CDR

UMass Amherst
COLLEGE OF ENGINEERING

Lazy Bob - SDP23 Team 5

Omar Areiqat, Vedatman Basu, Redwan Rahman, Jina Song

The Team

Omar Areiqat
CompE
Budget Lead
Movement and Power

Jina Song
EE
PCB Lead
Design

Vedatman Soham Basu
CompE
Software Lead
Sensors

Redwan Rahman
CompE
Logistic Lead
Positioning

William Leonard
Advisor

Problem Statement

Some people struggle with mundane actions in life, such as passing dishes on a large table.

Lazy Susans are nice, but only work on round tables and requires user to have reach.

What if there was a small, portable device that can move dishes across any table of arbitrary shape?

Our Solution

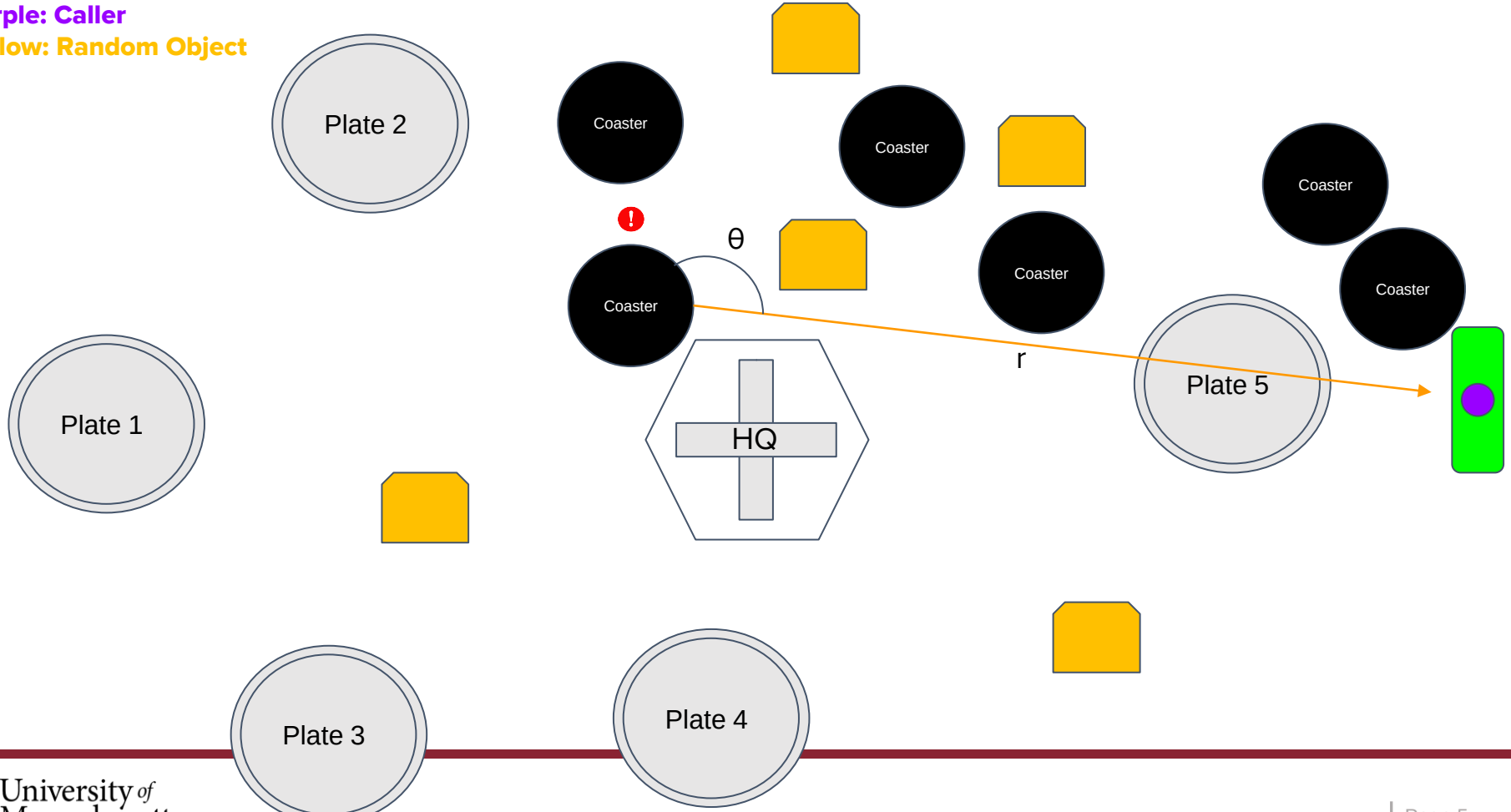
“Lazy Bob”: A moving coaster that approaches the user when called.

1. Stably transport items around a table to different users

1. Provide a simple way to interface with all coasters

1. Adapt to the environment

Purple: Caller
Yellow: Random Object



Functional and Design Specification

System Specifications: Functional

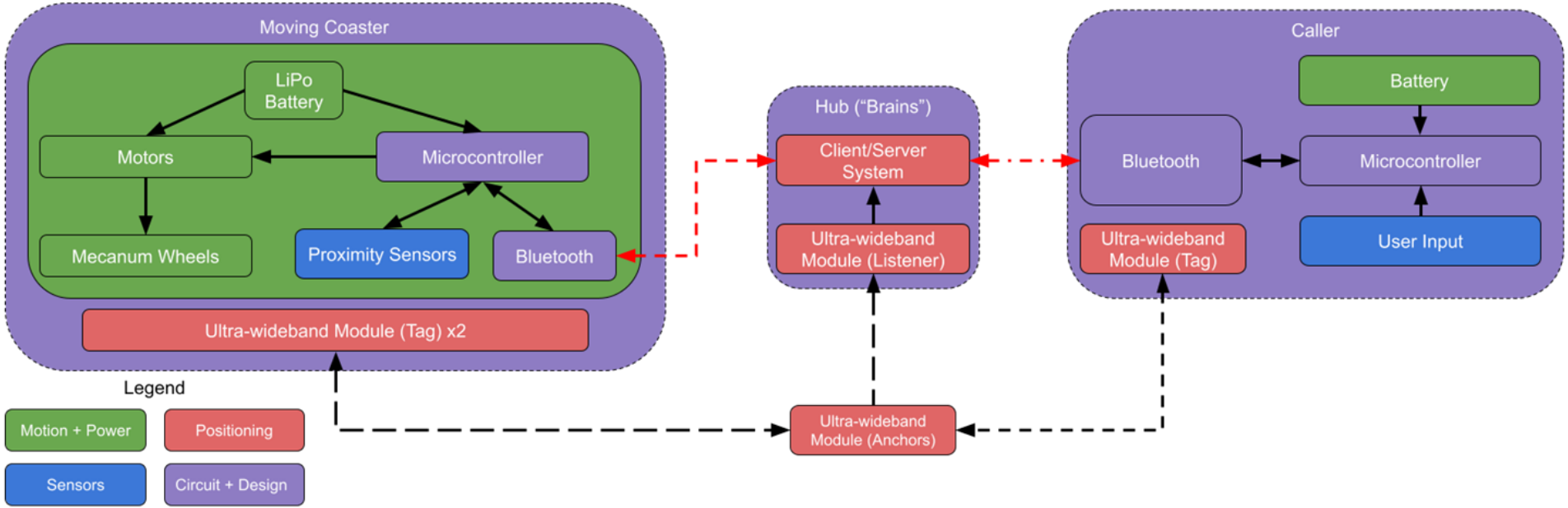
	Test Plan
1. Mostly full container of liquid doesn't spill when moving	1. Place an 80% full glass/bowl of water and visibly check that the moving coaster doesn't spill
1. Transport items without the items falling off	1. Put objects on moving coaster and watch if object falls
1. Detect the edges of the table and avoid falling off	1. Watch if the moving coaster falls off when caller is away from the table
1. Avoid objects that are in the coaster's moving path	1. Put objects in path and check if moving coaster bumps into object
1. Arrive at the caller to an accuracy of 10 cm from the edge of the coaster	1. Use a tape measure to check the distance from destination
1. Support up to 3 lbs.	1. Place an object that weighs 3 lbs and check if system is stable while moving

System Specifications: Design

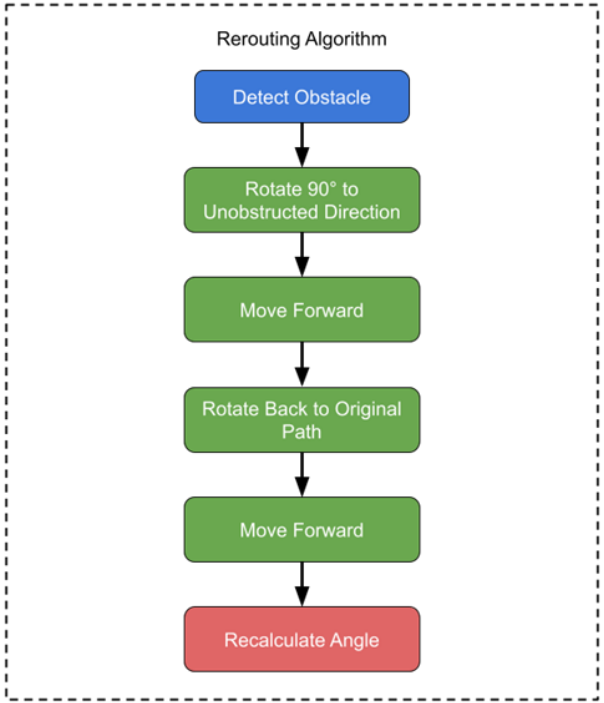
		Test Plan
7.	System supports at least 2 callers at the same time	Press both callers and see if the coaster follows first-in, first-out sequence
8.	Coaster diameter is within 30 cm	Use tape measure to check coaster size
9.	Be able to use the moving coaster regardless of table shape	Find and put on irregular shaped tables and check if they fall off
10.	If no path to caller is available, indicate to caller	Line up objects to allow no path to caller, see if system notifies caller

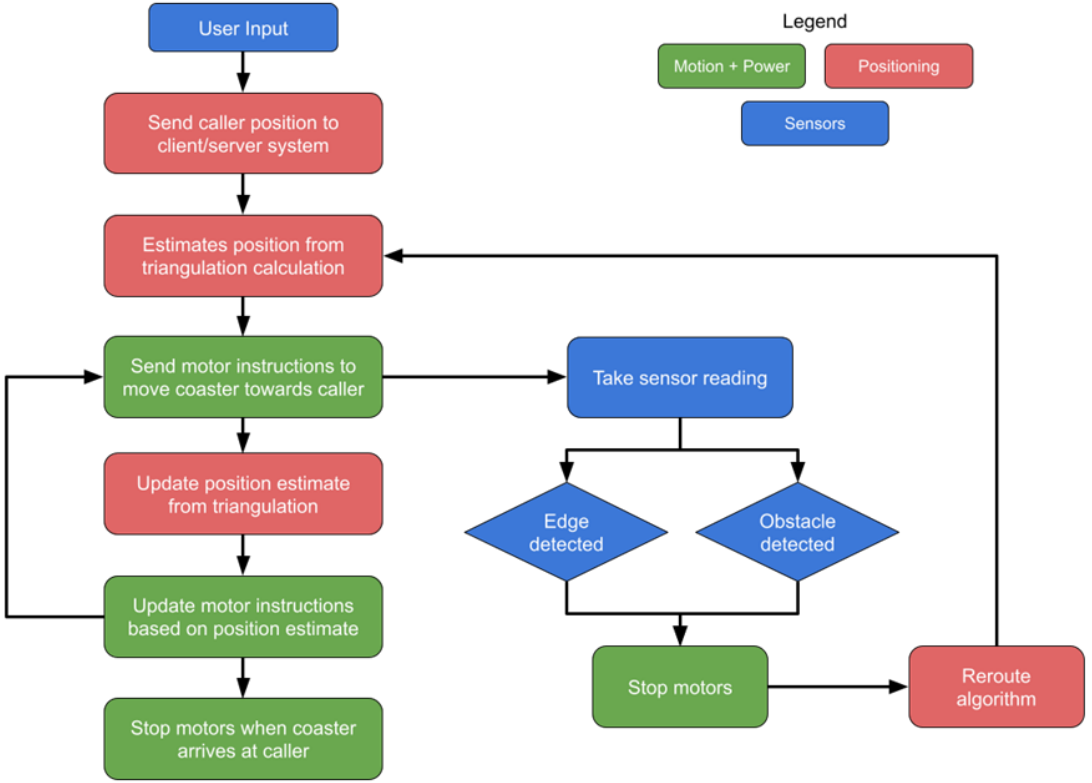
Block Diagrams

Hardware Block Diagram



Software Block Diagram

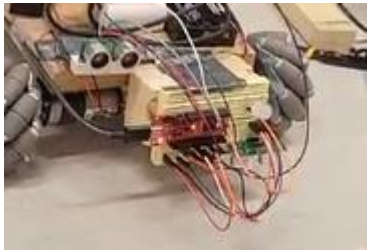


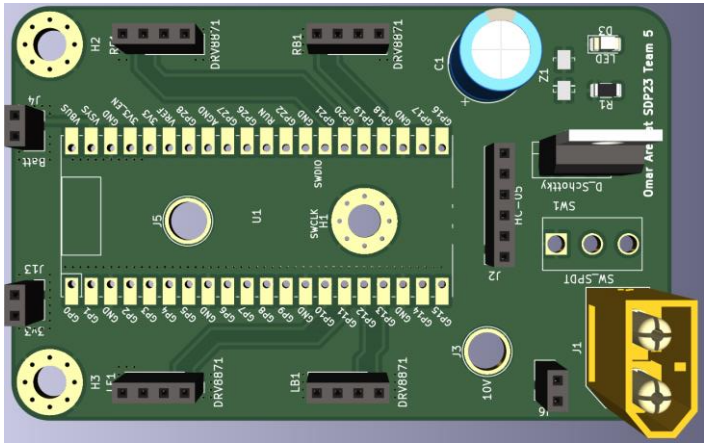


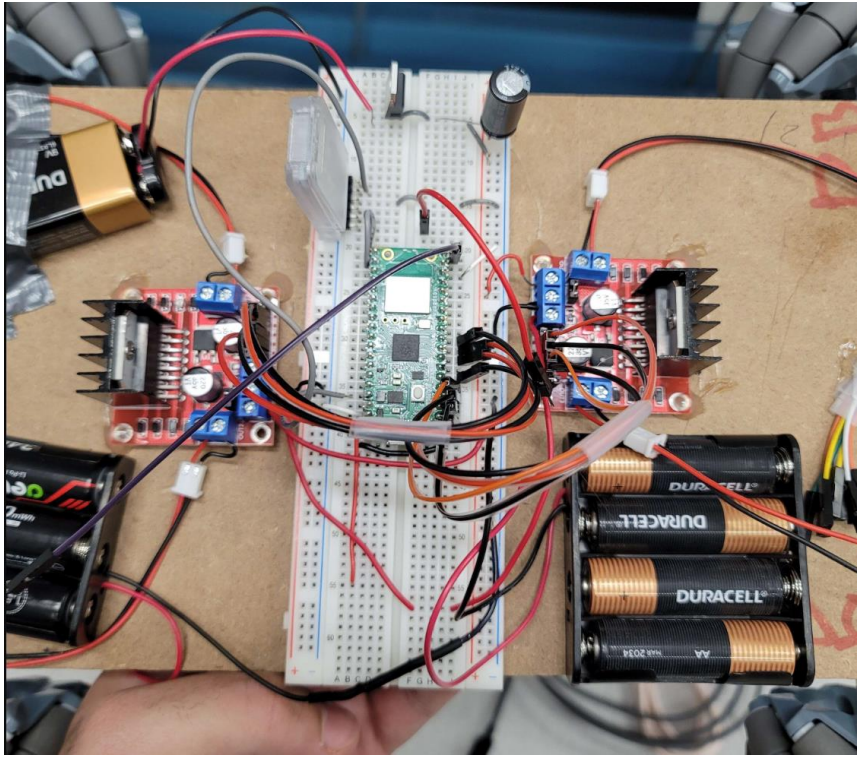
System Hardware

Coaster

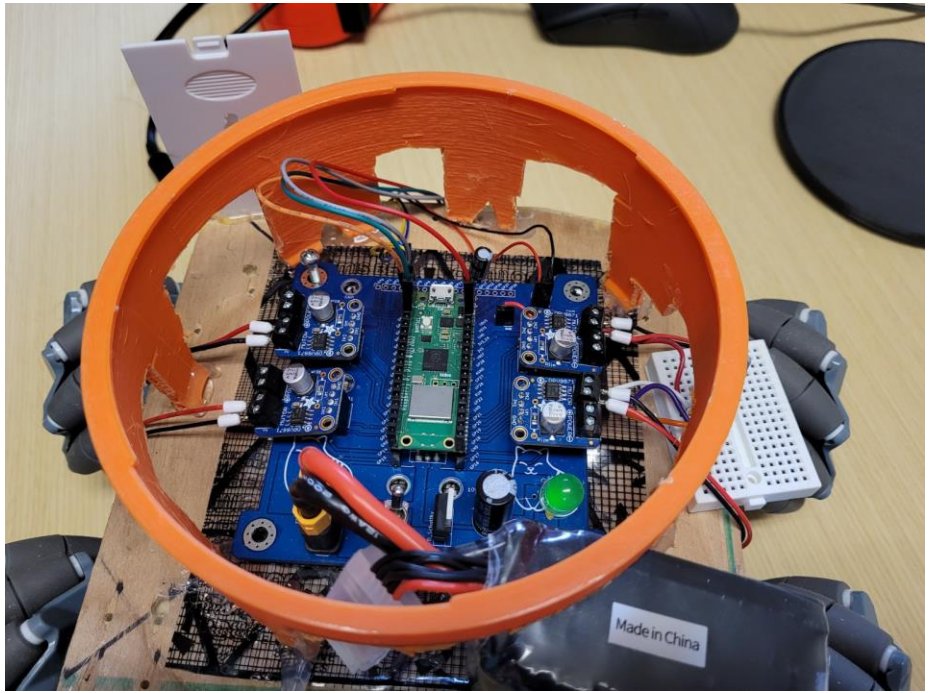
Coaster: MDR

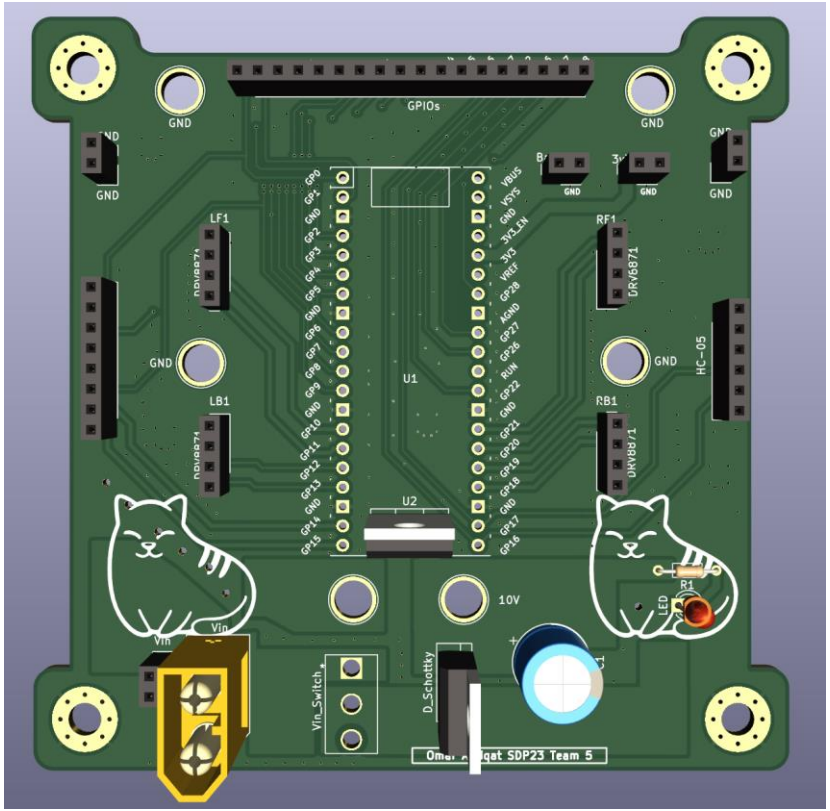


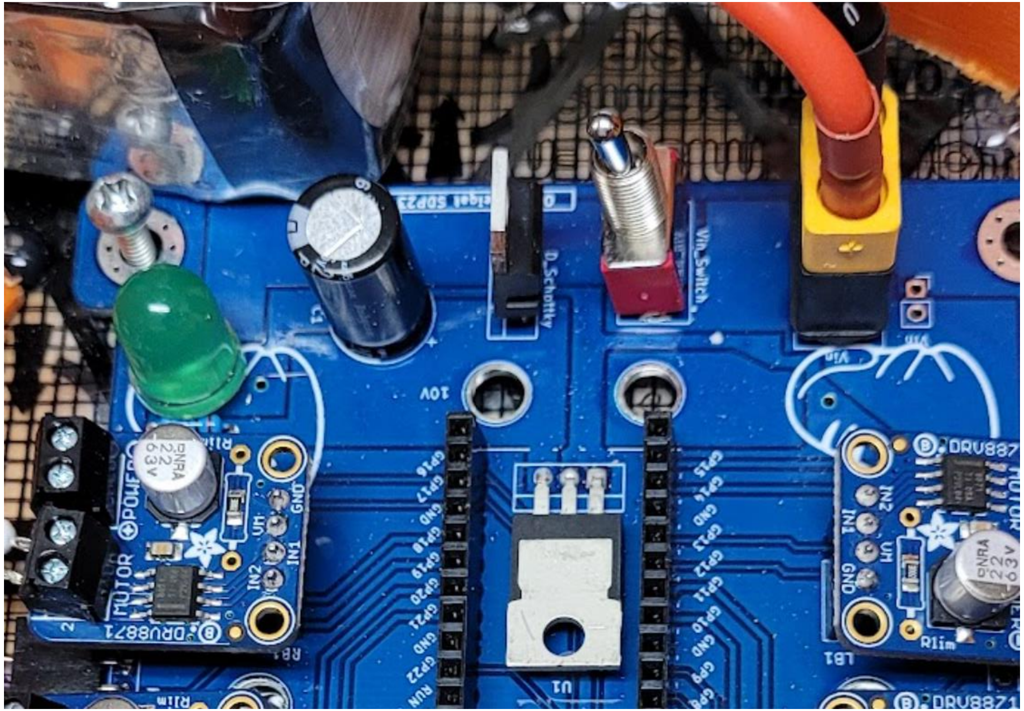

DRV8871
DRV8871
DRV8871
DRV8871
SDP23 Team 5
Omar Are
D_Schottky
SW_SPDT
SWDIO
SWCLK
HC-05
10V
LED


DURACELL
DURACELL
DURACELL

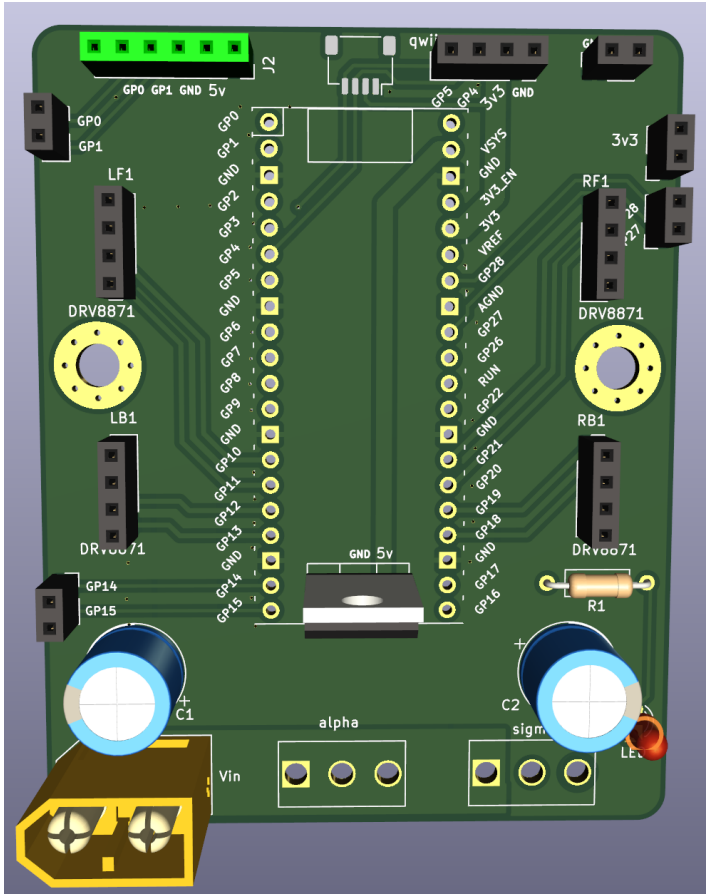
Coaster: CDR


Made in China


GPIOs
GND
LF1
RF1
LB1
RB1
U1
U2
HC-05
10V
Vin_Switch
D_Schottky
SDP23 Team 5


D_Schottky
Vin_Switch
10V
DRV8871
U1

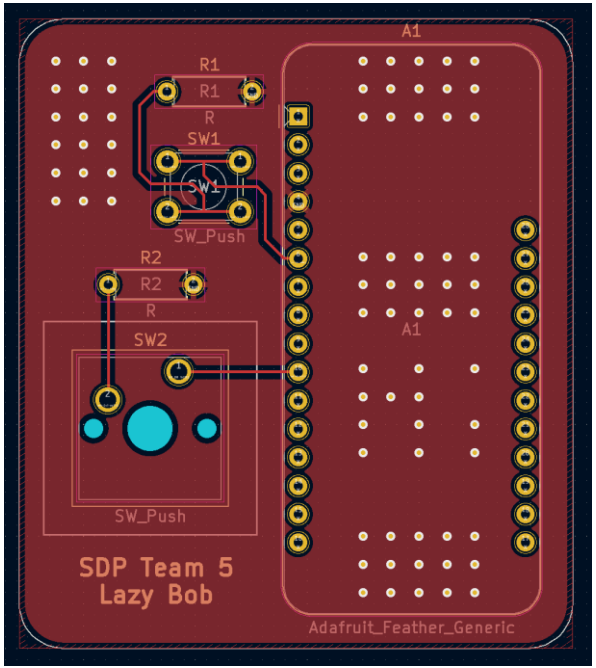
Coaster: FPR PCB



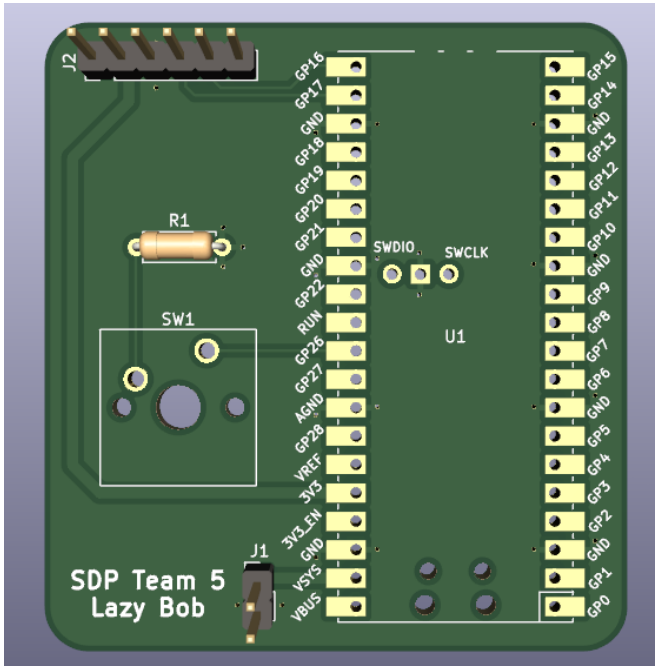
Caller

Subsystem: Caller PCB

MDR



CDR

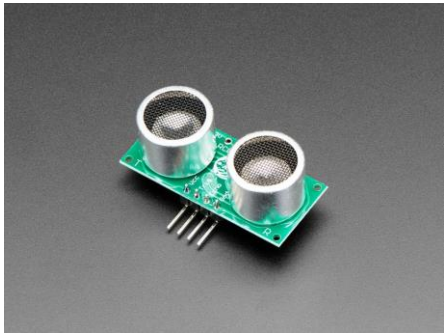


FPR

Sensors

Subsystem: Collision and Cliff Detection

- **Ultrasonic**
  - Collision detection (CDR)
  - Has not changed from before



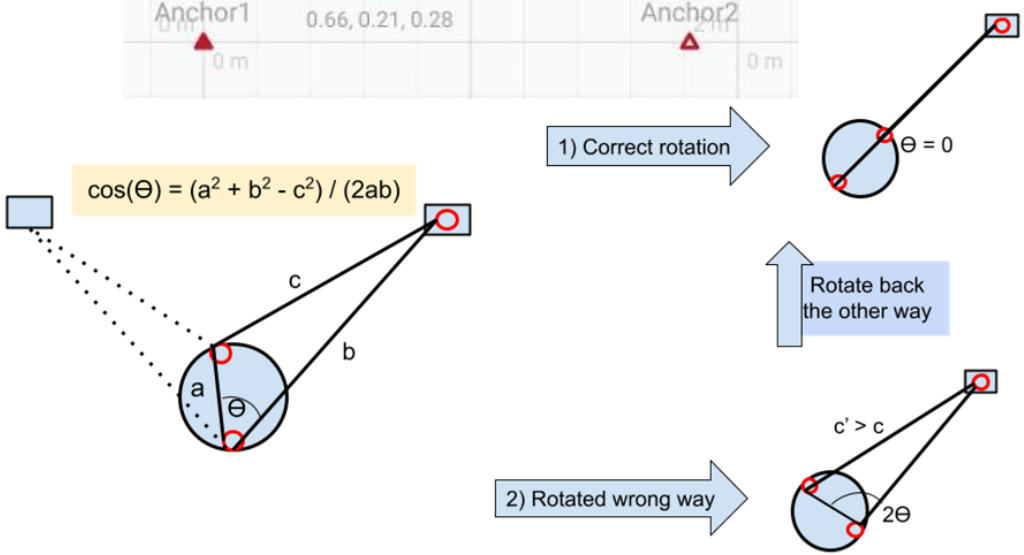
- **Light ToF**
  - Edge Detection (CDR)
  - Collision Detection (FPR)
  - Easy to implement many at a time

Positioning and Rerouting

Subsystem: Positioning

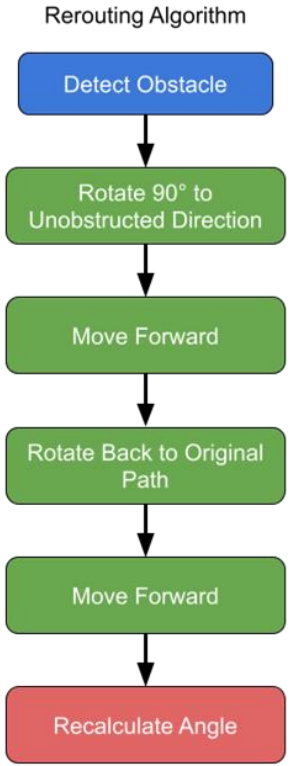
- **Ultra Wideband Sensor**


```
dwm> lep
dwm> POS,0,8312,-0.16,0.65,0.08,87,x00
POS,0,C404,0.16,-0.12,0.10,96,x0B
POS,0,8312,-0.18,0.67,0.02,90,x00
POS,0,C404,0.15,-0.15,0.11,87,x0B
POS,0,8312,-0.18,0.68,-0.05,88,x00
POS,0,C404,nan,nan,nan,0,x0B
POS,0,8312,-0.18,0.67,-0.10,88,x00
POS,0,C404,nan,nan,nan,0,x0B
lep
```



https://www.frontiersin.org/articles/10.3389/fnbot.2022.715440/full
https://www.qorvo.com/products/p/MDEK1001

Rerouting Block Diagram



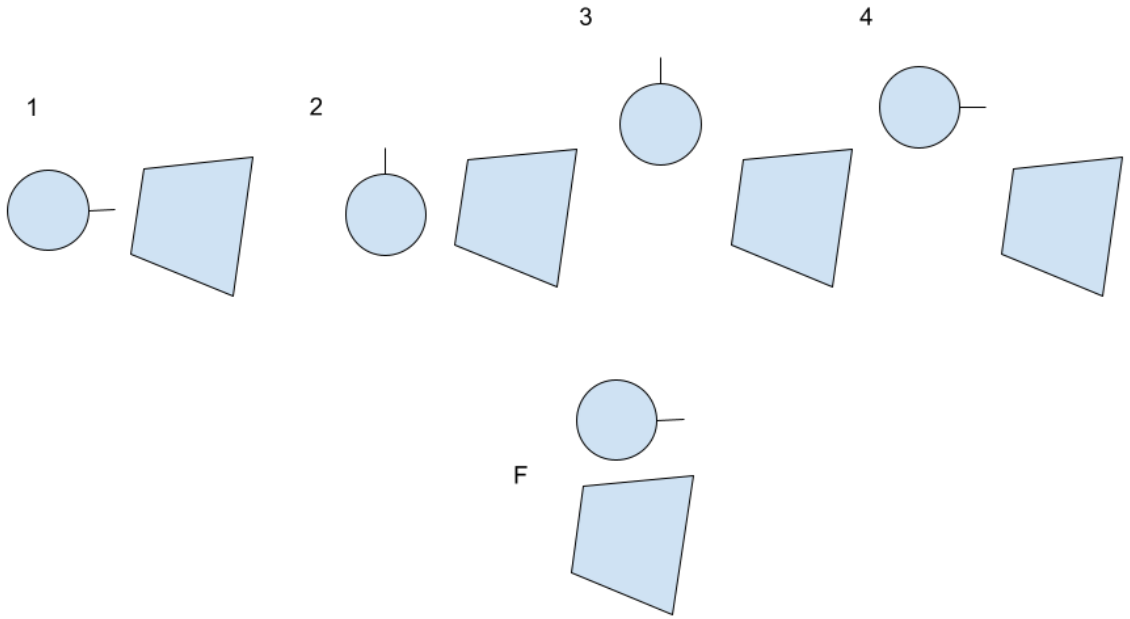
Subsystem: Algorithm

Current Iteration:

Brute Force Avoidance - Lazy Bob will back up, make a step to a side, then move forward again and check whether it has cleared the object. This will repeat if not TRUE.

Future Iteration:

Lazy Bob will fix two distances and trace its way around an object prioritizing the side with the most severe angle.



FPR Deliverables

1. **Improved consistency on movement**
   a. Move movement controls to the coaster itself for better consistency

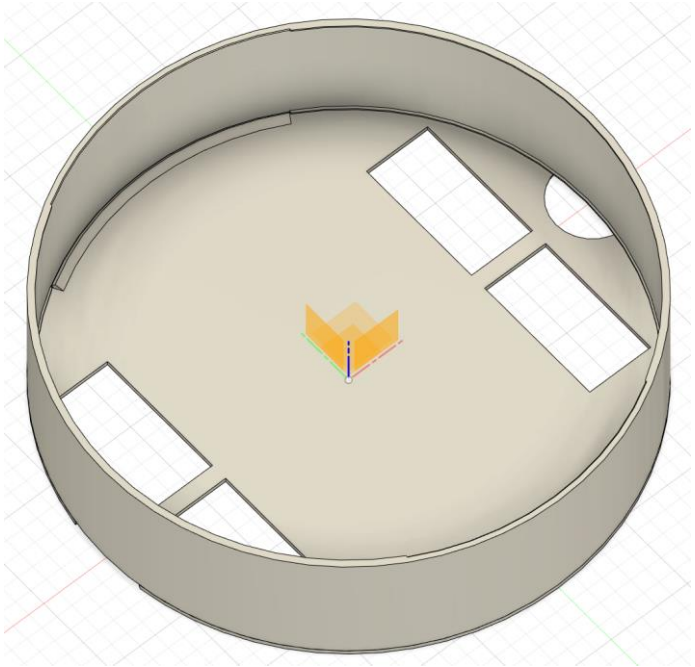
1. **Better Rerouting Algorithm**
   a. Using more sensors, implement a better way to navigate

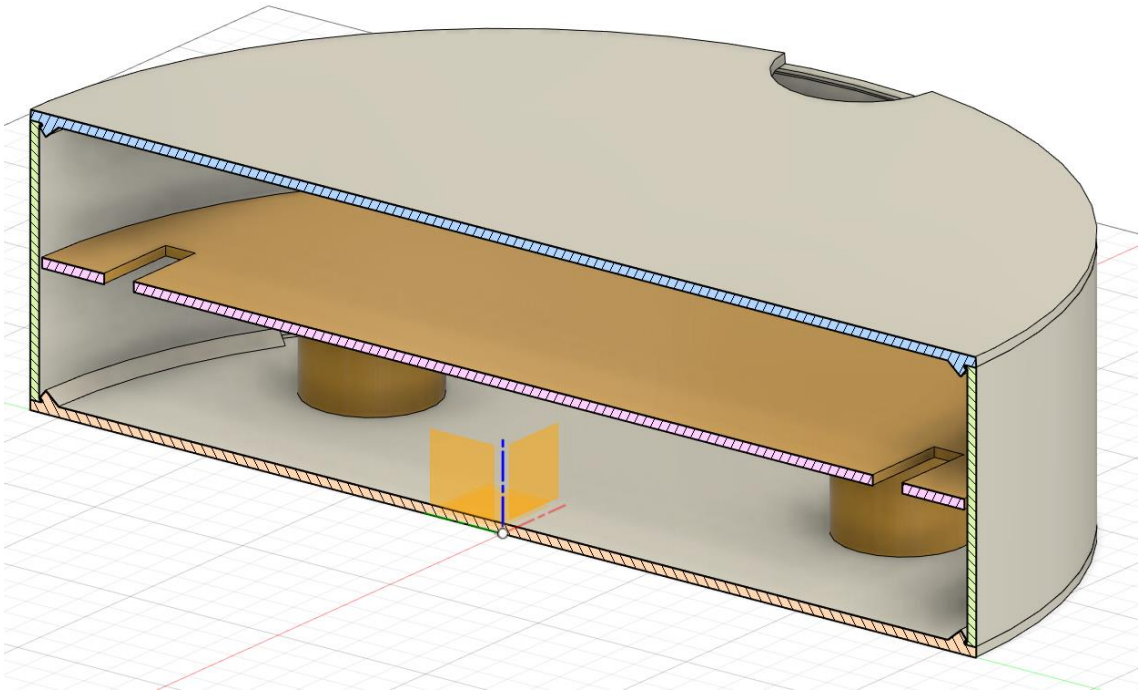
1. **2nd Caller**

4. **Chassis for Coaster**
   a. 3D print chassis for each part of the system

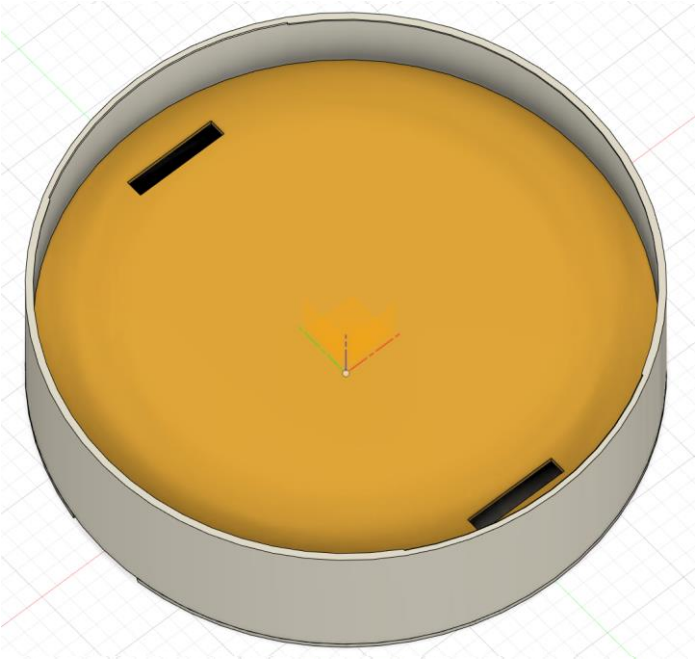
5. **PCB Plans**
   a. Motion - Further optimizations
   b. User input - Add 5V and LED

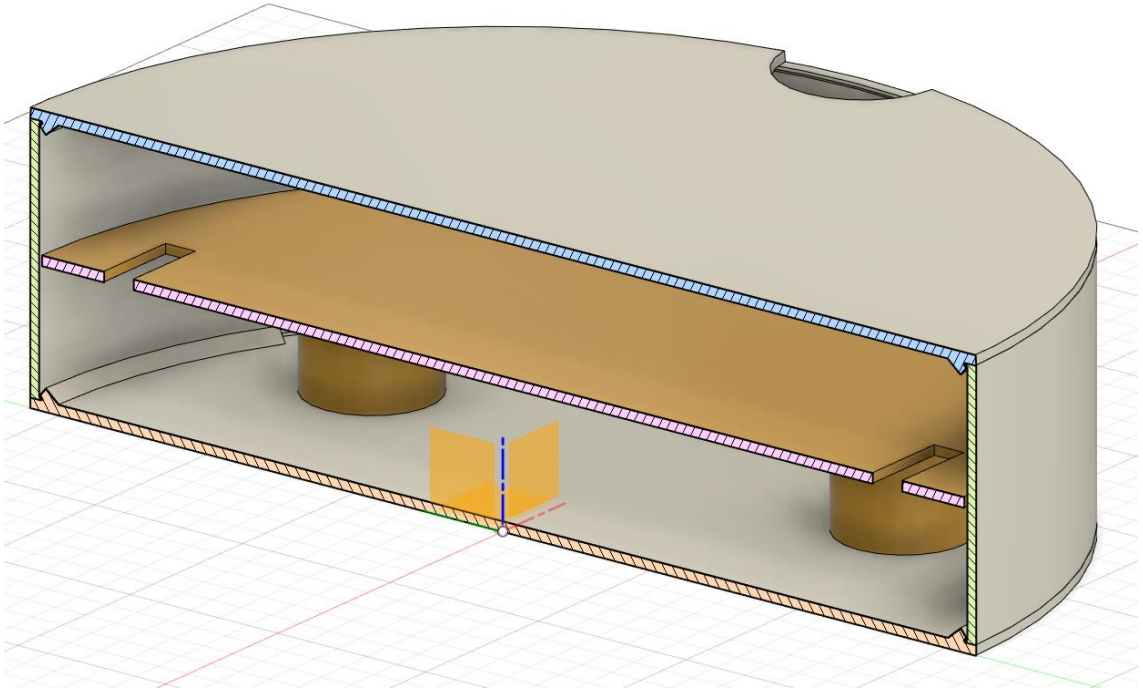
Design: Chassis (FPR)



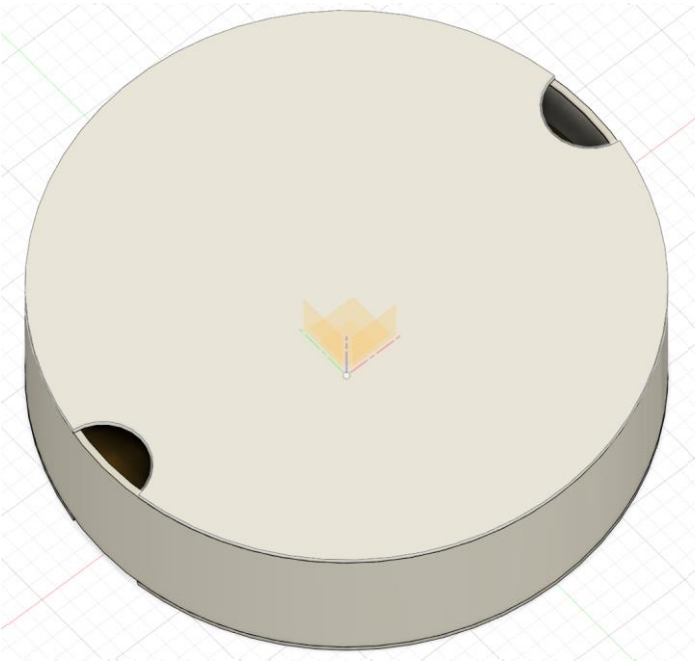


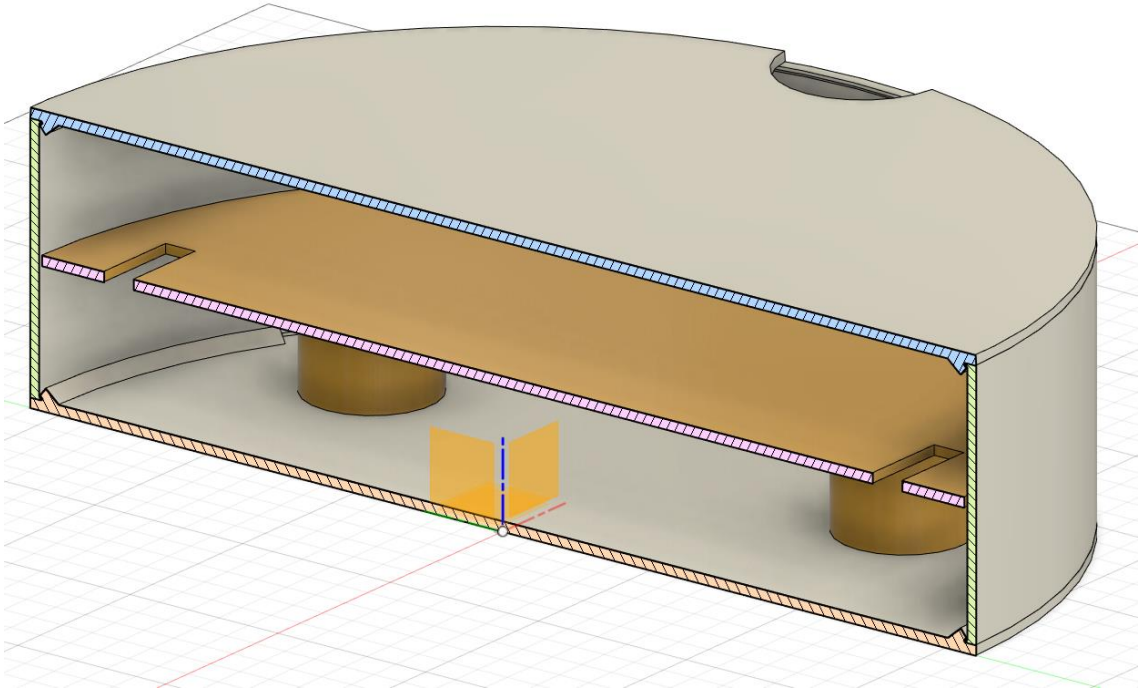
Design: Chassis (FPR)





Design: Chassis (FPR)





SDP Budget Used

Purchasing List		
Item Name	Link	Price $ (w/Tax & Ship)
UWB DIY Starter Kit	Tindie	74.98
2 x Pico W	Digikey	12
Gina PCB1	JLC	22.05
Gina PCBs 2	JLC	15.44
Omar PCB1	JLC	6.5
Omar PCB2	JLC	6.76
6 Dof Sensors	Amazon	15.95
10x HC05	Amazon	37.98
3D Filament	Amazon	38.99
New Wheels	Adafruit	28.17
$258.82		

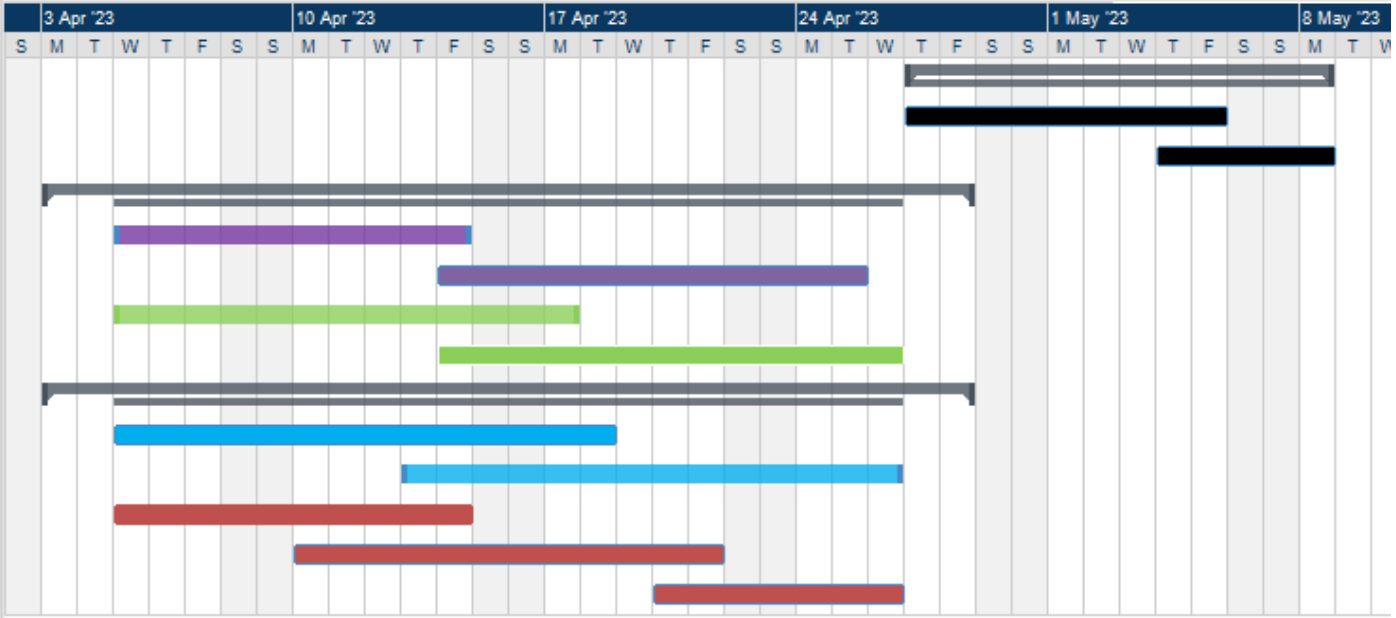
Development Cost - per set of Brains, 1 Coaster, and 2 Callers

Coaster List		
Item Name	Source	Price $ (with tax and shipping)
Qorvo UWB Kit	Qorvo	170
3 x Pico W	Wouter	18
PCBs	JLCPCB	10
Sensors	M5	25
Sensors	M5	25
8x RCR132a	Amazon	27
Motors and Wheels	Amazon	50
Raspberry Pi 4	M5	100
Misc. Hardware	Misc	30
Lipo	M5	20
$475.00		

Gantt Chart

Soham | Omar | Redwan | Gina

	Task Name
1	**Presentation Week**
2	Slides and Presentation
3	Poster
4	**Hardware**
5	3D Model for Coaster
6	3D Model for Caller
7	Updating PCB
8	2nd Caller Setup
9	**Software**
10	Implementing More Sensors
11	Rerouting Algorithms
12	Smoother Setup & Communications
13	Move some applications away from Pi
14	Integrating 2nd Caller



Videos

Straight Shot:

https://drive.google.com/file/d/141u86p3lzujfCnuI5bPtXzJ-QYSLi3L_/view?usp=sharing

Obstacle Avoidance:

https://drive.google.com/file/d/14BBHoazjFHobzXkXQCEPxUCetjSHFlHe/view?usp=sharing

QUESTIONS & ANSWERS

University of Massachusetts Amherst