AUTOMATED DRIVEWAY SALT DISPENSER

By

Candy Gao

Mayura Kulkarni

Arya Tyagi

Final Report for ECE 445, Senior Design, Spring 2024

TA: Koushik Udayachandran

01 May 2024

Project No. 31

Abstract

This paper describes the design, building, testing, and results of the Automated Driveway Salt Dispenser project. The device is able to efficiently distribute salt throughout a 2 m x 2 m driveway and come to an automatic stop once it has completed dispensing. The project uses a transmitter and receiver system to detect the edges of the driveway and allow the car to follow a zig-zag path throughout the area. The paper will describe the motivation of the project, overview, design, and testing of each subsystem, and conclude with a discussion of accomplishments and future work to improve the project.

Contents

1. Introduction

1.1 Problem

Freezing rain and extremely low temperatures during the winter season cause slippery driveways and sidewalks which make it difficult for people to walk and drive. Current methods of dispensing salt in these areas are done manually and are not very efficient. This is because salt is thrown randomly across these areas which results in wastage of salt and sometimes less salt in very icy areas. Also, these methods have safety risks as ice covering these areas makes it harder for people to walk to dispense the salt. It also increases the burden on homeowners to manually salt their driveways.

1.2 Solution

To solve this issue, we have created a fully autonomous salt dispenser. Our solution is a self-driving car that would dispense salt evenly across driveways and sidewalks. This would solve the issue of having slippery ice on the sidewalk/driveway when trying to leave your house. Also, allowing the car to only dispense salt on driveways and sidewalks will help to reduce the amount of salt that is wasted from randomly dispensing salt manually. The dispenser consists of two main components. The first component is the autonomous steering of the car which will prevent the car from driving out of bounds, such as on the grass or outside of the driveway. Also, the second component is the dispensing of the salt using motors to allow the salt to be spread evenly across the surface.

1.3 High-Level Requirements

The high-level requirements for our project include:

1. The robot is able to distribute the salt onto a 2m x 2m driveway in under 10 min.
2. The robot is able to detect the edge of the driveway within 10 in of the edge.
3. The robot is able to come to a complete stop once it has reached the end of the driveway in under 10 sec.

To achieve these requirements, our project is divided into five subsystems including the sensing, power, control, drivetrain, and salt dispenser subsystems. The sensing subsystem is responsible for ensuring the car stays within the boundary of the driveway using a transmitter and receiver system. The drivetrain subsystem controls the motion of the car as it goes straight and turns to follow a zig-zag path throughout the driveway. The salt dispenser subsystem allows for the salt to be evenly distributed throughout the driveway. The control subsystem includes an on/off button and the microcontroller which sends signals to the other subsystems to control the operation of the car. The power subsystem consists of the power supplies that are needed to power the other subsystems. A visual aid is depicted in Figure 1.1 and a block diagram of the different subsystems of the project is displayed in Figure 1.2.

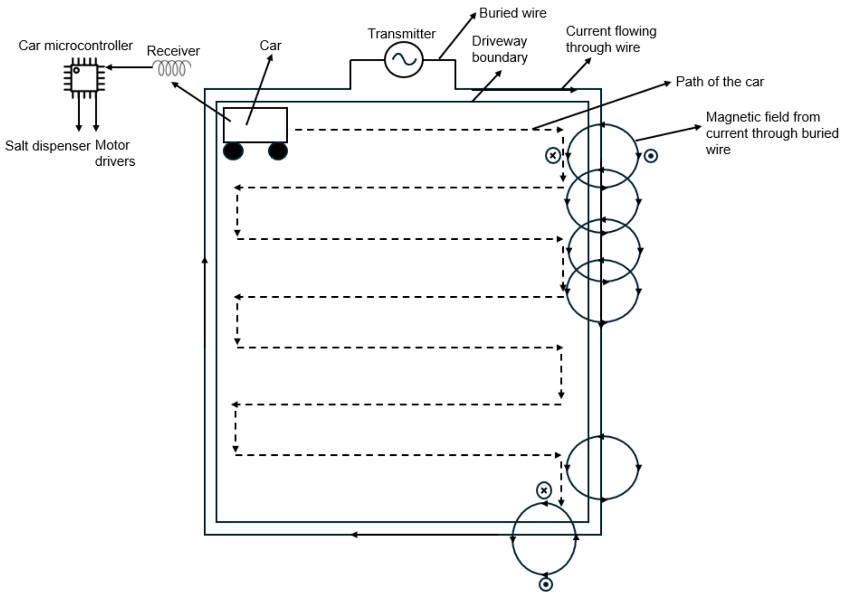

Figure 1.1. Visual Aid

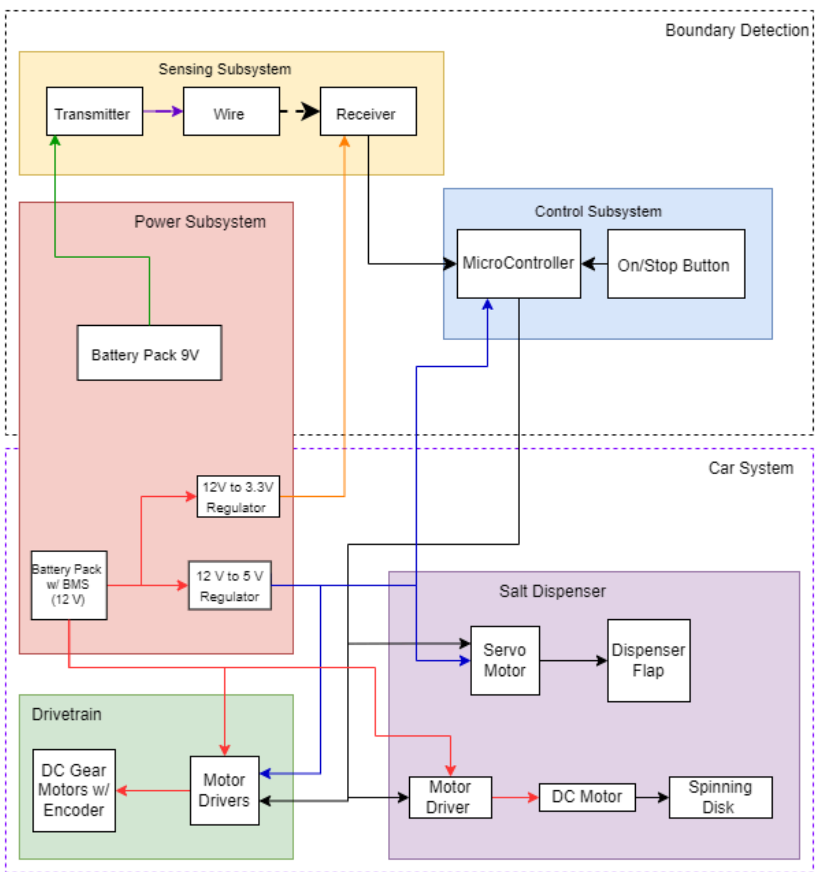

Figure 1.2. Block diagram

2 Design

2.1 Sensing Subsystem

2.1.1 Overview

The sensing subsystem consists of the transmitter, wire, and receiver. The transmitter will generate an AC signal that travels through a boundary wire around the driveway. This alternating signal will create a changing magnetic field surrounding the wire, which the car receiver will be able to detect once it has come in proximity to the edge and thus indicates that the car will need to start turning.

2.1.2 Design Procedure

As our idea for boundary detection was inspired by invisible pet fences, we first started by researching how the transmitter and receiver in such products on the market work. However, due to the complexity and cost, we were unable to simplify or incorporate a ready-made product into our project. Therefore as an easy and low-cost solution, we decided to use a coil wire as our receiver to detect the changing magnetic field.

For the transmitter, the AC signal was generated using the TLC555CP timer IC that can generate a square wave voltage output between 0 V and around the supply voltage, Vdd, which was set to 9 V [1]. The frequency of the wave was set to about 10 kHz, which is within the range of frequencies typically used for some invisible pet fences [2, 3]. Also, to create a sinusoidal wave that is differentiable and can be detected by the receiver, an RC low-pass filter was implemented to the output of the TLC555CP IC to filter out the higher frequencies.

For the receiver, the induced voltage generated from the coil will be first amplified using a negative feedback OpAmp which is then outputted to the ADC input of the microcontroller. To ensure that the bipolar induced voltage signal will be in the range of the unipolar ADC input, a rail splitter circuit is used to create a virtual ground between 0 V and AVcc. During the initial prototyping of the receiver circuit, we noticed that the output was very noisy, therefore a bandpass filter was added so that only frequencies around 10 kHz are kept.

In addition to the coil that will be placed in front of the car for edge detection, two more coils are added on the side. This is designed to help with end-of-driveway detection in which when the car is in a corner, two of the coils will see an increase in induced voltage. However, because the ATmega328P only has two ADC inputs, the two side coil induced voltages will be added together and outputted to one ADC input. Figure 2.1 shows the schematic of the receiver circuit design for the front coil only, but the side coils will also follow the same schematic design.

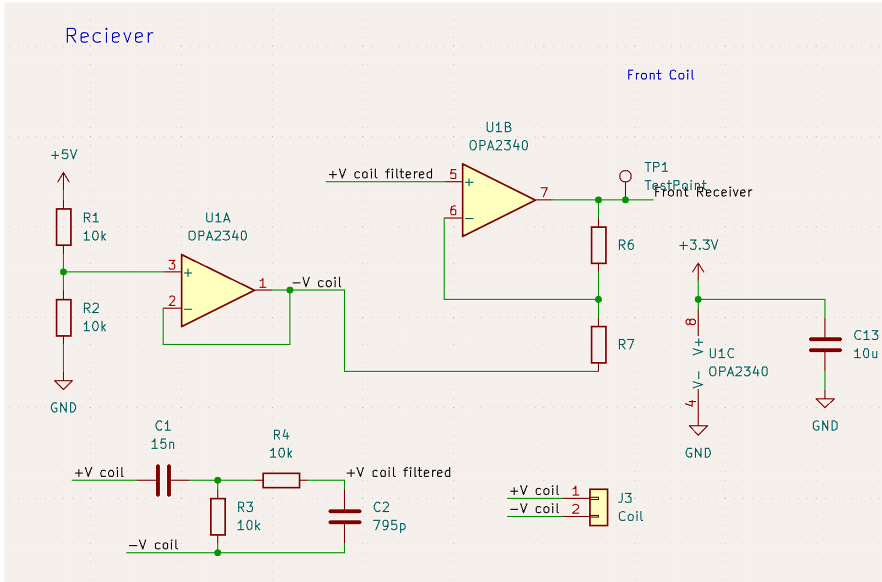

Figure 2.1. Receiver Schematic for the front coil

2.1.3 Design Details

To design the transmitter circuit, the voltage output of the transmitter should have a frequency of about 10 kHz. The astable mode of operation of the TLC555CP timer IC can be used to operate as an oscillator or a multivibrator with a set frequency [1, 4, 5]. This frequency is determined by the resistor values R1 and R2 and the capacitor value C2, as depicted in the schematic of the transmitter in Figure 2.2.

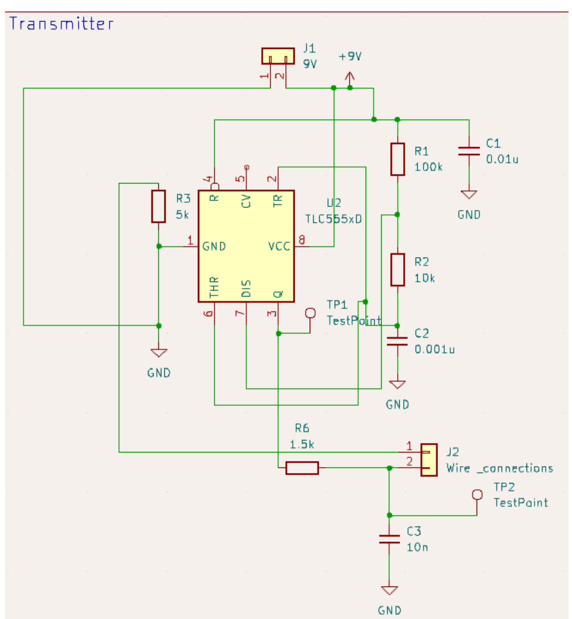

Figure 2.2. Schematic of the transmitter

The equation to find the frequency is displayed below:

$$f = \frac{1.44}{(R1 + 2R2)(C2)} \quad (2.1.1)$$

To achieve a frequency close to 10 kHz, we initially used two 330 Ω resistors for R1 and R2 and one 0.1 μF capacitor for C2. This design resulted in a frequency of about 9696.97 Hz, which is close to 10 kHz. We later changed these components to achieve a greater frequency to increase the induced voltage generated by the receiver coil and match the SMD component footprints that we already had on our PCB board. Thus, a 100 kΩ resistor was used for R1, a 10 kΩ resistor was used for R2, and a 0.001 μF capacitor was used for C2. This allows for a frequency of about 12 kHz. To design the RC low-pass filter circuit to generate a sinusoidal voltage waveform from the square wave output of the TLC555CP, we chose the resistor and capacitor values to filter out the frequencies that are above 10 kHz [6-8]. The cutoff frequency of an RC circuit is given by the equation:

$$f_{cutoff} = \frac{1}{2\pi RC} \quad (2.1.2)$$

where *R* is the value of the resistor and *C* is the value of the capacitor. To achieve a cutoff frequency of about 10 kHz, we chose a 1.5 kΩ resistor for *R* and a 10 nF capacitor for C as these components were already available in the lab. Thus, the cutoff frequency was about 10.61 kHz. These components for the RC circuit were kept the same for the new design of the transmitter as using larger resistors resulted in a lower peak-to-peak voltage of the transmitter output. Also, to achieve a higher peak-to-peak voltage of the transmitter to allow for a greater induced voltage generated by the receiver, the supply voltage of the TLC555CP was increased to 12 V. The TLC555CP can have a supply voltage within the range of 2 V to 15 V. Therefore, a 12 V battery was used to power the TLC555CP as it was still within the range and less expensive.

To design the amplifier circuit, we first need to figure out what the minimum value the amplified signal needs to be and the actual induced voltage we can get from the coil.

The ADC converter on the ATmega328P converts the analog signal into a 10-bit digital value through successive approximation where the minimal value is 0 (GND) and the maximum value is Vref. The conversion is calculated as

$$ADC = \frac{V_{IN} \cdot 1024}{V_{REF}} \quad (2.1.3)$$

where V_{IN} is the output of the receiver and V_{REF} is the selected voltage reference which is set to 5 V. To ensure that we can get a consistent and accurate reading, we want to make sure that the receiver can output a signal of at least 1 V peak-to-peak which will result in a reading difference of at least 205 using Equation (2.1.13).

We then can estimate the induced voltage using Maxwell's equations [9-12]:

$$\varepsilon = N \times \frac{d\phi}{dt} \tag{2.1.4}$$

$$\phi = B \times A \tag{2.1.3}$$

$$B = \frac{\mu_0 Iwire}{2\pi r} \quad where \ I_{wire} = Isin(\omega t) \tag{2.1.5}$$

Combining Equations (2.1.3), (2.1.4), and (2.1.5), we find that the induced voltage is given by:

$$|\varepsilon| = \frac{N\mu_0 AI\omega}{2\pi r} \tag{2.1.6}$$

where *N* is the number of turns in the coil, μ_0 is the permeability of free space, *A* is the cross-sectional area of the coil, *I* is the magnitude of the current traveling through the wire, ω is the frequency of the current traveling through the wire, and *r* is the distance between the wire and the coil.

To estimate the induced voltage, we assumed a coil of radius 3 cm with 10 turns. Since we are using a transmitter with a 9 V battery, we expected to get an output peak-to-peak voltage of around 9 V therefore an amplitude of 4.5 V. Assuming that we will be using a 24 gauge copper wire of length 8m (perimeter of a 2 m x 2 m area), which has a resistance of around 0.7 Ω. Then using Ohm's law, we can find the magnitude of the current through the wire. Plugging these values into Equation (2.1.6) we find the induced voltage of the coil at 10 in from the wire to be:

$$|\varepsilon| = \frac{N\mu_0 AI\omega}{2\pi r} = \frac{(10)(4\pi \times 10^{-7})(3 \times 10^{-2})^2(\frac{4.5}{0.7})(2\pi(10000))}{2\pi(0.254)} = 0.009\ V$$

Using LTspice simulations, we were able to see that using resistor values of 500 kΩ and 1 kΩ produced a peak-to-peak output voltage of 1.74 V which satisfies our needs.

However, when we prototyped this circuit onto a breadboard, we were only able to get an output voltage with a peak-to-peak of 0.111 V. We believe that the difference between the theoretical calculations and the measured values is a result of the various interferences in the setup. The theoretical calculations only take into account the magnetic field created by the changing current in the wire, while in our actual setup, the wires on the board and even the surrounding lab equipment can all affect how much induced voltage is produced.

While we used the estimated values as a starting point, we found the actual design values for the receiver using mainly trial and error by modifying both the coil turns and sizes and the gain of the amplifier. As shown in Figure 2.3, we found that with a coil of 50 turns and a radius of around

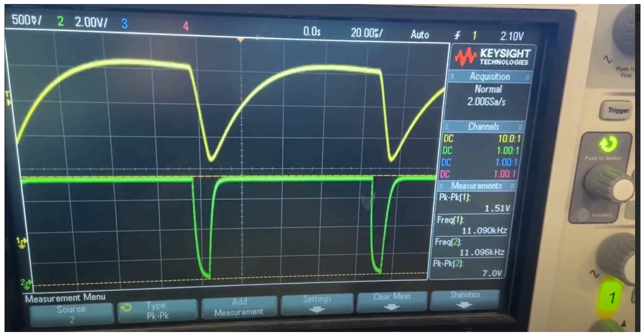

Figure 2.3. Receiver (yellow) and Transmitter (green) output

3 cm, we were able to get a receiver output with a peak-to-peak voltage of 1.51 V if we had a gain of nearly 3000.

2.2 Power Subsystem

2.2.1 Overview

The power subsystem is responsible for supplying power to all the subsystems. For the power subsystem, a 9 V battery pack will be used to power the transmitter (seen on the transmitter schematic). Another 12 V battery pack will be used to power all the other electrical components on the car (the microcontroller, receiver, drivetrain, and salt dispenser).

2.2.2 Design Procedure

For the low voltage components such as the motor drivers, microcontroller, and servo motor, a 5 V voltage converter was used. We selected the BD50FC0FP-E2 which has a fixed output current of 1A. The maximum current drawn from these three components is around 0.9 A which is under the current output from the converter. Since the OpAmp used in the receiver circuit needs to be powered at 3.3 V, the AZ1117CD-3.3TRG1 3.3 V converter was used. This part was chosen because it was readily available and also satisfied the current requirements for the OpAmp. Lastly, we calculated the max current drawn from all the components to be 3.14 A and the max power needed to be around 31.3 W which led us to select the 12 V 5200 mAh 58 Wh rechargeable Li-ion battery.

2.2.3 Design Details

Our power subsystem design changed once we decided to move the drivetrain onto the Arduino board. We needed a way to power the Arduino so we first tried powering it using the 12 V battery. While it can be powered at 12 V, since we are also using the 5 V output from the Arduino, this can overheat the 5 V converter on the board which can in turn damage the 12 V power supply input. Due to this, we decided to power it using the 9 V battery we had with us. As the PCB board now only had the salt dispenser and control subsystems, we only needed a 5 V power source and we were able to use the Arduino for that and no longer needed a 5 V converter. Figure 2.4 shows the power connection on the car during our final demo.

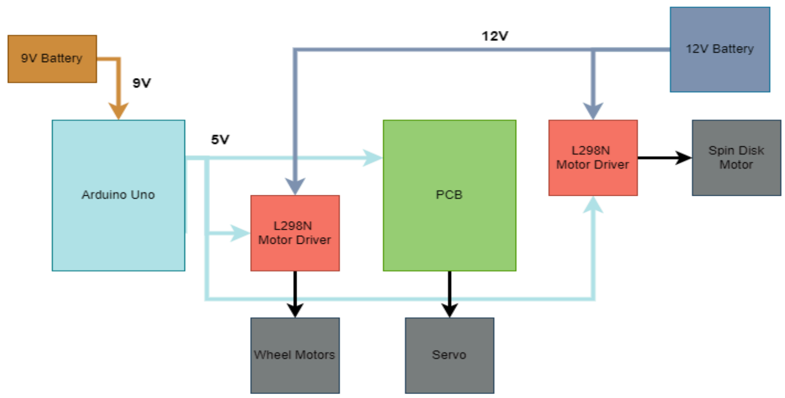

Figure 2.4. Power Connections on the car

2.3 Control Subsystem

2.3.1 Overview

The control subsystem will consist of the board microcontroller and an on/stop button for the car. Once the button has been pressed, the microcontroller will send an ON signal to start up the salt dispenser and drivetrain subsystem. If the car has reached a boundary, the microcontroller will receive a signal from the car and will then communicate with the drivetrain motors to indicate which direction to turn. The microcontroller will ensure that the car has traveled throughout the entire area of the driveway by stopping only when it has reached two consecutive boundaries of the driveway. Once the car has completed dispensing the salt throughout the driveway or if the on/ off button has been pressed again (in case of an emergency stop), the microcontroller will send an OFF signal to the salt dispenser and drivetrain subsystem.

2.3.2 Design Procedure

For our control subsystem, we didn't deviate from our original plan. We used the Atmega328P and a push button for our on/off control. Toward the end of our project, we were discussing using a switch rather than a button but ultimately decided to stick with our original button plan. We did have to use another microcontroller on an Arduino to fully satisfy all of our requirements.

2.3.3 Design Details

The PCB Atmega328P was used to control the on/off signal from the button, the servo dispenser flap, and the spinning disk. The Arduino was used to control the drivetrain. Though we were able to get the drivetrain programmed on the PCB, we decided to switch to the Arduino because the PID control system that was functional on the Arduino did not work on the PCB. This was very difficult to debug since the PCB didn't have many test points for us to monitor the encoder pulses. We also had to create a way for the microcontroller on our PCB to communicate with the microcontroller on the Arduino. The Arduino, which controlled the drivetrain, would be in charge of the automated stop, and the PCB would be in charge of the on/off button stop. We had to send two signals, one from the Arduino to the PCB and one from the PCB to the Arduino, to communicate that a stop had occurred.

2.4 Drivetrain Subsystem

2.4.1 Overview

The drivetrain subsystem will consist of the motor driver and two DC gear motors which will each have an encoder and will be attached to the front two wheels of the car. The motor drivers will receive a signal from the microcontroller to determine which way the car should turn. For the car to turn, only one motor will be on depending on the direction. For example, if the car turns right, the left wheel will be on while the right wheel will be off. Likewise, if the car turns left, the right wheel will turn on while the left wheel will remain off. The encoders for the DC gear motors will be used for more precise control of the wheels' direction and speed [13-16].

2.4.2 Design Procedure

For steering, there are many ways you can go about creating a straight path or a turn. In the earlier stages, we were discussing using a gyroscope sensor to be able to stay on course. We ultimately decided against it for this project because we felt adding another sensor to our already complicated project would not be good in the interest of time. We did implement a software solution using a PID control system to ensure that the wheels would be traveling at the same speed [17-20].

2.4.3 Design Details

Within the Drivetrain subsystem, the motors themselves were connected to the microcontroller to read the encoder pulses outputted from the motor, and the motor driver [13-16]. The motor driver would be sent signals from the microcontroller to change the PWM of each motor separately. The PID control system worked by reading the encoder pulses from the motor and trying to match the right encoder pulses to the left wheel encoder pulses [17-20]. Below you can see in Figure 2.5 the encoder pulses converging together to make the wheels run at the same speed.

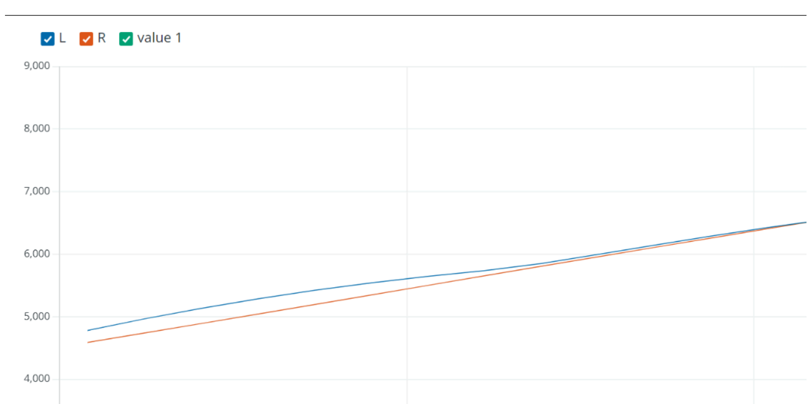

Figure 2.5. Right encoder Pulses (blue) matching left encoder pulses (red)

2.5 Salt Dispenser Subsystem

2.5.1 Overview

The salt dispenser subsystem consists of a dispenser flap and a spinning disk to allow for the salt to fall onto the driveway and be distributed evenly. The dispenser flap is controlled using a servo motor placed at the bottom of the salt dispenser. The spinning disk is controlled using a 100 RPM DC gear motor and a motor driver. Once the on/off button has been pressed, the control subsystem will send signals to the salt dispenser subsystem to open the dispenser flap and rotate the spinning disk. If the car has completed traveling throughout the driveway or if the on/off button has been pressed a second time, the control subsystem will send signals to close the dispenser flap and stop the rotation of the spinning disk.

2.5.2 Design Procedure

To allow the salt dispenser to efficiently distribute salt evenly throughout the driveway area, we needed to implement a mechanism that pushes and spreads the salt out of the car. To implement this, we researched ways in which salt spreaders on the market have been designed. Most salt spreaders that were found used a spinning disk with blades installed on the surface [21-24]. This design allows the material landing on the spinning disk to accelerate outwards once it hits the blades [25, 26]. Therefore, the spinning disk controlled using a DC gear motor and motor driver was chosen for the project due to its effective and simple design as well as its low cost.

To prevent the salt from being dispensed in the off state of the car, a dispenser flap that can open and close and is placed at the bottom of the salt dispenser was implemented. The dispenser flap was designed to be a sliding flap controlled by a servo motor such that once the car is switched on or off, the servo motor rotates 90° to open or close the dispenser. The flap was designed to be circular and to rotate horizontally rather than vertically to prevent the flap from getting stuck with the pressure from the salt. Figure 2.6 displays the schematic for the salt dispenser subsystem.

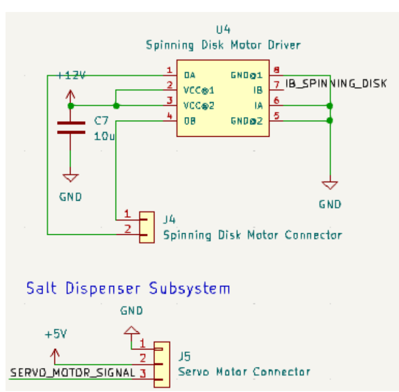

Figure 2.6. Salt dispenser subsystem schematic

2.5.3. Design Details

For the initial design of the spinning disk, the L9110 motor driver was used to control the DC gear motor. However, due to the damage to the motor driver from supplying a maximum supply voltage of 15 V, the L298N motor driver was used instead. Also, during the testing of the salt dispenser subsystem, large particles of salt would sometimes get stuck between the spinning wheel and the outer covering of the salt dispenser which would prevent the spinning disk from turning. To solve this, we reduced the size of the spinning disk to allow for more area between the edge of the spinning disk and the container for the salt dispenser. We also tried to allow less salt to fall through the dispenser flap by turning the servo motor less than 90° instead of from the initial 0° position to the 90° position to partially open the flap. However, the servo motor was only able to turn to certain angles including 0°, 90°, or 180°. We also switched to using rice for testing rather than salt as rice is smaller and more consistent in size. Another issue that was encountered was that upon the start-up of the car, the state of the car would toggle between the on and off states and start in the on position. Thus, the salt dispenser flap would open and close repeatedly and begin in the incorrect position. To solve this, we switched the initial state of the car by setting it to the on state rather than the off state to ensure that the car begins in the off position. Figure 2.7 displays the open and closed positions of the dispenser flap and the spinning disk.

Figure 2.7. Closed (left) and open (middle) positions of the salt dispenser flap and spinning disk (right)

3. Design Verification

3.1 Sensing Subsystem

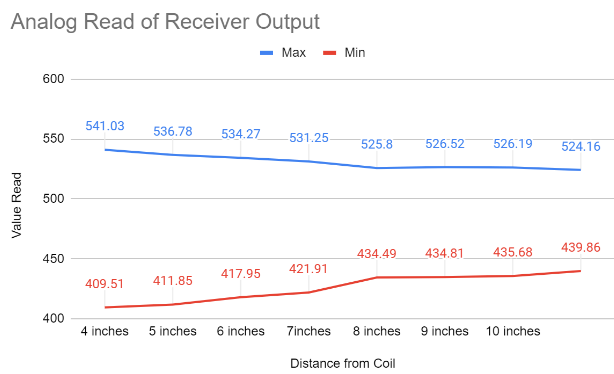

Figure 3.1. Arduino analog reading of the receiver output

To find the receiver threshold value we need to set, we used an Arduino Uno board to see what the analog reading looks like when the coil is at various distances away from the wire. We sampled the readings at every 5ms and found the max and min values over 100 readings. This was repeated 5 times to find the overall average max and min values. Figure 3.1 shows the resulting data collected. Since the induced voltage generated is still low, we were not able to see a consistent pattern in the analog readings, therefore we decided to set the threshold values based on the difference between them rather than two threshold levels. We also noticed that there were random peaks in the readings, meaning that there could be other interferences and noises. To solve this issue, we added counters such that the detection only occurs if the threshold is hit continuously in a row for some time.

An issue that was encountered when the sensing subsystem was being integrated with the rest of the subsystems was that a large amount of interference was observed with the receiver output. As the receiver coils were placed on the car and close to the DC motors for the wheels, the magnetic field variations from the wheels resulted in induced voltages on the receiver coil. This prevented the receiver from detecting the magnetic field variations from the transmitter wire. Thus, the sensing subsystem was built and implemented separately from the rest of the car.

3.2 Power Subsystem

To test that we had the required power supply, we used the digital multimeters in the lab to measure the voltage supply of each of the components. The first major block was that we had gotten the wrong 5 V converter which was outputting a current that did not satisfy our requirements. Fortunately, this was resolved once we added the Arduino board as it was able to output a voltage supply of 5 V. One critical issue was how much drained batteries affected our project. For the 12 V battery, we noticed a significant change in the drivetrain behavior as the battery declined over time. The wheels would spin slower which affected our turns since they are coded based on time duration. For the 9 V battery, we noticed that once it has drained below 9 V, the control subsystem wouldn't respond as well to the button presses. There weren't any better solutions other than continuously recharging/replacing the batteries or having the Arduino be powered from the laptop instead of the 9 V battery.

3.3 Control Subsystem

To test the control subsystem, we just had to ensure that our button was working correctly before we connected it to the PCB. We did this by breadboarding it before we attached it to the PCB.

3.4 Drivetrain Subsystem

To test the drivetrain subsystem we monitored the encoder pulse output from the wheels. We also monitored the actual position of the wheels as the car moved. When we placed the car on the ground it was fairly easy to tell if the car's wheels were going at the same speed or not based on the direction the car moved.

3.5 Salt Dispenser Subsystem

To test that the salt dispenser subsystem was working correctly, we observed the position of the salt dispenser flap and the motion of the spinning disk in the on-and-off states of the car. Once the on/off button was pressed, the servo motor for the salt dispenser flap rotated from the initial 0° position to about 90° to open. Also, the spinning disk rotated at about 100 RPM using the DC gear motor. Once the car reached the end of the driveway or if the on/off button was pressed a second time during the operation of the car, the servo motor rotated in the opposite direction back to the initial 0° position to close the flap. The spinning disk also stopped rotating. Thus, there were no major blocks that impacted the operation of the salt dispenser subsystem once the design changes discussed in section 2.5.3 were implemented.

4. Costs and Schedule

4.1 Parts

Description	Quantity	Unit Price	Link
GEARMOTOR 83 RPM 12V W/ENCODER (wheel)	2	$29.00	link
Hs-311 Standard Servo (dispenser flap)	1	Self-Service Inventory	datasheet
TB6612FNG (wheel)	1	$1.97	link
Linear Voltage Regulator IC Positive Fixed 1 Output 1A TO-252 (value: 5V) (BD50FC0FP-E2)	1	Electronic Service Shop	datasheet
Linear Voltage Regulator IC Positive Fixed 1 Output 1A TO-252-3 (value: 3.3V) (AZ1117CD-3.3TRG1)	1	Electronic Service Shop	datasheet
ATmega328P Microcontroller	1	Electronic Service Shop	datasheet
Push Button	1	Self-Service Inventory	
Motor control driver chip L9110 (spinning disk)	1	$0.80	datasheet
Greartisan DC 12V 100RPM Gear Motor High Torque (spinning disk)	1	From machine shop	
Funnel for Salt	1	$9.49	link
KBT 12V 5200mAh Rechargeable Li-ion Battery	1	$32.99	link
9V Battery - Alkaline (transmitter)	1	ECE Supply Center: $2.32	datasheet
CMOS TIMER TLC555CP (transmitter)	1	ECE Supply Center: $1.14	datasheet
OPA2340 Single-Supply, Rail-to-Rail Operational Amplifiers (receiver)	2	$4.06	link
Misc. SMD parts (estimated unit price)	40	$0.10	
Total Cost			**$114.77***

4.2 Labor

From the AY21-22, the average starting salary for a UIUC Electrical Engineering graduate is $87,769 and for a Computer Engineering graduate is $109,176 [27]. Assuming a 40-hour work week and 48 working weeks per year, the hourly salary is $45.71 for EE graduates and $56.86 for CE graduates.

Labor Costs Formula: **ideal salary (hourly rate) * actual hours spent * 2.5**

Arya : $56.86 * 106 hrs * 2.5 = $15,067.90

Mayura : $45.71 * 106 hrs * 2.5 = $12,113.15

Candy : $45.71 * 106 hrs * 2.5 = $12,113.15

4.3 Schedule

Week	**Task**	**Person**
February 18th - February 24th	- Research the transmitter and the receiver system for the fence. - Talk with TA Jason - Complete the Design Document by Thursday, February 22nd	Everyone
	Order the parts needed for the prototype.	Arya
	Create circuit diagram for drivetrain subsystem	Candy
	Create circuit diagram for salt dispenser subsystem	Mayura
February 25th - March 2nd	Complete design review with the instructor and TAs.	Everyone
	Order the remaining parts that are needed for the car	
	Create circuit diagram and prototype sensing subsystem (making sure receiver is getting signal from transmitter)	Mayura & Candy
	Create circuit diagram and prototype power subsystem	Arya
	Complete the PCB design and the PCB review.	Everyone
March 3rd - March 9th	Complete finalizing the PCB design	Everyone
	PCB order by March 5th.	
	Complete teamwork evaluation.	
	Talk with the machine shop about physical design of car & make revision as needed	
	Work on Sensing subsystem prototyping	Mayura & Candy
	Work on drivetrain subsystem 1. programming microcontroller 2. connecting motors to motor drivers	Arya
March 10th - March 16th	SPRING BREAK	Everyone
	Can be used as catch-up week if needed	

March 17th - March 23rd	Revise & Order PCB	Everyone
	Start working on salt dispenser subsystem: 1. Programming microcontroller for servo motor (flap) and dc motor (spinning disk) 2. Finalize physical dimensions for flap and spinning disk	Mayura
	Work on sensing subsystem prototype	Candy
	Work on drivetrain subsystem prototype	Arya
March 24th - March 30th	Revise & Order PCB	Everyone
	Give machine shop information to make spinning disk with fins/dispenser flap/funnel holder	
	Brainstorm enclosure designs	
	Continue working on salt dispenser prototype	Mayura
	Create final version of sensing subsystem	Candy
	Create final version of drivetrain subsystem	Arya
	Put together the drivetrain subsystem and the sensing subsystem and test & debug	Everyone
March 31st - April 6th	Create final version of salt dispenser	Mayura
	Add salt dispenser to drivetrain/sensing subsystem	Everyone
	Revise & Order PCB	
	Creating the enclosures, 3D print them and add to car	Candy & Arya
April 7th - April 13th	Revise & Order PCB	Everyone
	Test and revise salt dispenser/ drivetrain/ sensing subsystem together	
	Revise enclosures	
April 14th - April 20th	Fixing Minor Errors	Everyone
April 21st - April 27th	Final Demo	Everyone
	Work on Final presentation	
April 28th - May 4th	Final Presentation	Everyone
	Work on Final paper	
May 5th - May 11th	Lab Notebook due	Everyone

5. Conclusion

5.1 Accomplishments

While we were not able to completely integrate all of our subsystems, we were able to get a majority of the subsystems functional and integrated. Our final demo showed two main accomplishments: the car can make the desired zig-zag patterns with the salt dispenser distributing salt in response to the ON/OFF button and the motors can stop once the coil has come within 10 in of the buried wire. Due to the interference issues from the motors, we were not able to add our sensing subsystem onto the car which meant that the car did not have a way of detecting the boundary and thus we had to manually code when the turns would occur. However, we were able to get the PID controller working and the car is capable of self-adjusting itself to go straight. In addition, our sensing subsystem was successful as it was able to detect the wire and performed as expected according to our design. If we had focused more on the integration during our initial design phase or allotted more time for integration, we believed we could have come up with a solution.

5.2 Uncertainties

One uncertainty was that we were not able to get the PID controller working on the ATmega328P even when the controller was working on the Arduino. Our biggest problem was that we didn't add enough test points on our PCB thus making it difficult to monitor the encoder pulses and effectively tune the PID. In addition, the PCB board had many errors such as incorrect pin connections, power/current requirements not being met, and wrong trace widths, all of which could have also affected the functionality of the PID. While we were able to get the PID working on the Arduino, getting the PCB board to work would have made our overall project look more professional and cleaner.

Another uncertainty was the inconsistency of the salt dispenser. While we were able to demo successfully using rice instead of actual salt, the nonuniform shape of the salt will cause the bigger pieces to result in the dispenser flap getting stuck. If we had more time, we believe that we could solve this problem by making a smaller opening in the cylinder and by using a faster motor for the disk. This way, the salt would be distributed out at a faster rate and thus decrease the chances that it will accumulate on the car base.

5.3 Future work

There are many ways in which we could improve the functionality of our project in the future. While we were able to implement the sensing, drivetrain, and salt dispenser subsystems successfully, we were unable to integrate the sensing subsystem with the rest of the subsystems due to interference from the DC wheel motors. To solve this, we could reduce the interference by adding capacitors to the motor terminals of the DC wheel motors to filter out the AC noise frequencies. Another way in which this could be solved is to add an LC circuit with a resonant frequency that matches the frequency of the transmitter. This would be an effective hardware filter to prevent noise from impacting the receiver output. Also, a digital implementation to solve this problem would be to create a bandpass filter in the code for the microcontroller to filter the noise from the motors.

The transmitter and receiver sensing mechanism used in this project was implemented as it was cost-effective and efficient for our purposes. However, we could also implement other methods of detecting the edge of the driveway. For example, an RFID system could be developed by placing RFID readers along the edge of the driveway and a tag on the car. This would allow for more consistent detection of the car once it reaches the edge of the driveway.

Another way in which the project could be improved is by using wheels that allow for more traction on the roads. The wheels that were used for the project were readily available and could work on most surfaces. However, to allow the car to also be able to operate on icy or snowy road conditions, better wheels could be used to prevent the car from sliding.

5.4 Ethical considerations

As described in the IEEE Code of Ethics II, we will create a fair and non-discriminatory working environment throughout the project [28]. To achieve this, all work was distributed evenly and fairly and all members were expected to treat each other and any mentor TA and Professor with respect. In addition, according to IEEE Code of Ethics I.1 and I.5, we will take in any feedback and practice proper testing procedures to produce honest and accurate results to ensure the functionality of the device and address any safety concerns.

References

[1] Ti, www.ti.com/lit/gpn/TLC555. Accessed 28 Mar. 2024.

[2] Amazon.Com : Perimeter Technology - R21 Invisible Fence Compatible Dog Collar - 7K Frequency : Perimeter Technologies : Wireless Pet Fence Products : Pet Supplies, www.amazon.com/PERIMETER-TECHNOLOGY-Invisible-Compatible-Collar/dp/B002QG0IUI. Accessed 1 May 2024.

[3] Amazon.Com: Perimeter Technologies Invisible Fence 700 Series Compatible Dog Fence Collar - Works with the 10,000 Fence Frequency System : Pet Supplies, www.amazon.com/Perimeter-Technologies-Invisible-Compatible-Collar/dp/B002QG5J1Q. Accessed 2 May 2024.

[4] "555 Timer Monostable Circuit Calculator - Engineering Calculators & Tools." All About Circuits, www.allaboutcircuits.com/tools/555-timer-monostable-circuit/. Accessed 27 Mar. 2024.

[5] Storr, Wayne. "555 Oscillator Tutorial - the Astable Multivibrator." Basic Electronics Tutorials, 15 Mar. 2024, www.electronics-tutorials.ws/waveforms/555_oscillator.html.

[6] "Simple Square Wave to Sine Wave Converter." Circuit Diagram, circuitdigest.com/electronic-circuits/simple-square-wave-to-sine-wave-converter. Accessed 27 Mar. 2024.

[7] "Calculating RC Low-Pass Filter Cut-off Frequency and Transfer Function - Technical Articles." EEPower, eepower.com/technical-articles/calculating-rc-low-pass-filter-cut-off-frequency-and-transfer-function/#. Accessed 27 Mar. 2024.

[8] Hunter, Geoffrey, and Geoffrey HunterDude making stuff. Twitter GitHubkofiwidget2.init("Support Me on Ko-fi." "How to Create Sine Waves from Square Waves and RC Filters." Mbedded.Ninja, 26 Aug. 2022, blog.mbedded.ninja/electronics/circuit-design/how-to-create-sine-waves-from-square-waves-and-rc-filters/.

[9] "Electromagnetic Induction: Magnetism and Electromagnetism: Electronics Textbook." All About Circuits, www.allaboutcircuits.com/textbook/direct-current/chpt-14/electromagnetic-induction/#:~:text=Faraday's%20equation%20for%20induced%20voltage,voltage%20according%20to%20Faraday's%20formula). Accessed 27 Mar. 2024.

[10] "Flux Loop." Wikipedia, Wikimedia Foundation, 1 June 2023, en.wikipedia.org/wiki/Flux_loop.

[11] nagwa_label. “Lesson Explainer: The Magnetic Field Due to a Current in a Straight Wire Physics.” Nagwa, www.nagwa.com/en/explainers/909137183476/. Accessed 27 Mar. 2024.

[12] Walker , Patrick, and Christian Miller . “How to Calculate the Magnetic Flux through a Circular Loop with Arbitrary Orientation to the Field.” Https://Study.Com/Skill/Learn/How-to-Calculate-the-Magnetic-Flux-through-a-Circular-Loop-with-Arbitrary-Orientation-Relative-to-the-Field-Explanation.Html. Accessed 27 Mar. 2024.

[13] “Metal DC Geared Motor w/Encoder - 12V 251rpm 18kg.CM Wiki.” DFRobot, wiki.dfrobot.com/12V_DC_Motor_251rpm_w_Encoder__SKU__FIT0186_. Accessed 1 May 2024.

[14] “12V Metal DC Geared Motor with Encoder (131:1, 83rpm, 45kg.Cm).” DFRobot, www.dfrobot.com/product-633.html. Accessed 1 May 2024.

[15] “Calculate Pulses per Revolution for a DC Motor with Encoder.” Automatic Addison, 22 Apr. 2021, automaticaddison.com/calculate-pulses-per-revolution-for-a-dc-motor-with-encoder/.

[16] What Is the Difference between CPR and PPR?, www.rls.si/eng/faq/index/show/cat_id/2/faq/59/. Accessed 1 May 2024.

[17] “DC Motor Speed: PID Controller Design.” Control Tutorials for MATLAB and Simulink - Motor Speed: PID Controller Design, ctms.engin.umich.edu/CTMS/index.php?example=MotorSpeed&section=ControlPID. Accessed 1 May 2024.

[18] AnnaMai, and Instructables. “DC Motor Speed Control Pid.” Instructables, Instructables, 23 Oct. 2021, www.instructables.com/DC-Motor-Speed-Control-PID/.

[19] The Bored Robot, theboredrobot.com/. Accessed 1 May 2024.

[20] “Design an Arduino Based Encoder Motor Using PID Controller.” Circuit Digest - Electronics Engineering News, Latest Products, Articles and Projects, circuitdigest.com/microcontroller-projects/arduino-based-encoder-motor-using-pid-controller. Accessed 1 May 2024.

[21] “125 Lb Capacity, Pneumatic, Broadcast Spreader.” Grainger, www.grainger.com/product/4UHD1?gucid=N%3AN%3APS%3APaid%3AGGL%3ACSM-2295%3A4P7A1P%3A20501231&gad_source=1&gclid=Cj0KCQjw0MexBhD3ARIsAEI3WHL2r3dJdxFo7YrnCseRCg0g0Hw2ScI7gEqRfPwdH7UNzFywnGEFcl0aAoX1EALw_wcB&gclsrc=aw.ds. Accessed 1 May 2024.

[22] “30 Lb Capacity, Pneumatic, Broadcast Spreader.” Grainger, www.grainger.com/product/WESTWARD-Broadcast-Spreader-30-lb-Capacity-10F635?opr=PDPRRDSP&analytics=dsrrItems_4UHD1. Accessed 1 May 2024.

[23] “Echo RB-60 Spreader.” Georgia Equipment Co., georgiaequipment.com/shop/echo-power-equipment/echo-rb-60-spreader/. Accessed 1 May 2024.

[24] “Salt Spreaders.” Meyer, www.meyerproducts.com/salt-spreaders. Accessed 1 May 2024.

[25] Lawrence, Written. “Your Comprehensive Guide to Salt Spreaders.” Cresco Spreaders, 7 Oct. 2022, crescospreaders.com/your-comprehensive-guide-to-salt-spreaders/#:~:text=Salt%20spreaders%20are%20generally%20simple,material%20over%20the%20desired%20area.

[26] Report 12: Flow of Salt Particles on a Spinning Spreader Disk, pure.au.dk/portal/files/87361608/Report12_Flow_of_salt_particles_03jun.pdf. Accessed 2 May 2024.

[27] Grainger Engineering Office of Marketing and Communications. (n.d.). Salary averages. Electrical & Computer Engineering | UIUC.

[28] IEEE Code of Ethics, www.ieee.org/about/corporate/governance/p7-8.html. Accessed 1 May 2024.

Appendix A Requirement and Verification Table

Table 1. Sensing Subsystem

Requirements	**Verification**	**Verification status (Y or N)**
Once the transmitter has been powered on by the 9 V battery, the voltage of the output pin and the wire must be sinusoidal and alternating between about 0 V to about 9 V.	● Ensure that the transmitter is powered off by ensuring that the 9 V battery is not connected to the transmitter. ● Then, power the transmitter by connecting the 9 V battery to the ‘Vdd,’ pin of the TLC555. ● Then, measure the output of the TLC555 and the wire using an oscilloscope. ● Confirm that the voltage read on the digital multimeter is sinusoidal and alternates between about 0 V and about 9 V.	**Y**
Once the transmitter has been powered off after it has been on, the voltage of the output pin and the wire must be 0 V.	● Ensure that the transmitter is powered on by ensuring that the 9 V battery is connected to the transmitter. Also, measure the output of the TLC555 and the wire using an oscilloscope and confirm that the voltage read on the digital multimeter is alternating between about 0 V and about 9 V. ● Then, power off the transmitter by disconnecting the transmitter from the 9 V battery. ● Then, measure the output of the TLC555 and the wire using a digital multimeter. ● Confirm that the voltage read on the digital multimeter is 0 V.	**Y**
The output from the receiver to the microcontroller must be able to output a voltage signal that is sinusoidal and alternates between 0 V and 3.3 V with a 10% margin once it has come within 10 in of the buried wire.	● Ensure that the receiver is in the OFF state and is away from the buried wire ● Then press the ON/OFF button once ● Move the receiver toward the buried wire such that they are 10 in apart and use an oscilloscope to view the output waveform ● Confirm that the output waveform on the oscilloscope is sinusoidal and alternates between 0 V and 3.3 V.	**Y**

Table 2. Power Subsystem

Requirements	Verification	Verification status (Y or N)
Must provide a constant supply voltage of 12 V with a ± 10% error to the DC motors	• Ensure that the car is in the 'off' state and the wheel and disk is not spinning • Then, press the on/off button once and use a digital multimeter to measure the voltage provided to the DC motor • Confirm that the value read on the digital multimeter is within range • Repeat same procedure for the other wheel DC motor and spinning disk DC motor	**Y**
Must provide voltage in the range of 4.8-6.0 V to ensure servo motor is operating correctly	• Ensure that the car is in the 'off' state and the dispenser flap is covering the opening on the dispenser. • Then, press the on/off button once and use a digital multimeter to measure the voltage provided to the servo motor. • Confirm that the value read on the digital multimeter is about 4.8 V- 6.0 V.	**Y**
Must provide voltage in the range of 2.7-5.5 V to ensure the DC motor drivers are operating correctly	• Ensure that the car is in the 'off' state and the wheel is not spinning • Then, press the on/off button once and use a digital multimeter to measure the voltage provided to the DC motor driver • Confirm that the value read on the digital multimeter is about 2.7-5.5 V. • Repeat the same procedure to verify the DC motor driver used for the spinning disk	**Y**
Must provide voltage in the range of 2.7-5.5 V to ensure ATmega328P microcontroller is operating correctly	• Ensure that the car is in the 'off' state • Then, press the on/off button once and use a digital multimeter to measure the voltage provided to the microcontroller • Confirm that the value read on the digital multimeter is within 2.7-5.5 V.	**Y**
Must provide a constant supply voltage of 9 V ± 10% error to ensure the transmitter is operating	• Ensure that the transmitter is powered off by ensuring that the 9 V battery is not connected to the transmitter.	**Y**

correctly.	• Then, power the transmitter by connecting the 9 V battery to the 'Vdd,' pin of the TLC555. • Then, measure the 'Vdd' of the TLC555 and the wire using a digital multimeter. • Confirm that the voltage read on the digital multimeter is within range	
Must provide a constant supply voltage of 3.3 V with a ± 10% error to the receiver	• Ensure that the car is in the 'off' state • Then, press the on/off button once and use a digital multimeter to measure the voltage provided to the receiver • Confirm that the value read on the digital multimeter is within range.	**Y**
Should shut off once the emergency stop button has been triggered. The car should turn off in 10 seconds.	• Ensure that the car is in the 'on' state (dispenser flap is opened, disk is spinning, and wheels are moving) • Using the digital multimeter confirm that all components are turned off and the car is in the 'off' state (dispenser flap is closed, the disk has stopped spinning, and wheels are not moving)	**Y**
Shut off automatically once it has finished salting the driveway. The car should turn off in 10 seconds.	• Ensure that the car is in the 'on' state (dispenser flap is opened, disk is spinning, and wheels are moving) • Place the car at the corner of the wired boundary to mimic the position it will be at once it has reached the end of the driveway. • Using the digital multimeter confirm that all components are turned off and the car is in the 'off' state (dispenser flap is closed, the disk has stopped spinning, and wheels are not moving)	**Y**

Table 3. Control Subsystem

Requirements	**Verification**	**Verification status (Y or N)**
The initial press of the on/off	• Ensure that the car is initially in the 'off,'	**Y**

button must start up the car and begin dispensing the salt.	state and the car is not moving. ● Then, press the on/off button once and ensure that the ATmega328 microcontroller has been powered using a digital multimeter to measure the voltage at the Vcc pin of the microcontroller which should be between the range of 2.7 V to 5.5 V. ● Then, measure the voltage at the pins that are connected to the motor driver for the servo motor, the motor driver for the DC gear motors with the encoder, and the motor driver for the DC motor to ensure that the motors have received the signal to turn on. ● Confirm that the dispenser flap on the car has opened, the car is moving, and the spinning wheel is rotating.	
The second press of the on/off button must stop the car moving, the spinning disk rotating, and close the dispenser flap on the car.	● Ensure that the car is initially in the 'on,' state and the car is moving. ● Then, press the on/off button and ensure that the ATmega328 microcontroller has been powered off using a digital multimeter to measure the voltage at the Vcc pin of the microcontroller, which should be 0 V. ● Then, measure the voltage at the pins that are connected to the motor driver for the servo motor, the motor driver for the DC gear motors with the encoder, and the motor driver for the DC motor to ensure that the motors have received the signal to turn off. ● Confirm that the dispenser flap on the car has closed, the car has stopped moving, and the spinning wheel has stopped rotating.	**Y**
Once the car has completed dispensing the salt throughout the entire area of the driveway, the dispenser flap on the car should close, the car should stop moving, and the spinning disk should stop rotating.	● Ensure that the car is initially in the 'on,' state and the car is moving. ● Then, wait for the car to dispense all of the salt throughout the entire area of the driveway. The car will have completed traveling throughout the entire driveway once it reaches the wired fence two consecutive times, or the car receives the signal from the transmitter two consecutive times when it turns. ● Then, ensure that the ATmega328 microcontroller has been powered off using	**Y**

	a digital multimeter to measure the voltage at the Vcc pin of the microcontroller, which should be 0 V. ● Then, measure the voltage at the pins that are connected to the motor driver for the servo motor, the motor driver for the DC gear motors with the encoder, and the motor driver for the DC motor to ensure that the motors have received the signal to turn off. ● Confirm that the dispenser flap on the car has closed, the car has stopped moving, and the spinning wheel has stopped rotating.	

Table 4. DriveTrain Subsystem

Requirements	**Verification**	**Verification status (Y or N)**
The Gear Motors connected to the two front wheels will always be at the same position to ensure they are traveling at the same rpm.	● Ensure that the car is in the 'off' state ● Ensure that the output pins of the DC Gear Motor (Yellow and White wires) are connected to Input pins of the microcontroller. ● The microcontroller should already be programmed to read the input of these wires. ● Then, press the on/off button once and wait for the car to turn on. ● Once the car is on, confirm that the output of the motor signals are the same.	**Y**
Once the car reaches the edge of the driveway the car should turn 90 +/- 5 degrees.	● Ensure that the car is in the 'off,' state. ● Place the car within the driveway boundaries. ● Press the on/off button once and wait for the car to turn on. ● Once the car reaches the edge, mark where on the ground the direction the car is facing. ● After the car turns, again mark the direction the car is facing. ● Measure the angle between these two directions with a ruler and verify that it is indeed within the range of 90 +/- 5 degrees.	**Y**

Once the car turns 90 degrees, the car should move forward 6 +/- 2 inches before turning 90 degrees again.	● Ensure that the car is in the 'off,' state. ● Place the car within the driveway boundaries. ● Press the on/off button once and wait for the car to turn on ● Once the car reaches the edge and turns, mark where on the ground the front of the car is. ● After the forward movement of the car, again mark the position of the car. ● Measure the distance between these two marks with a ruler and verify that it is indeed within the range of 6 +/- 2 inches.	**Y**

Table 5. Salt Dispensing Subsystem

Requirements	**Verification**	**Verification status (Y or N)**
The servo motor for the dispenser flap must rotate in the clockwise direction about 90 +/- 5 degrees to open the flap.	● Ensure that the car is in the 'off,' state and the dispenser flap is covering the opening on the dispenser. ● Then, press the on/off button once and record the value read from the microcontroller to provide the signal to the servo motor. ● Confirm that the dispenser flap rotates about 90 +/- 5 degrees to open the flap.	**Y**
The servo motor for the dispenser flap must rotate in the counterclockwise direction for about 90 +/- 5 degrees to close the flap.	● Ensure that the car is in the 'off,' state and the dispenser flap is covering the opening on the dispenser. ● Then, press the on/off button once and record the value read from the microcontroller to provide the signal to the servo motor. ● Confirm that the dispenser flap rotates about 90 +/- 5 degrees to close the flap.	**Y**
The servo motor for the dispenser flap must rotate back to the starting position once the car has traveled throughout the entire driveway.	● Ensure that the car is in the 'on,' state and the dispenser flap has been opened to allow the salt to fall through the dispenser. ● Then, allow the car to travel throughout the	**Y**

	entire driveway. ● Then, use a digital multimeter to measure the voltage provided by the servo motor. ● Confirm that the dispenser flap has rotated back to its original position.	
The servo motor for the dispenser flap must rotate back to the starting position once the on/off button has been pressed.	● Ensure that the car is in the ‘on,’ state and the dispenser flap has been opened to allow the salt to fall through the dispenser. ● Then, press the on/off button once and record the value read from the microcontroller to provide the signal to the servo motor. ● Then, use a digital multimeter to measure the voltage provided by the servo motor. ● Confirm that the dispenser flap has rotated back to its original position.	**Y**
The spinning disk must rotate at a speed of about 200 rpm.	● Ensure that the car is in the ‘off,’ state and the disk is not spinning. ● Then, press the on/off button once and confirm that the spinning disk rotates at a speed of about 200 RPM.	**Y**
The DC motor for the spinning disk must stop rotating once the car has traveled throughout the entire driveway or the on/off button has been pressed.	● Ensure that the car is in the ‘on,’ state and the spinning disk is rotating. ● Then, press the on/off button once and record the value read from the microcontroller to provide the signal to the DC motor. ● Then, use a digital multimeter to measure the voltage provided by the DC motor. ● Confirm that the spinning disk has stopped rotating.	**Y**