Drink Dispenser Robot

ECE 445 Design Document - Fall 2025

Group #23

Andrew Jung, Ethan Cao, Megan Cheng

Professor: Rakesh Kumar

TA: Frey Zhao

Contents

1 Introduction

1.1 Problem

Too often, we're tired or distracted and put off getting a drink of water. Those small delays add up, leaving us dehydrated and drained without even realizing it. In social situations, the host tends to be responsible for providing beverages to a group. This can pull them out of the moment and distract them from the conversation. Additionally, many users may get tired of drinking water and would prefer flavored drinks. Dehydration impacts focus, energy, and overall well-being, yet it happens so easily in our daily lives. Also, the additional fun aspect of having a personal bartender and a mini robot bringing your drinks encourages you to keep drinking, staying hydrated.

1.2 Solution

The solution is to create a drink delivery ecosystem that seamlessly connects a mobile robot, a drink dispenser hub, and a callback system for user interaction. The robot is responsible for navigating the environment, locating both the dispenser hub and the user's cup, and safely transporting beverages. As this is an at-home bar-like experience, it is assumed that the robot will be on a table setting with no complex obstacles. The hub acts as the liquid source, with the ability to dispense various ratios of drinks, allowing users to choose of mixed drink. The callback system enables the user to request service and call the robot without needing to approach the dispenser themselves, and allows the user to customize the drink ratio they want. Together, the robot docks at the hub, the hub dispenses the desired beverage into the cup, and the robot delivers the drink back to the user. Specifically, the robot integrates multiple subsystems. Its detection subsystem includes a bumper with left/right detection switches, cliff-detection sensors capable of recognizing drops greater than 1 inch, a load cell to detect the weight of the cup and IR detectors for locating both the dispenser and coasters. These components are connected to the robot's microcontroller ESP32's GPIO pins. On the hub side, the pump subsystem controls two liquid channels via motor-driven pumps, with encoders ensuring accurate dispensing. A precision docking subsystem with IR transmitters to aligns the robot under the dispenser nozzle to prevent spillage. Like the robot, the hub uses an ESP32 microcontroller to receive the ratio of the drink from the callback subsystem. Finally, the callback subsystem is the user control system, allowing the user to call the robot and adjust a dial for a custom ratio. It is in the shape of a coaster so that it can blend in with the environment flawlessly instead of an awkward black box. It is surrounded by a modulated IR detector so that the robot locates the user and travels with the cup. This ecosystem ensures hydration is made convenient, safe, and customizable with minimal user effort.

1.3 Visual Aid

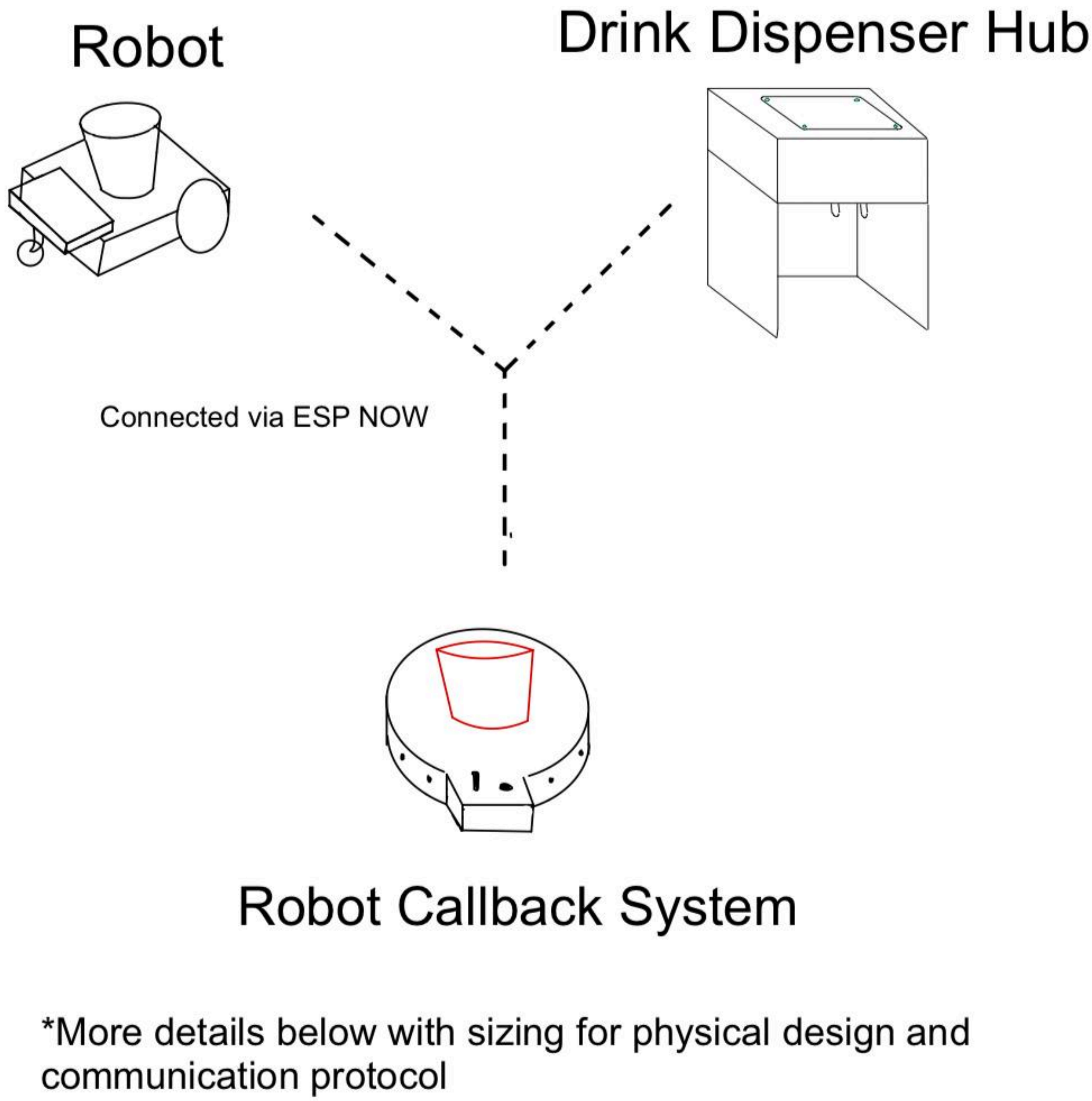

Figure 1: Visual Aid for Dispenser Project

1.4 High-Level Requirements

To consider our project successful, our system must fulfill the following:

1. The robot will retrieve the drink within 90 seconds on a clear tabletop without any obstacles. The introduction of obstacles shall not inhibit functionality but may increase service time.
2. The dispenser shall be able to fill a standard red solo cup. No more than 1 mL of liquid will be spilled on the table for every 5 drink retrieval cycles.
3. The robot will not get stuck on the dispensing station/coaster or fall off of the table.

2 Design

2.1 Physical Design

2.1.1 Robot

For our robot chassis, we will be using a combination of 3D printed and off-the-shelf parts. The chassis consists of two driven wheels with encoders and a caster wheel attached to a load cell (Figure 2). We will be using an N20 motor with an integrated encoder for the driven wheels and 3D print the caster wheel. A 3D printed cupholder will be used to help the user place the cup in a repeatable manner. Without a cup, this robot will approximately be 160mm long, 110mm wide, and 90mm tall.

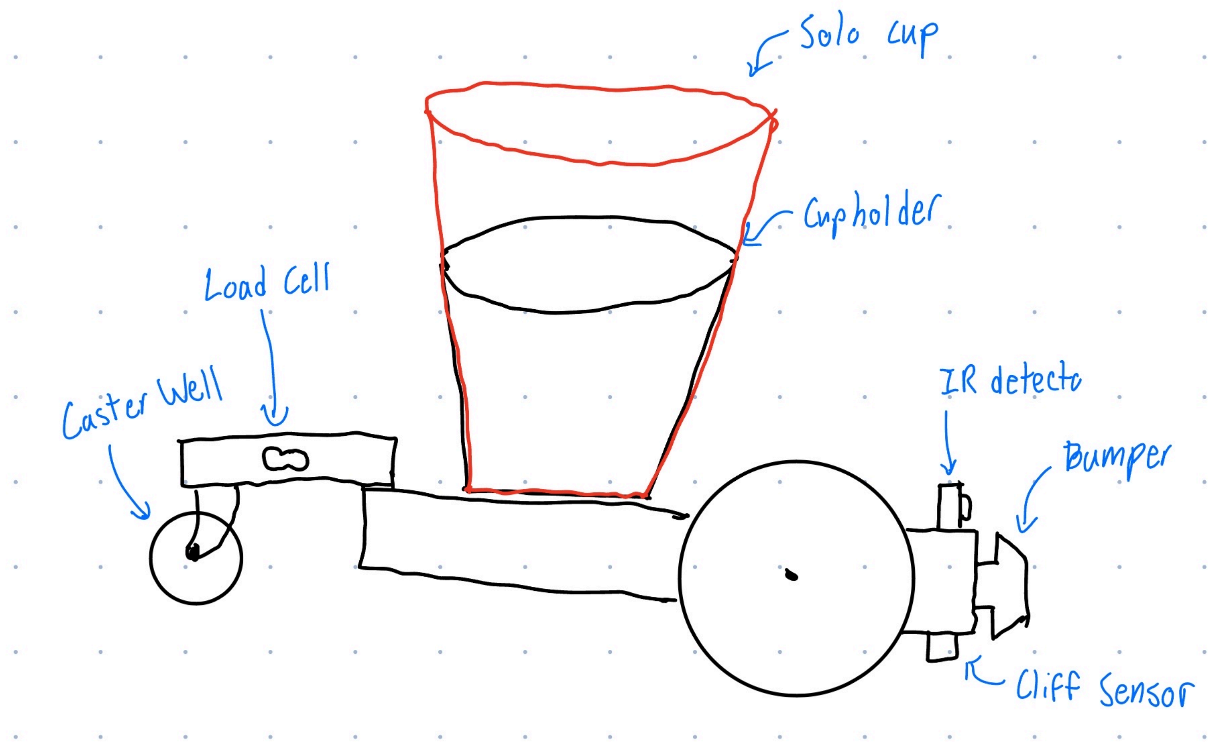

Figure 2: Sketch of Physical Design of Robot

The bumper mechanism will utilize a mechanical keyboard switch. This was chosen because of its easy mounting and the ability to experiment with a wide variety of spring weights and activation distances.

2.1.2 Dispenser

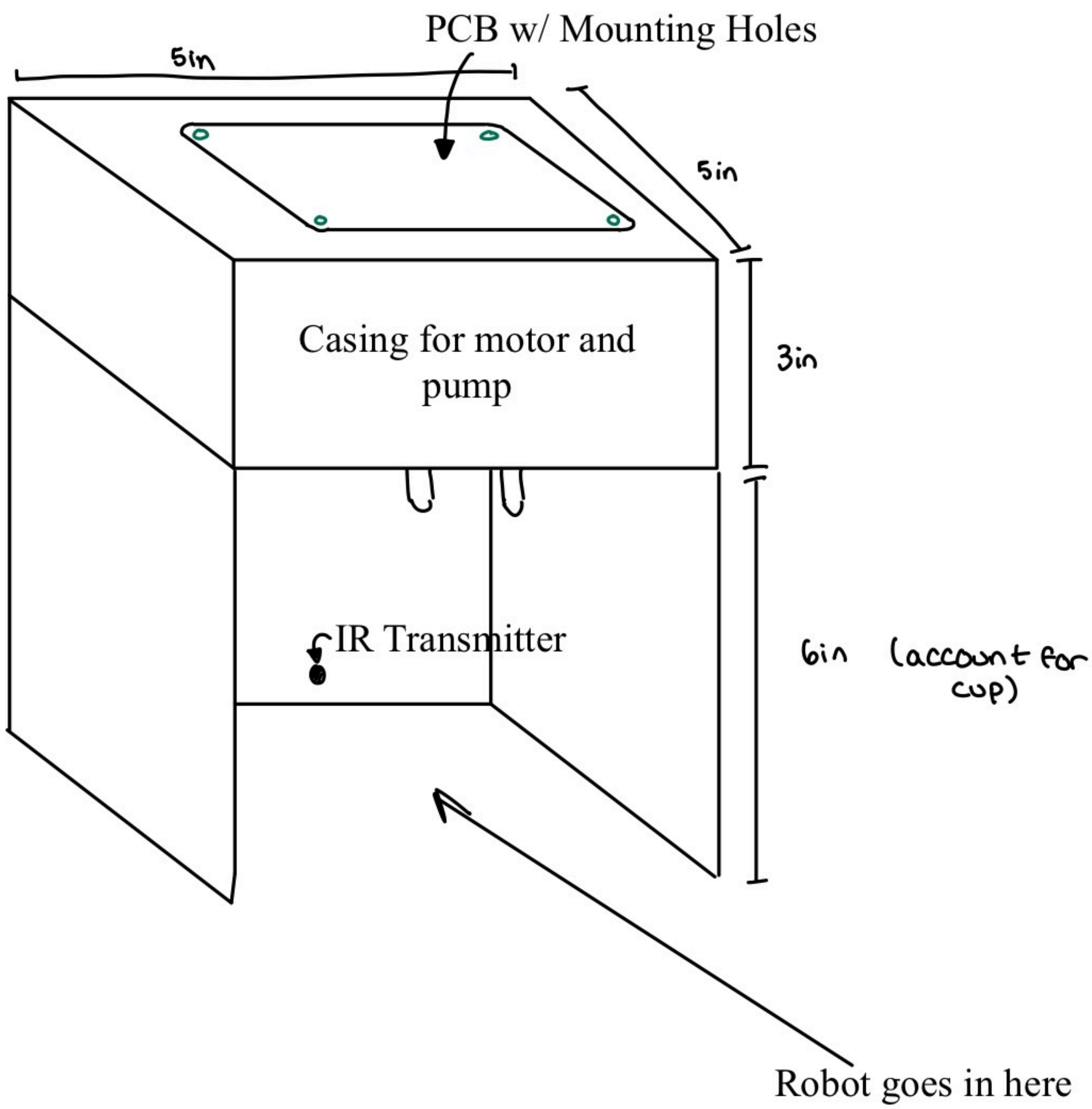

Figure 3: Sketch of Physical Design of Dispenser

The drink dispenser will be a combination of different 3D printed parts meant to be wide enough and tall enough to fit the robot underneath so that the cup can be held under the tubing. The storage for the liquid that will be used of the drink dispenser will be held behind the drink dispenser hub in food safe containers. The liquid will also only touch food safe materials like food safe tubing.

2.1.3 Coaster

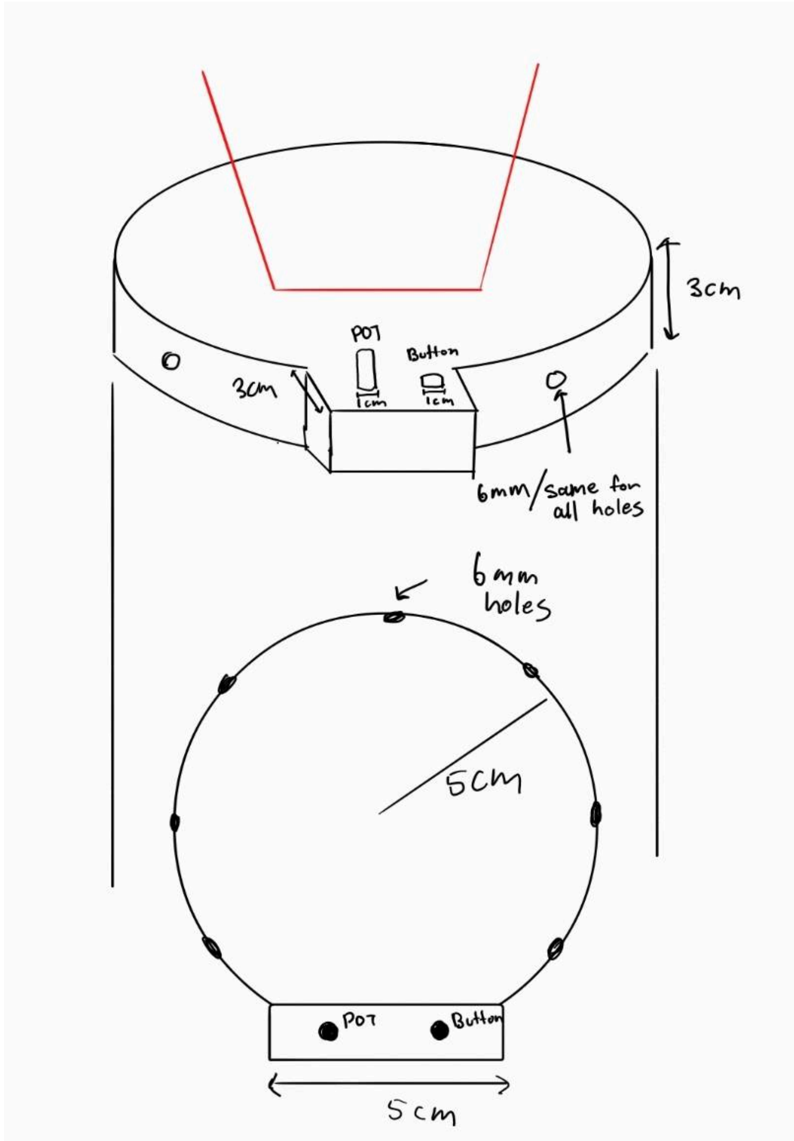

Figure 4: Sketch of Physical Design of Coaster

The coaster will contain the user input side where they can have the dial and button as the user input. There will be 7 IR transmitters surrounding the coaster so that it isn't affected by the orientation.

2.2 Block Diagram

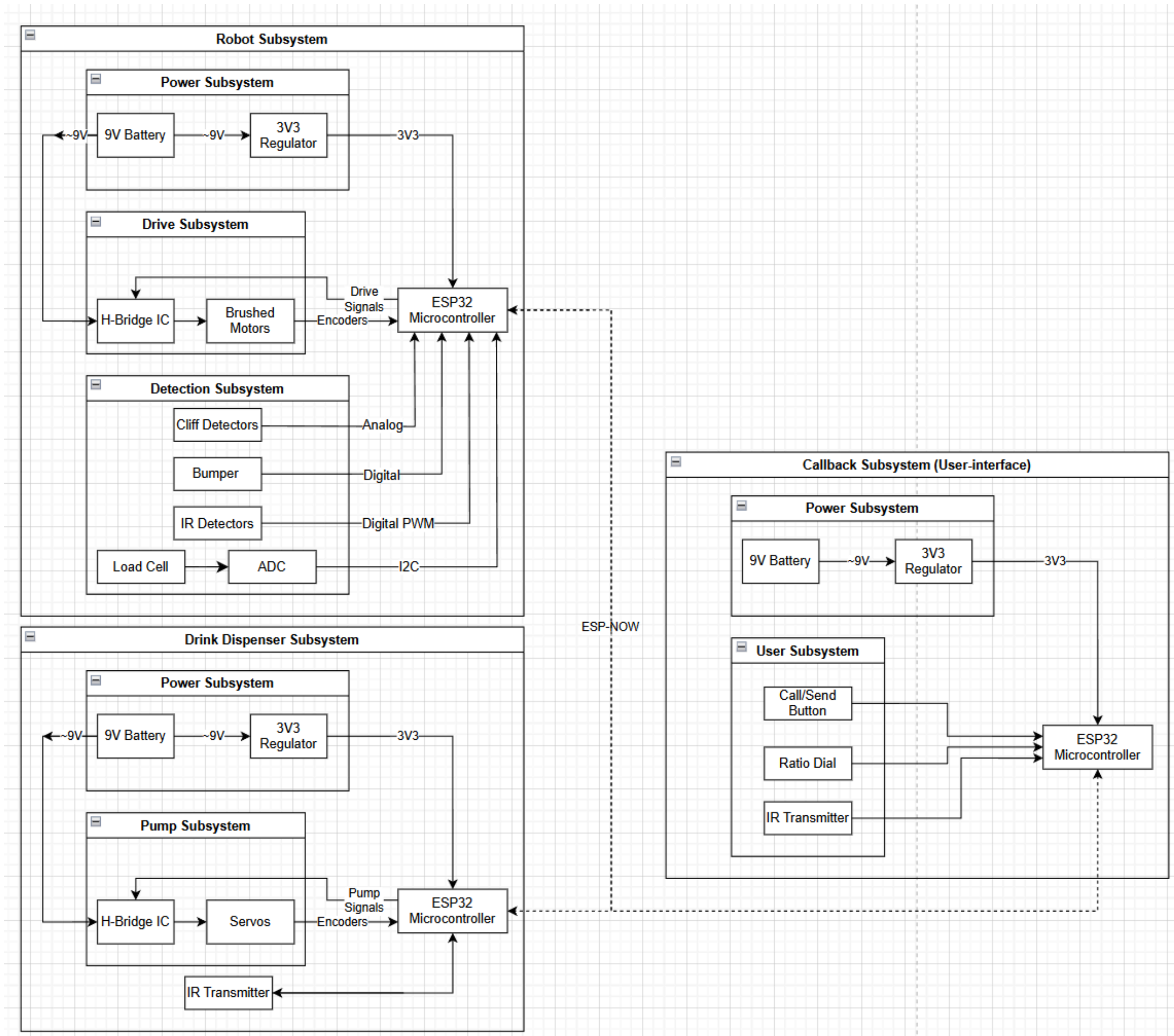

Figure 5: System Block Diagram

2.3 Functional Overview & Block Diagram Requirements

2.3.1 Robot

Detection

The detection subsystem contains all of the sensors required to navigate the environment effectively. The sensors will be housed on a separate PCB and communicate with the microcontroller over I2C. The bumper will use a bar and 2 switches to determine if the robot is contacting any object. The cliff detection system will use downward facing distance sensors to determine if the robot is approaching the edge of the table. The IR detectors will help the robot find the location of the dispenser and coasters.

Requirement	Verification
The bumper system shall be able to detect a collision anywhere on the front surface of the robot and the side which the collision occurred on (front left or front right)	Place objects in the path of the robot, ensure that the robot stops and paths around the object
The cliff detection system shall detect any drop greater than 1 inch	Place a coaster outside of the table which the robot is on and send a request. Ensure that the robot does not fall off the table while attempting to reach the coaster
The IR detector shall determine the intensity of the IR light received	Place a coaster to the front left/right of the robot, check that the robot turns towards the coaster

Drive

The drive subsystem will use an H-bridge IC to drive 2 brushed motors with integrated encoders. It will communicate wheel positions back to the microcontroller using a quadrature signal and receive commands from the microcontroller via PWM and direction signals.

Requirement	Verification
The drive subsystem shall drive at over 1 feet/second unloaded and 0.5 feet/second loaded with a cup	Write test code which drives forward for a foot and backwards for the same distance. Use a stopwatch to time how long this takes and calculate speed. Conduct this for both the loaded and unloaded case

Power

The power subsystem will power the robot using a 9V battery and regulate the battery's voltage down to 3.3 volts for our logic devices.

Requirement	Verification
The power system shall provide up to 500mA continuously and 750mA bursts at 3.3 volts.	Disconnect the buck converter from the rest of the robot circuit. Use a resistive load from the lab to test maximum current.
The power system shall protect against overcurrent events using a fuse	Disconnect the buck converter from the rest of the robot circuit. Short the 3.3V output to ground and connect power.
The battery shall be replaceable with basic hand tools	Provide a user with a screwdriver and allen key set. Have the user replace the battery

Microcontroller

The microcontroller will send signals to all of the other subsystems on the robot and communicate with the dispenser and coaster over the ESP-NOW protocol.

Requirement	Verification
The microcontroller shall not exceed 80 degrees Celsius	Use a thermal probe to probe the microcontroller while the robot is running. Ensure that maximum temperature does not exceed limit

2.3.2 Dispenser

Pump

The pump subsystem contains two motors that will drive two pump systems. This system features an H-bridge IC that controls the two motors. The pump ensures that the liquid dispenses in a timely manner.

Requirement	Verification
The pump subsystems shall not consume current above the maximum current capability of the h-bridge, motor, or battery	Use a multimeter to measure the current draw during the operation. Verify that the peak current is within the rated tolerance specifications of the H-bridge IC, motor, and battery under max conditions.
Both of the two pumps in the subsystem must dispense liquid at the same speed	Operate both pumps simultaneously and collect the liquid over a fixed time interval. Measure the volume of liquid from each pump and verify that it is within the tolerance and not spilling from the cup.

Power

This subsystem has the same design and requirements as the Robot Power subsystem.

Docking

The docking subsystem will contain an IR LED, which the Robot's docking subsystem will use.

Requirement	Verification
When the Robot microcontroller wants to find the Drink Dispenser system. The microcontroller will communicate and run the IR LED at a 38 kHZ frequency	Probe the IR LED control signal with an oscilloscope to confirm the 38 kHz square wave output when triggered by the microcontroller

Microcontroller

This subsystem has the same design and requirements as the Robot Microcontroller Subsystem.

2.3.3 Coaster

User Subsystem

Allows the user to call and send the robot with custom ratios of drink. Uses ESP32 to communicate with the robot given the inputs from User Subsystem. Has call and send buttons that will call the robot if currently not at user, and send the robot to the hub if robot is at the user. The ratio dial will customize the ratio between the water and mixer.

Requirement	Verification
The Call and Send buttons shall register a single press with no false triggers.	Rapidly press the button 10 times. If the ESP reports "presses: 10" it is successful. If it overshoots, debounce must be increased as there are false trigger
Reported mix ratio shall be accurate to ±3 %FS after 1-point calibration.	Mechanically check 10 evenly spaced dial positions vs computed ratio. Pass: error ≤ ±3 %FS each point.

IR Subsystem

IR transmitter will allow the robot to detect the user by emitting modulated IR at 38kHz.

Requirement	Verification
Carrier frequency 38 kHz ± 1%; duty cycle 33% ± 5%.	Scope the MOSFET gate waveform for 60 s. Pass: f and duty within limits; jitter < 2%.

2.4 Hardware Design

2.4.1 Robot

IR Detection

We will be using the TSSP58P38 IR receiver module for detecting the IR beacons provided by the dispenser and coasters. According to the TSSP58P38 datasheet [1], the detector needs to be powered with 3.3V and will provide an active low output when an 38kHz modulated IR source is detected. Additionally, the duration of this active low output will reflect the intensity of the received light. We will use 3 detectors pointed in different directions to locate IR beacons.

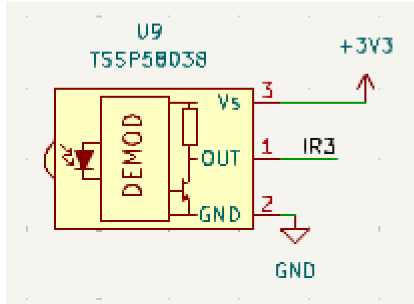

Figure 6: IR Detector Circuit

Cliff Detection

The cliff detection subsystem will use a QRE1113 [2] sensor which provides an infrared transmitter and receiver in a single package. The QRE1113 operates on the same principles as the TSSP58P38 but provides us with convenient packaging.

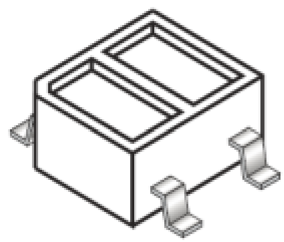

Figure 7: QRE1113 Sensor

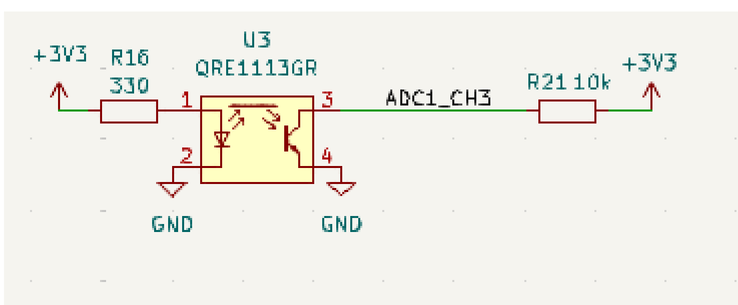

Figure 8: Cliff Detection Circuit

Bumper

The bumper circuit will provide a connector to all us to attach and utilize any switch. This will provide mechanical flexibility. Debouncing will occur in software.

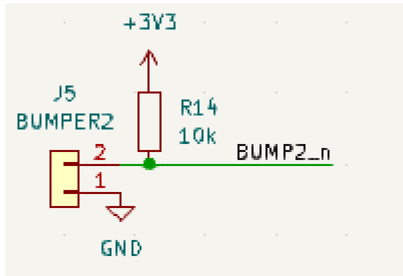

Figure 9: Bumper Button Circuit

Load Cell

We will be using a whetstone bridge load cell and a NAU7802SGI [3] ADC to detect the weight on top of the robot. We will use the load cell to determine whether or not a cup is placed on top of the robot as well as how full the cup is.

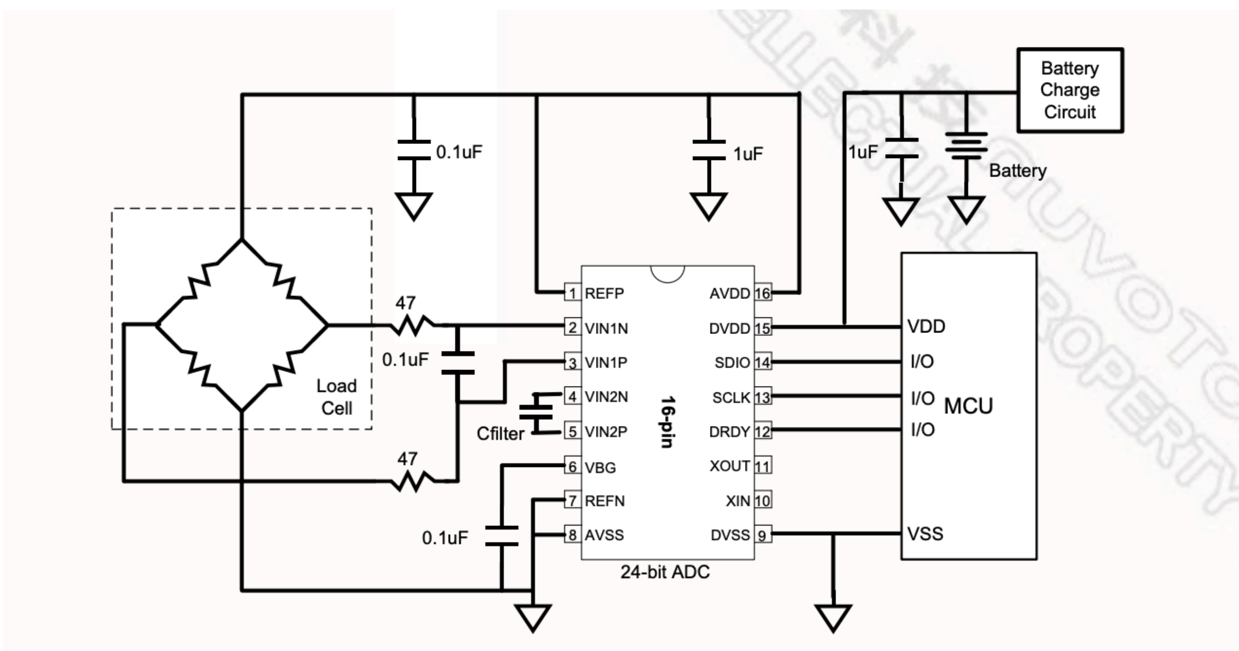

Figure 10: Load Cell Circuit

Drive

The drive subsystem utilizes an L298N motor driver IC [4] which uses the following inputs to control up to two motors

MW.15	Power SO	Name	Function
1, 15	2, 19	Sense A, Sense B	Between this pin and ground is connected the sense resistor to control the current of the load.
2, 3	4, 5	Out 1, Out 2	Outputs of the bridge A; the current that flows through the load connected between these two pins is monitored at pin 1.
4	6	V_S	Supply voltage for the power output stages. Anon-inductive 100nF capacitor must be connected between this pin and ground.
5, 7	7, 9	Input 1, Input 2	TTL compatible inputs of the bridge A.
6, 11	8, 14	Enable A, Enable B	TTL compatible enable input: the L state disables the bridge A (enable A) and/or the bridge B (enable B).
8	1, 10, 11, 20	GND	Ground.
9	12	VSS	Supply voltage for the logic blocks. A 100nF capacitor must be connected between this pin and ground.
10, 12	13, 15	Input 3, Input 4	TTL compatible inputs of the bridge B.
13, 14	16, 17	Out 3, Out 4	Outputs of the bridge B. The current that flows through the load connected between these two pins is monitored at pin 15.

The following table provides information on how input combinations control motor direction and braking where C and D are the bridge inputs.

Inputs		Function
V_{en} = H	C = H; D = L	Forward
	C = L; D = H	Reverse
	C = D	Fast motor stop
V_{en} = L	C = X; D = X	Free running motor stop

Although the L298N operates on TTL logic (5V), the logic high and low input values provided by the data sheet suggest that the 3.3V outputs of our ESP32 should be sufficient to control our h-bridge.

V_{iL}	Input low voltage (pins 5, 7, 10, 12)		–0.3		1.5	V
V_{iH}	Input high voltage (pins 5, 7, 10, 12)		2.3		V_{SS}	V

V_{enL}	Enable low voltage (pins 6, 11)		–0.3		1.5	V
V_{enH}	Enable high voltage (pins 6, 11)		2.3		V_{SS}	V

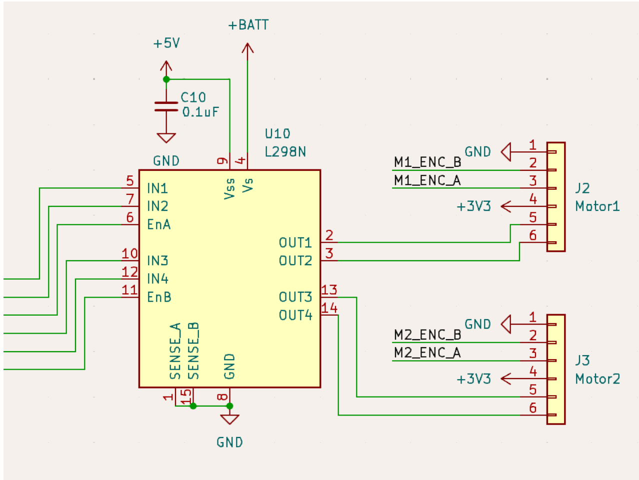

Figure 8: Cliff Detection Circuit

Power

The power subsystem will be used to power the servos and the microcontroller. TI's Power Designer software [10] was used to assist in creating the buck converter. Additionally, a 5V LDO is used to power the TTL logic of the H-bridge.

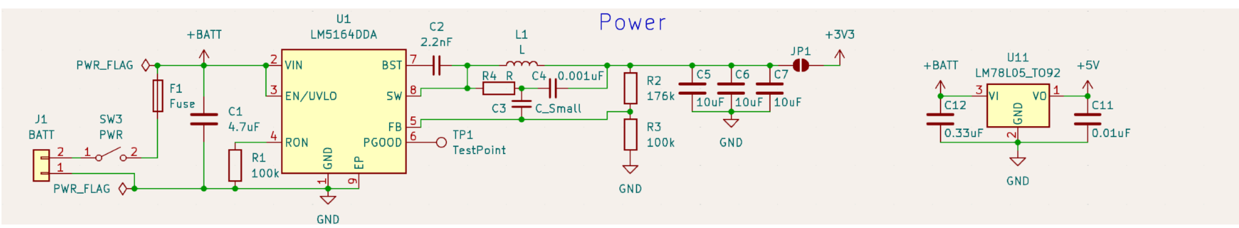

Figure 9: Power Supply Circuit

Microcontroller

The microcontroller will send signals to all of the other subsystems on the robot and communicate with the dispenser and coaster over the ESP-NOW protocol. The programming circuit is based off of the circuit provided to us on the ECE 445 course website [5].

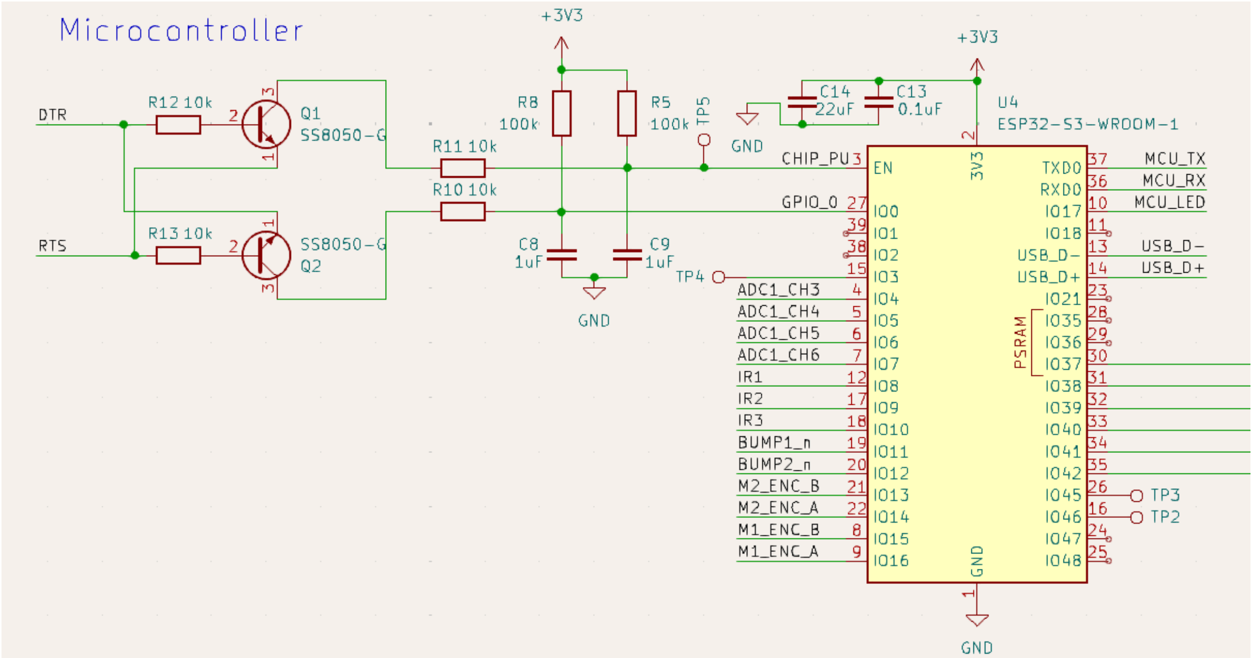

Figure 10: Microcontroller and Programming Circuit

2.4.2 Dispenser

IR Transmitter

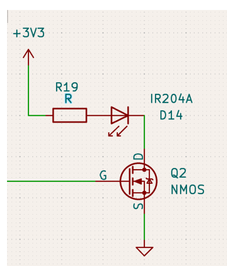

Figure 11: IR LED Circuit

The IR LED will be controlled by the microcontroller and will function at a frequency of 38 kHz. The number of IR LEDs will vary depending on which device is utilizing the LEDs. The coaster will have 7 IR LEDs, and the Drink Dispenser will have 1 LED.

Power

This will be the same as the Power Hardware design for the robot. The requirement is that there will be 3.3V output to the microcontroller and 5V output for the H-bridge system.

Microcontroller

The microcontroller for the drink dispenser system will send signals to the IR LEDs and the H-bridge that is connected to the two motors for the pumps. Similar to the robot microcontroller, the drink dispenser microcontroller will communicate via ESP-NOW with the other microcontrollers. It will let the robot know when the drink is finished dispensing and the robot will let the drink dispenser know when the IR transmitter needs to be turned on.

2.4.3 Coaster

The user interface is made of the button and potentiometer. The ESP reads the state of the button and the current value of potentiometer. Using the ESP, the IR transmitters are modulated at 38kHz for the receiver. There are 7 in total so that the leds can go around the coaster so the user has directional freedom when placing the coaster.

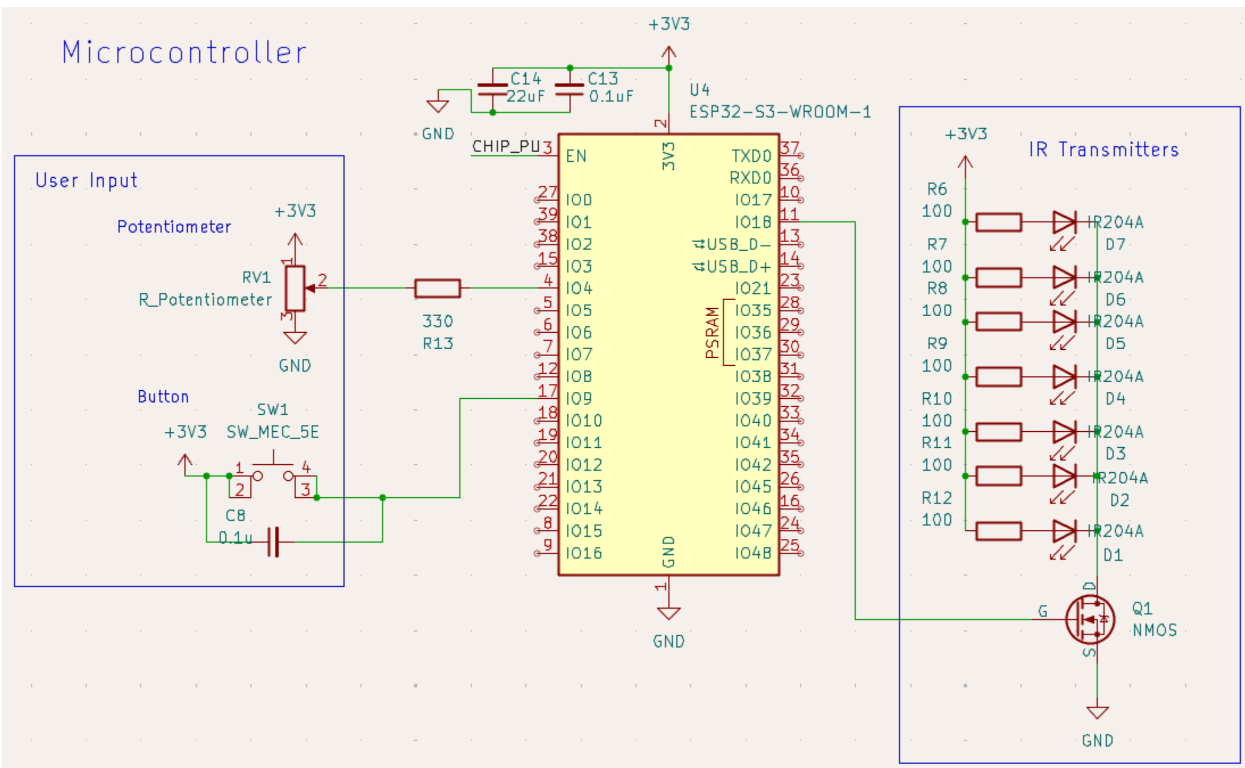

Figure 12: Coaster Circuit

1. 10K Potentiometer
    a. Connected to the GPIO4 as it is a stable analog reading
    b. 330Ω Resistor limits charge spikes into the ADC sample cap
2. Tactile Switch
    a. Connected to the GPIO4 for digital reading
    b. Input 1&2 and output 3&4 connected for stronger reading

3. IR Transmitters
    a. Connected to GPIO18 for clean, flexible output for a timed waveform
    b. IR204A has the Vf of 1.2V at 20mA meaning that each IR should be connected to about 100Ω
        i. R = (3.3 -1.2 - 0.05)/0.02 = 100
    c. It is connected to the NMOS for modulation from ESP

2.5 Software Design

2.5.1 Robot FSM

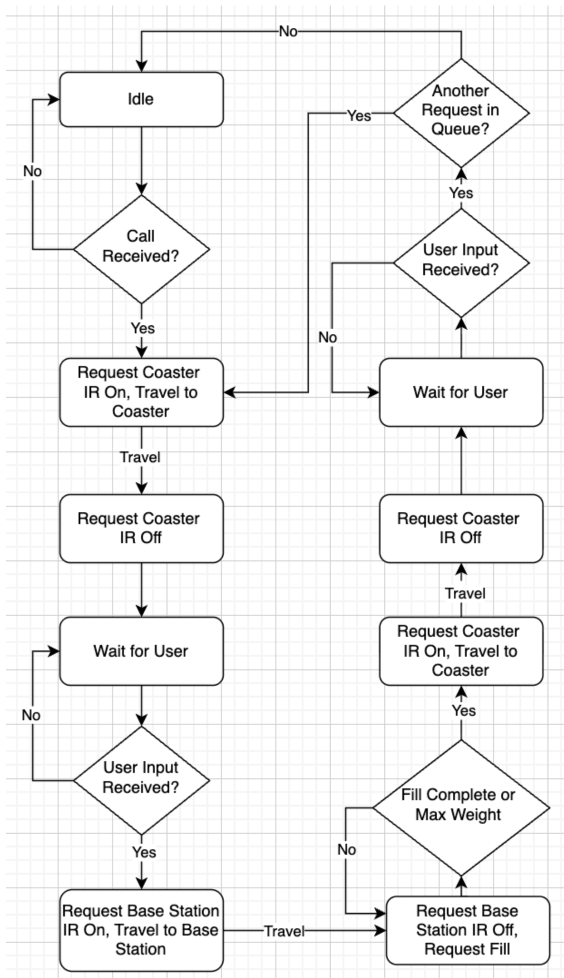

Figure 13: Robot FSM

The robot's FSM determines the next action it should take and what commands it should send. Additionally, the robot will have a request queue which will queue user requests and service them in order.

2.5.2 Dispenser FSM

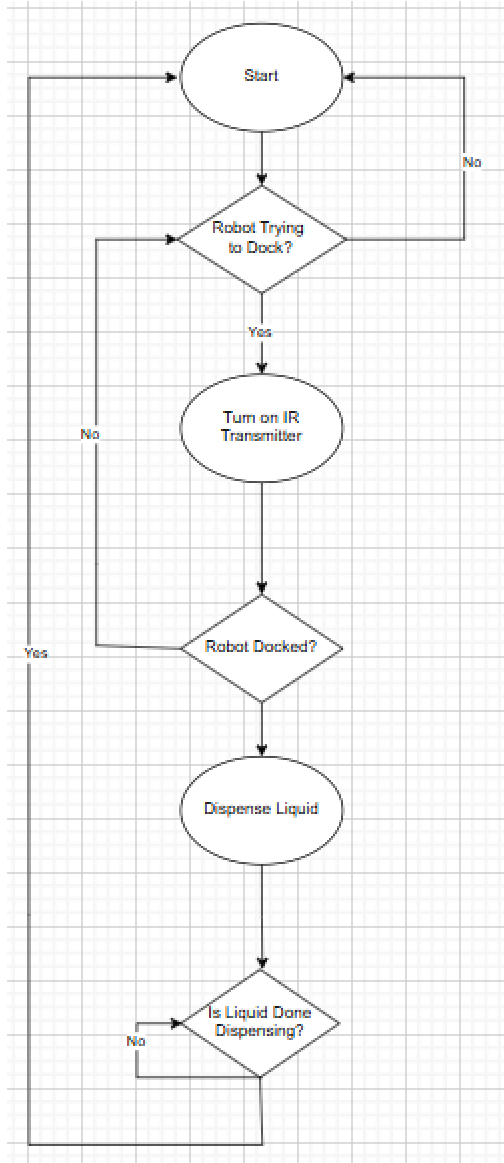

Figure 14: Dispenser FSM

The drink dispenser subsystem turns on the IR transmitter when the robot is trying to locate the drink dock. After locating the drink dispenser, the robot will dock and start dispensing liquid. Then there will be a timer to tell the drink dispenser how long it will take to fill the cup and split it between the two motors.

2.5.3 Coaster FSM

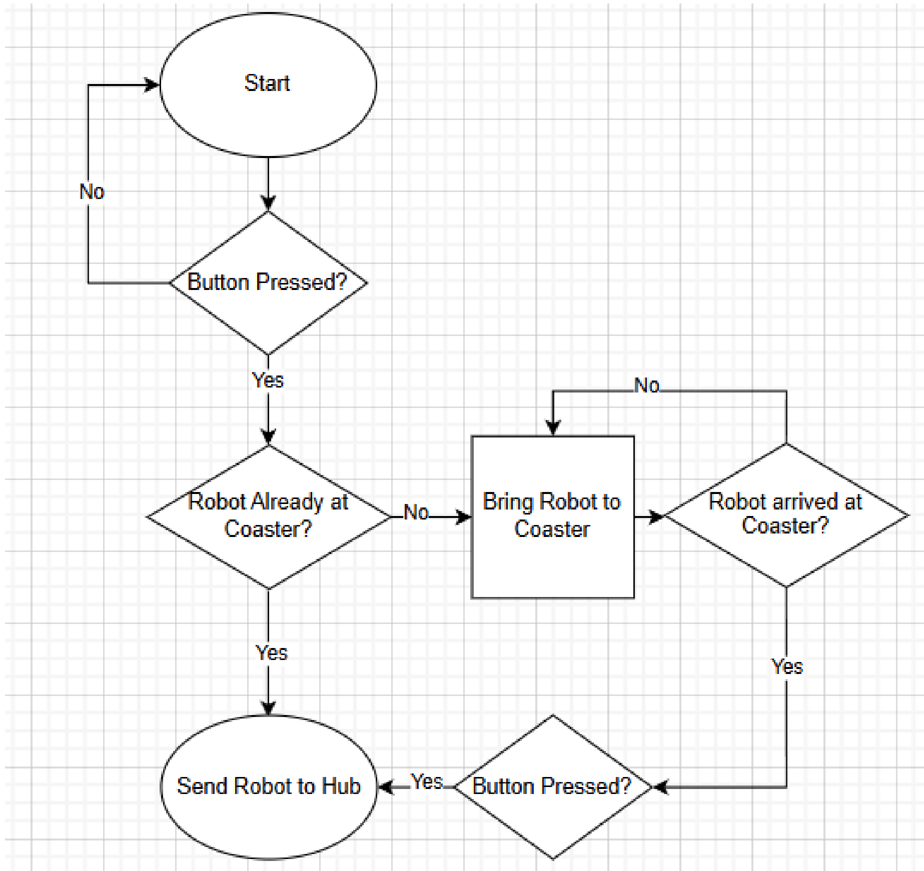

Figure 15: Coaster FSM

The callback subsystem makes sure that the robot is brought to the coaster once the button is pressed. If the robot is already at the coaster, then the button will send the robot. If it is not, it will initially call the robot, then the button needs to be pressed again to send the robot to the hub.

2.5.4 System Communication

All components of the system will broadcast messages to each other and it will be up to the individual devices to decide if the message is relevant to it. Each message will consist of one byte and will follow the following convention. Additionally, each device will have its own device identifier with 0 being the dispensing station and 1-7 being different coasters. This allows us to support up to 7 coasters.

Sender ID [2]	Message ID [2]	Message Data [4]

Message	Sender ID	Message ID	Message Data
Robot Status	00	00	IsTraveling[1], Location ID[3]
Dispenser Status	01	00	X, X, IsReady, IRState
Coaster Status	10	00	CoasterID [3], IRState
Robot Call	10	01	CoasterID[3], X
Fill Cup	00	01	X, X, X, Stop
Fill Status	01	01	X, X, X, Done
Request IR State	00	11	DeviceID[3], IRState

Additionally, there will be a special message for sending the robot to the dispenser and transmitting the user's requested ratio. This message provides more data bits for higher resolution.

11	Message Data [6]

2.6 Commercial Component Selection

For our motors, we decided to go with the commonly used N20 form factor. This would allow use to find cheap motors with integrated encoders and have lots of options for shaft lengths and gear reductions.

2.7 Tolerance Analysis

Robot Subsystem

Component	Peak Voltage	Peak Current
ESP32-S3-WROOM-1	3.3V	500 mA bursts (Wi-Fi TX), ~80–150 mA typical under load
Motor	5 V	300 mA per motor Total peak (600 mA)

Drink Dispenser Subsystem

Component	Peak Voltage	Peak Current
ESP32-S3-WROOM-1	3.3V	500 mA bursts (Wi-Fi TX), ~80–150 mA typical under load
IR Transmitter IR204-A	3.3 V	20 mA per IR Total peak (7 IR) ≈ 140 mA
Motor	5 V	300 mA per motor Total peak (600 mA)

Callback Subsystem

Component	Peak Voltage	Peak Current
ESP32-S3-WROOM-1	3.3V	500 mA bursts (Wi-Fi TX), ~80–150 mA typical under load
IR Transmitter IR204-A	3.3 V	20 mA per IR Total peak (7 IR) ≈ 140 mA
Tactile switch	3.3 V (Logic High)	0 mA idle 70 µA when pressed
Potentiometer 10 kΩ	3.3 V	≪1 mA

2.8 Cost Analysis

The labor costs were estimated by taking the average yearly salary of an electrical engineering graduate and calculating the average hourly rate. Then it was estimated that there would be about 6 hours of work done a week for 14 weeks so that would result in around a $10k labor cost with a parts cost of around $89. This results in a cost of around $10170 spent for the production of this drink dispenser hub

Subsystem	Part	Part Number	Amount	Price	Column 1	Procure
All	Microcontroller	ESP32-S3 WROOM 2	3	0	0	ECE Service Shop
All	LED	General LED	8	0	0	ECE Service Shop
All	Buck Converter	LM5164DDA	4	4.06	16.24	Digikey
All	Shipping Cost	General Cost	1	7	7	Digikey
Cup	Potentiometer	General Potentiometer	1	0	0	ECE Service Shop
Cup	Buttons	General Button	6	0	0	ECE Service Shop
Dispenser + Cup	IR Transmitter	IR204A	8	0	0	ECE Service Shop
Robot	Robot Motor	Amazon	2	20	40	Amazon
Robot	IR Reviever	TSSP58038	5	1.84	9.2	Digikey
Robot	Proximity/Color Sensor	QRE1113GR	5	1.09	5.45	Digikey
Robot	Load Cell	Amazon	1	8	8	Amazon
Robot	Load Cell Amplifier	NAU7802SGI	2	1.62	3.24	Digikey
Robot + Dispenser	Motor Driver	L298N	2	0	0	ECE Service Shop
Robot + Dispenser	Motor Header	Generic 6 Position Berg	4	0	0	ECE Service Shop
All	Fixed Output Linear Regulator	LM78L05	4	0		ECE Service Shop
All	Resistors	General Resistors	20	0	0	ECE Service Shop
All	Capacitors	General Capacitor	20	0	0	ECE Service Shop
All	NPN/PNP		10	0	0	ECE Service Shop
				Total Parts Cost	89.13	
Labor Costs						
Team Members	$ per hour	Hours Worked per week	Weeks Worked	Total Cost		
Ethan Cao	40	6	14	3360		
Megan Cheng	40	6	14	3360		
Andrew Jung	40	6	14	3360		
			Total Labor Costs	10080		
			Total Costs	10169.13		

2.9 Schedule

All
Andrew
Ethan
Megan

Week	Tasks / Deadlines (summary)
Week 1 (8/25)	Decide each subsystem everyone is in charge of
Week 2 (9/1)	Decide the method robot will locate coaster Decide the pump method Decide how to do cliff detection Decide Drive train
Week 3 (9/8)	Finish Project approval
Week 4 (9/15)	Refine proposal Finish Block Diagram for each subsystem Team Contract
Week 5 (9/22)	Prepare for proposal review
Week 6 (9/29)	Build Breadboard Demo sign-up opens Start PCB design

Week 7 (10/6)	Breadboard Demo First-round PCBWay orders
Week 8 (10/13)	Design Document Teamwork Evaluation I Second-round PCBWay orders
Week 9 (10/20)	Prepare for Breadboard demo 2 Check that IR detection intensity works All ESPs should be wirelessly communicating
Week 10 (10/27)	Breadboard Demo 2
Week 11 (11/3)	Third-round PCBWay orders Adjust PID for Robot tracking IR
Week 12 (11/10)	Fourth-round PCBWay order Adjust PID for Robot tracking IR
Week 13 (11/17)	Mock demo during weekly TA meeting Team Contract Finish up project
Week 14 (11/24)	Finish up project
Week 15 (12/1)	Final Demo Mock Presentation Extra Credit Video
Week 16 (12/8)	Final Presentation Final Papers Lab Checkout Lab Notebook

2.10 Risk Analysis

The Drink-Dispensing Robot is low-voltage but still presents risks. Liquid ingress is the top hazard. We isolate fluid paths from electronics, but the liquid ingress due to wear and tear or damaged product is unavoidable. Therefore, we should suggest service frequencies for critical components. Users may try unsafe fluids; we clearly label water/room-temp mixers only and choose food-safe, room-temp materials. Electrical and battery risks are mitigated with a certified AC adapter, SELV rails, no exposed conductors, fusing/overcurrent protection. Heat from regulators/MOSFETs/motors is controlled via thermal vias/heatsinks. Mechanical hazards are reduced by speed limits, bumper/cliff sensors, guided slow docking, and the small size of the robot.

3 Ethics and Safety

The primary ethical concern of the project is to avoid harm to the end user according to ACM general ethical principles [6] sections 1.2: Avoid Harm and 1.3: Be Honest. Due to the nature of our project, users will likely ingest liquids which have interacted with our project. Therefore, we need to ensure our components are food safe and that we warn our users of any potential contamination. We will need to provide functions for the user to clean the dispenser and flush any potentially harmful substances out. We will also need to prevent misuse by labeling what liquids are allowed in the dispenser station.

4 References

[1] Vishay Semiconductors. TSSP58P38: IR Detector for Mid-Range Proximity Sensor. Rev. 2.0, 21 Sept. 2023, Vishay Semiconductors, www.vishay.com/docs/82476/tssp58p38.pdf

[2] ON Semiconductor. QRE1113: Miniature Reflective Object Sensor (Datasheet). onsemi, Dec. 2024, www.onsemi.com/pdf/datasheet/qre1113-d.pdf

[3] Nuvoton Technology Corporation. NAU7802 Data Sheet V1.7. Nuvoton, 7 Oct. 2010, www.nuvoton.com/export/resource-files/NAU7802%20Data%20Sheet%20V1.7.pdf

[4] STMicroelectronics. L298: Dual Full-Bridge Driver (Datasheet). STMicroelectronics, [no date], www.st.com/resource/en/datasheet/l298.pdf

[5] “ESP32 Example / ECE 445 Wiki.” ECE 445 – Senior Design Laboratory, University of Illinois, courses.grainger.illinois.edu/ece445/wiki/#/esp32_example/index.

[6] ACM. “Code of Ethics.” Association for Computing Machinery, www.acm.org/code-of-ethics

[7] DroneBot Workshop. “IR Remotes Revisited – 2023.” DroneBot Workshop, dronebotworkshop.com/ir-remotes/.

[8] Everlight Electronics Co., Ltd. EVER-S A0003161355-1 Specsheet. Mouser, www.mouser.com/catalog/specsheets/everlight_ever-s-a0003161355-1.pdf

[9] ZombieCat. Peristaltic Pump for Hobby Servo or Motor. Thingiverse, thingiverse.com/thing:5170737.

[10] TI. “SDP Tool (Webench).” TI Webench, Texas Instruments, webench.ti.com/appinfo/webench/scripts/SDP.cgi?ID=3912C433EB0814E8.