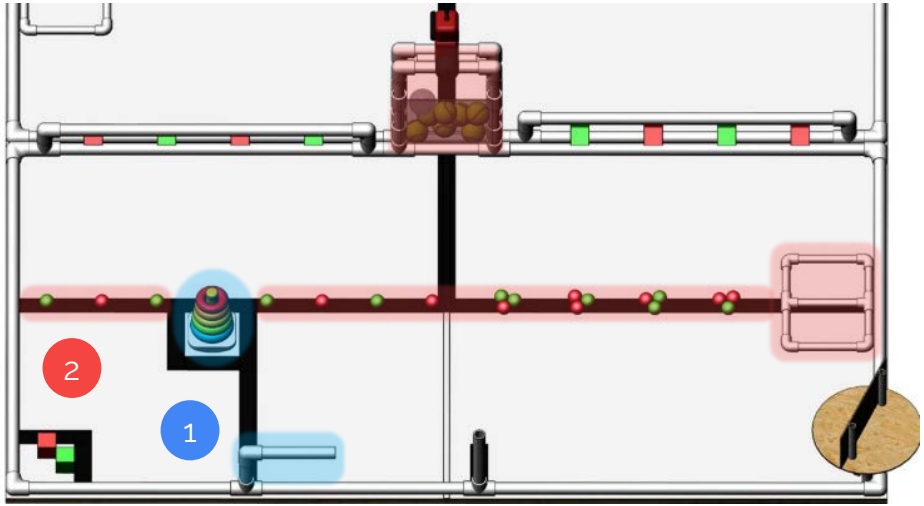
Mid GCER Game Strategy

Roomba Bot

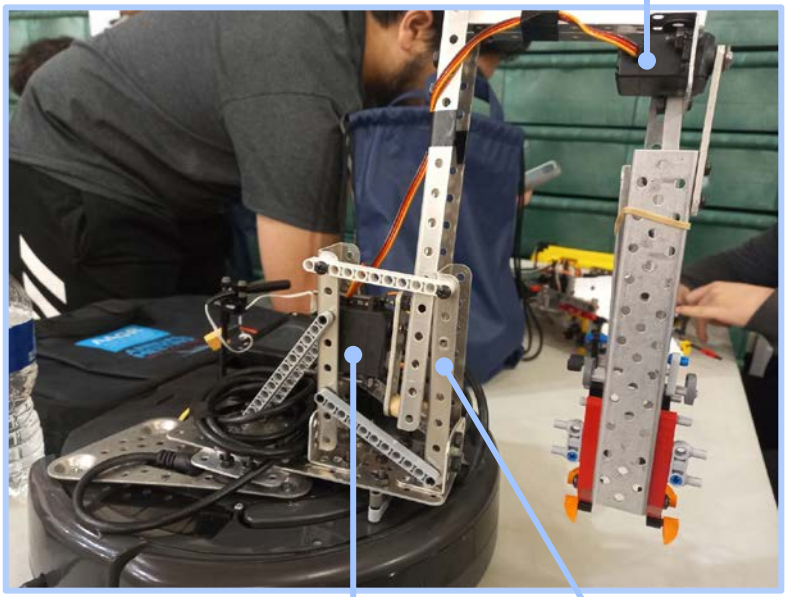
Get 3 rings onto horizontal electrophoresis

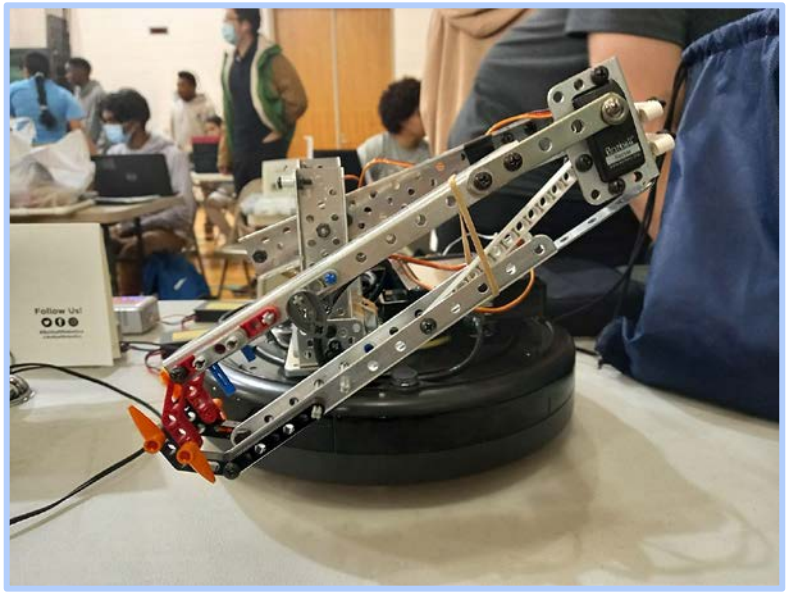
Bulldozer Bot

- Get poms into the transporter
- Drag transporter to PCR machine
- Release tennis balls
- Drag everything to starting box

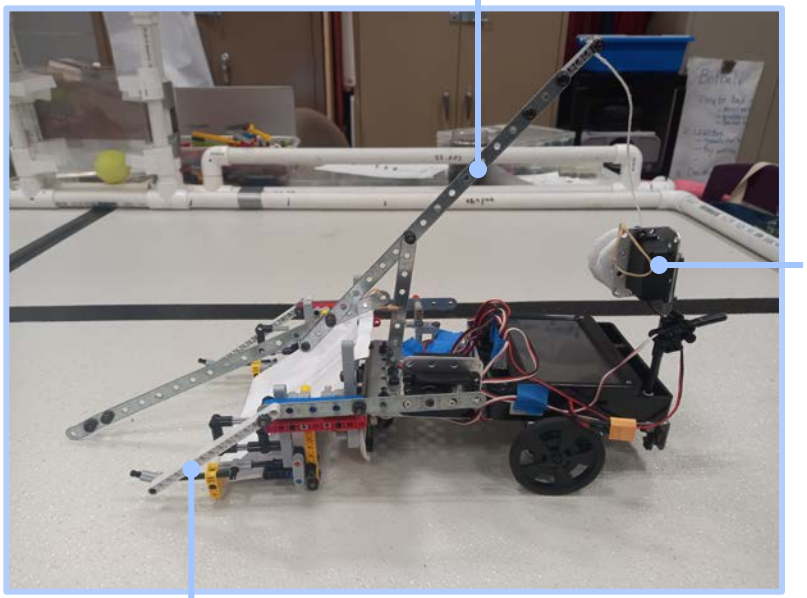


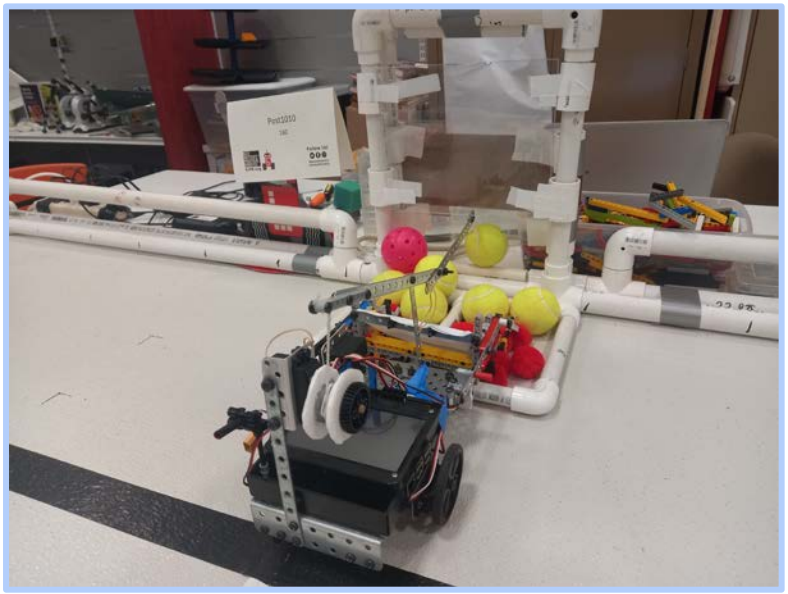
Mid GCER Design





Mid GCER Design





Mid GCER Code

Roomba Bot

```
def main():
#Roomba Setup
• • •
# Back align with wall
   back_align()
# Move claw into operating position
   straighten_claw()
   claw_open()
# Set Roomba to Claw Position
   drive_towards_rings()
# Reduce extraneous momentum
   stop(50)
# Tighten claw around rings
   claw_tighten()
# Rotate claw and Roomba to remove rings
   left_rotate_servo()
# Align Roomba to rotation position
   drive_toward_cylinder()
```

Mid GCER Code

Bulldozer Bot

```
def main():
# Align bot on line
    go_to_black(100, 100)
    move(100, 0, 1400)
# Black line follow
    line_follow(20500)
# Going to middle of game board
• • •
# Rotate 90d to the left
    move(-25, 25, 3450)
    stop(100)
# Releasing tennis balls
# Rotate 90d to the left
    move(-50, 50, 1200)
    while(KIPR.analog(TOPH_RIGHT) > BLACK):
        move(0, 25)
    stop(500)
    line_follow(8000)
```

```
def line_follow (time, sensor=TOPH_LEFT):
    end_time = KIPR.seconds() + time
    while (KIPR.seconds() < end_time):
        if (KIPR.analog(sensor) > BLACK):
            if (sensor == TOPH_LEFT):
                move(37, 50)
            else:
                move(50, 37)
        else:
            if (sensor == TOPH_LEFT):
                move(50, 37)
            else:
                move(37, 50)
```

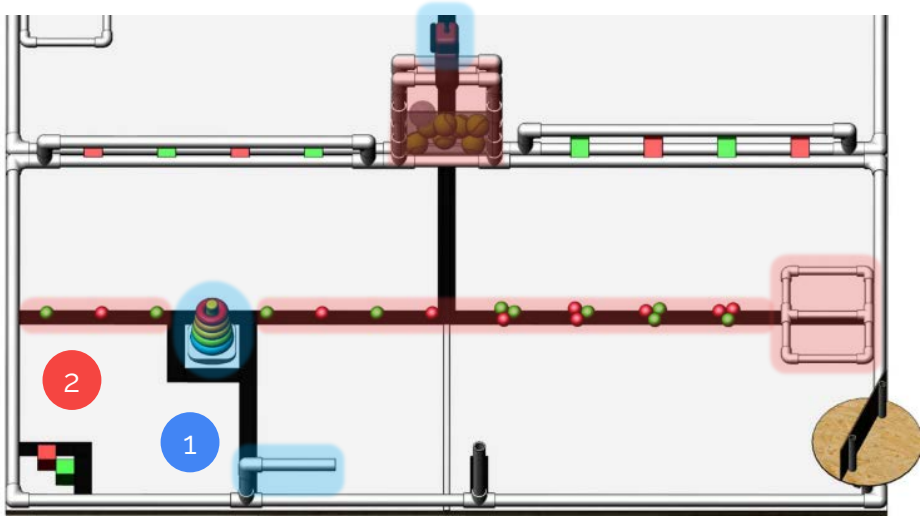
Final GCER Game Strategy

Roomba Bot

- Get 3 rings onto horizontal electrophoresis
- Knock Botguy down

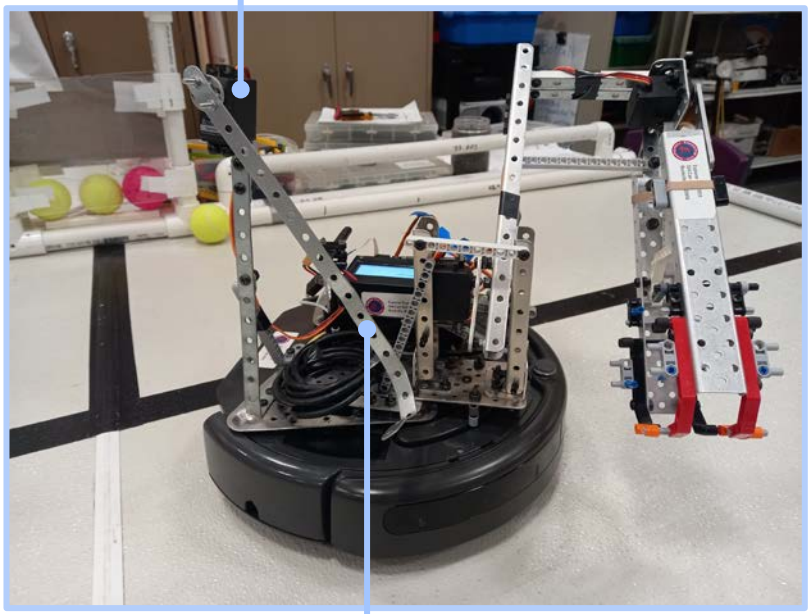
Bulldozer Bot

- Get poms into the transporter
- Drag transporter to PCR machine
- Release tennis balls
- Drag everything to starting box

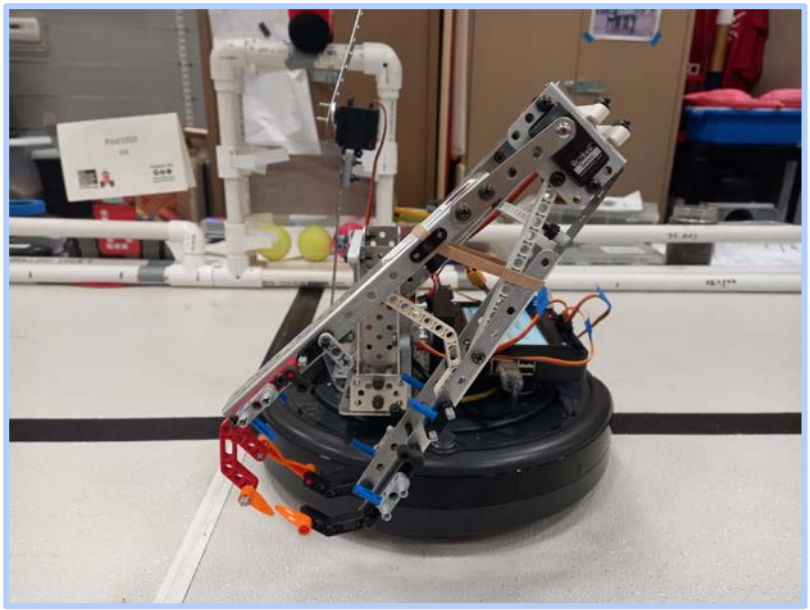


Final GCER Design

Motor



Arm



Final GCER Code

Roomba Bot

```
def main():
    ...
    # Release rings
    # Wait for other bot for 80 secs
    drive_backward_to_middlepipe()
    rotate_to_botguy()
```

```
def drive_backward_to_middlepipe():
    drive(-150, -150, 800)
    drive(150, -150, 400)
    drive(-150, -150, 2800)
    drive(-150, 150, 400)
    claw_close()
    KIPR.msleep(500)
    raise_botguy_hook()
    drive(-150, -150, 4000)
    drive(100, 100, 500)
```

Final GCER Code

Bulldozer Bot

```
def main():
    . . .
# Rotate 90d to the right
    move(25, -25, 1600)
    while(KIPR.analog(TOPH_LEFT) < BLACK):
        move(25, -25)
    stop(500)
# Back line follow
    blf(3000, 30)
```

```
def arm_control(position):
    KIPR.clear_motor_position_counter(ARM_MOTOR)
    while(KIPR.get_motor_position_counter(ARM_MOTOR) > position):
        if (position == UP):
            KIPR.motor(ARM_MOTOR, 10)
        elif (position == DOWN):
            KIPR.motor(ARM_MOTOR, -10)
    KIPR.off(ARM_MOTOR)
```

Risks

Timing

Timing of the two bots

Clustering

Friction/clustering of the poms and tennis balls

Slip

Roomba slipping, causing angle inconsistency

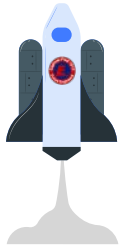
Community Impact

Running our Instagram account @exp1010botball

Introducing our team to visitors of the Rockville Science Center

Volunteering during the Science Center events (Rockville Science Day)

Helping out FLL Teams



Thank You

Explorer Post 1010