I. Introduction

Our project was to design a robot for the IEEE SoutheastCon Hardware Competition that can navigate and complete tasks within the course. It needs to be able to follow a path, collect beads off of “trees” and throw/place them in nets and cups, and push a marshmallow into the “alleyways”.

II. High Fidelity Concepts

1:

- Brain: Raspberry Pi 4
- Language: C
- Wheels: 2 standard wheels, 1 caster
- Steering: Center line tracking and barrier tracking
- Motors: Brushless motor
- Battery: LiPo batteries
- Sensors: LiDAR for barriers, reflective IR for line tracking, and camera for Net/Tree/Cup Detection
- Display: Generic screen and folding joint for multimedia display
- Speaker: Simple circuit speaker (may be included in generic screen)
- Collector: Fish Hook for Bead Collection
- Storage: Container to store beads connected to launching mechanism
- Launching Mechanism: Flywheel launcher for beads
- Marshmallow Mechanism: Fan to collect marshmallow
- Marshmallow Storage: No storage - Drag it along
- Base: Aluminum Base
- Attachment Methods: Bolts for attaching base and wheels, etc.
- Connection Methods: Polyurethane connection cables, soldered connections to components, and breadboard to connect components to the base
- Decoration: COE colors and text

2:

- Brain: Raspberry Pi 4
- Coding Language: Python
- Wheels: Standard wheels with caster
- Steering: Line tracking and barrier tracking
- Motors: Linear motor
- Power Supply: Lithium battery
- Sensors: LiDAR
- Display and speaker: Multimedia display with an accordion extender arm
- Bead Collector and launcher: Vacuum arm with a retractable grate and a flywheel launcher
- Marshmallow Mechanism: Moving extensions of walls longitudinal
- Physical Base: Aluminum sheet
- Attachment Methods: Screws and bolts, solder
- Connection Methods: Copper wiring
- Decoration: COE colors

3:

- Brain: Arduino Mega 2560
- Coding Language: C++
- Wheels: Standard(x2) and Caster(x1) for simple programming
- Steering: Line and Barrier Tracking for tracking the environment and the robot's position
- Motors: Brushless for more efficient operation of the wheels
- Power Supply: Lithium
- Sensors: LiDAR, Camera, and IR for environmental scanning and data collection
- Display and speaker: Android Phone and Robotic accordion extender arm
- Bead Collector: Vacuum Collector
- Bead Storage: Vacuum Collector container with door and grate
- Bead Launcher: Flywheel Launcher
- Marshmallow Mechanism Triangular Bumper pushing marshmallow aside and Accordion arm extender with caster wheel to push marshmallow into the opening on the track
- Physical Base: Aluminum Sheet for lighter chassis
- Attachment Methods: Hot Glue and screws and bolts
- Wiring: Polyurethane Cables
- Decorations: COE Colors and Goose for school spirit
- Connections: Breadboard for ease of swapping out parts and soldering for parts that need to be soldered

III. Selected Concept

- Brain: Raspberry pi
- Coding Language: C
- Wheels: Standard(x2) and Caster(x1) for simple programming
- Linear Movement/Steering: Line and Barrier Tracking for tracking of the environment and the robots position
- Motors: Brushless for more efficient operation of the wheels
- Power Supply: NiMH
- Sensors: camera, transmissive IR for environmental scanning and data collection
- Display and sound: Display and speaker
- Bead Collector: Arm
- Bead Storage: Arm
- Bead Launcher: Flywheel
- Marshmallow Mechanism: Accordion arm w cup
- Physical Base: Aluminum Sheet for lighter chassis
- Attachment Methods: screws and bolts
- Wiring: Polyurethane Cables
- Decorations: COE Colors for school spirit
- Connections: Breadboard for ease of swapping out parts and soldering for parts that need to be soldered

IV. Design Revisions

The selected concept was derived from thorough comparisons of the high fidelity designs. The analysis and comparison was done using a variety of methods including House of quality charts, AHP/ PUGH tables, etc.. The resulting selected design is therefore a combination of the best qualities of all three high fidelity concepts.

The overall design went through many revisions from the initial selected concept to the final design. The discussed revisions first go over the structural changes and the changes of the components and systems. These changes were made in the process of testing and integration. While assembling the robot, major design flaws became clear and these changes reflect the solutions to problems encountered.

The first major change to the design was making the base round instead of square. This change would allow the robot to turn in place within the small track without making contact with the barriers. With this change the robot also needed a second caster wheel for stability. The second large design change involved increasing the size of the robot by adding a raised platform in order to add more components to the robot. To reduce the amount of complex mechanisms and due to a rule change within the competition rules, the marshmallow mechanism was changed and instead replaced with a passive bumper that prevents the marshmallow from getting run over and instead pushes it onto the side of the track. The final structural revision was the change from a metallic base to 3D printed bases made from Polylactic Acid (PLA). This change made our robot much lighter as well as made the design much more customizable, allowing for holes and indents to be printed to accommodate the components.

One of the most important changes was the removal of the Raspberry Pi and the Pi camera and replacing them with Arduino Mega 2560 and a PixyCam. This change simplified the image recognition process by using pre-made image training software instead of having to create and train an image recognition algorithm. The loss of the Raspberry Pi meant that we also lost the ability to play music on the robot since Arduinos are not well suited to playing music. A revision made through the initial stages of the development of the robot was using the Arduino for the navigation algorithm, drivetrain functions, and sensor controls. This was done in order to take advantage of its real-time operation, making the system simpler and more responsive than its implementation within the Raspberry Pi would have been. The brushless motors were swapped for brushed ones since the slightly higher efficiency of the brushless motors did not warrant the cost. The flywheel launcher was removed in favor of a simpler catapult that could simultaneously store the collected beads. The final large revision is the removal of IR sensors within the design. The reliability of the IR sensors was not good enough to justify their use, therefore the robot’s navigation relied solely on line tracking in the final design.

V. Final Detailed design

- Microprocessor: Arduino Mega(x2) and Arduino UNO
- Coding Language: C++17
- Wheels: Standard(x2) and Caster(x2)
- Steering: Line tracking robots position
- Motors: Brushed DC electric motor(x2)
- Power Supply: NiMH
- Sensors: PixyCam and line sensors(x7)
- Display: LCD display
- Bead Collector: Arm
- Bead Storage: Catapult
- Bead Launcher: Catapult
- Marshmallow Mechanism: Front bumper
- Physical Base: 3D printed two-tier base