SYSC4805 |
Computer Systems Design Lab

Cardinal

Final Report - Autonomous Snowplow

Aaryan Ahuja 101172239
Eric Hardy 101115423
Josh Lalonde 101109655
Zakariyya Almalki 101152124

December 8th, 2023

Table of Contents

1. Updated Project Proposal

Our project underwent several modifications during its development to meet our design objectives. One of the major changes was the elimination of the ultrasonic sensor, which we initially planned to use for obstacle detection. As well, we opted for an infrared light sensor, which proved to be more reliable and accurate than the time-of-flight sensor originally proposed.

The ultrasonic sensor was eliminated because it was not performing well and time constraints prevented further development time spent on this sensor. The reason for changing to IR sensors instead of using the time-of-flight sensor was because of the strong interference experienced with I2C when the motors were active. The IR sensor still experienced interference but it was less of an issue as it still continued to provide readings and did not freeze like the time-of-flight sensor did.

A complete copy of the progress proposal can be found below in Appendix A.

2. Control Charts

The following control charts show the performance of our system during the testing phase. They display the data points and the control limits for each process variable that we measured. The control limits are based on the target values and the acceptable deviations that we specified in our project proposal. We can use the control charts to analyze how well our system met the requirements and to identify any potential problems or improvements.

2.1. Border Line Detection

The figure below shows the control chart that we used to evaluate the line detection of our autonomous snow plow. The line detection system was designed to stop the plow 5mm away from the border line, as stated in our project proposal. Therefore, we set the upper and lower control limits at +5mm and -5mm, respectively. The control chart plots the deviation of the plow from the target value for each sample run. We performed 10 sample runs of the vehicle driving over the line and stopping as soon as it detects the line, under the same speed conditions. The deviation fluctuated between the control limits, with a maximum of 7mm and a minimum of -3mm. The control chart shows that the data tended to stay close to the average target of 0mm, which suggests that the variation was random and not due to any assignable causes. This means that our line detection system was consistent and reliable, and met the specifications of our project.

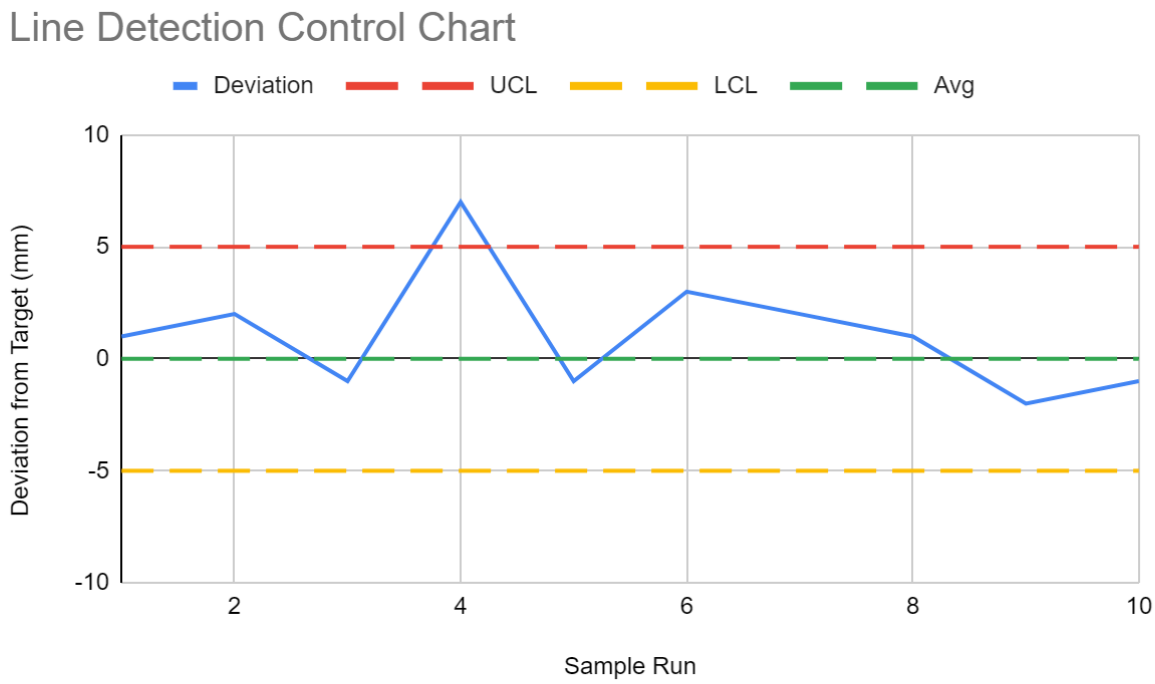


Figure 1: Line Detection Control Chart

2.2. Object Avoidance

The time-of-flight control chart is shown below. These results were taken with the motors running to accurately show the issue with the I2C interference. The system was designed to have a deviation of less than 5 cm to allow the robot to properly avoid obstacles but due to the interference of the motors the results were not satisfactory. The sensor readings are very dependent on the distance the TOF sensor was reading before the motors were started. It essentially froze on that reading. A total of 10 sample runs were performed and a maximum of 7 cm and minimum of -7 cm was recorded under various starting distances from an object. This shows that the TOF sensor could not be reliably used while the motors were operating and was therefore removed from the device as it was not suitable.

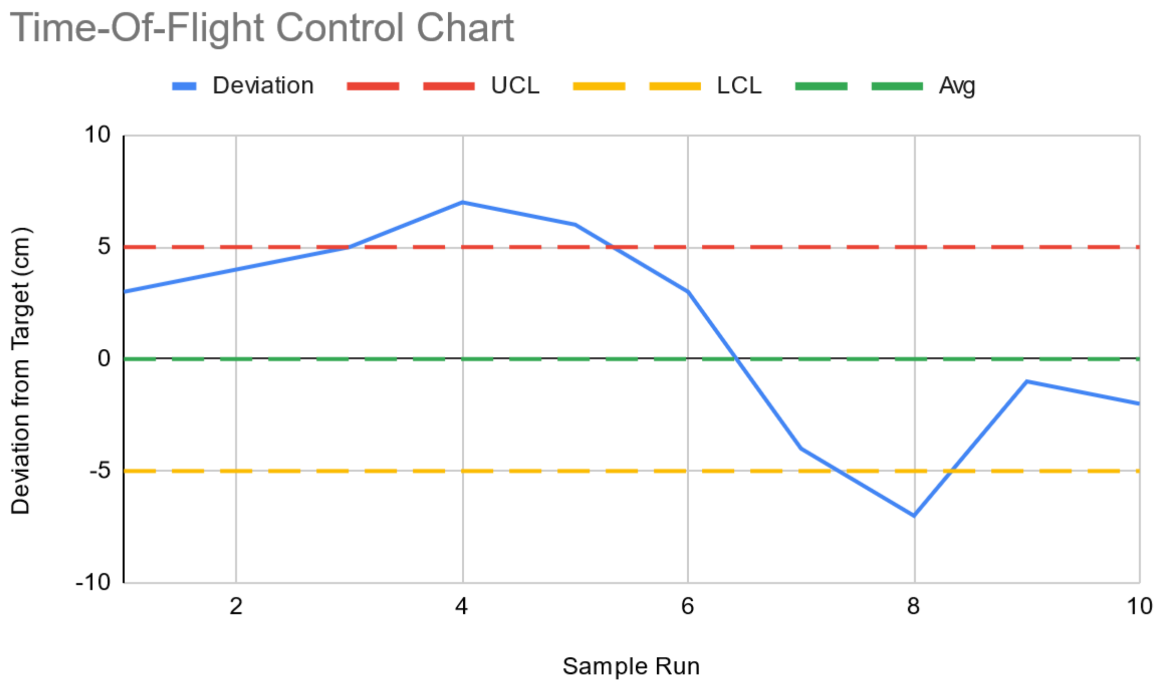


Figure 2: Time of Flight Control Chart

The IR sensor performed more consistently even though it was also affected by the interference from the motors. The control chart for the IR sensor is shown below. Again, with a total of 10 sample runs, this sensor performed better. The sensors were tuned to be accurate when the motors are running, which dropped the voltage available to the sensor, so that they are consistent during operation. The results were deemed satisfactory and are within the expected range.

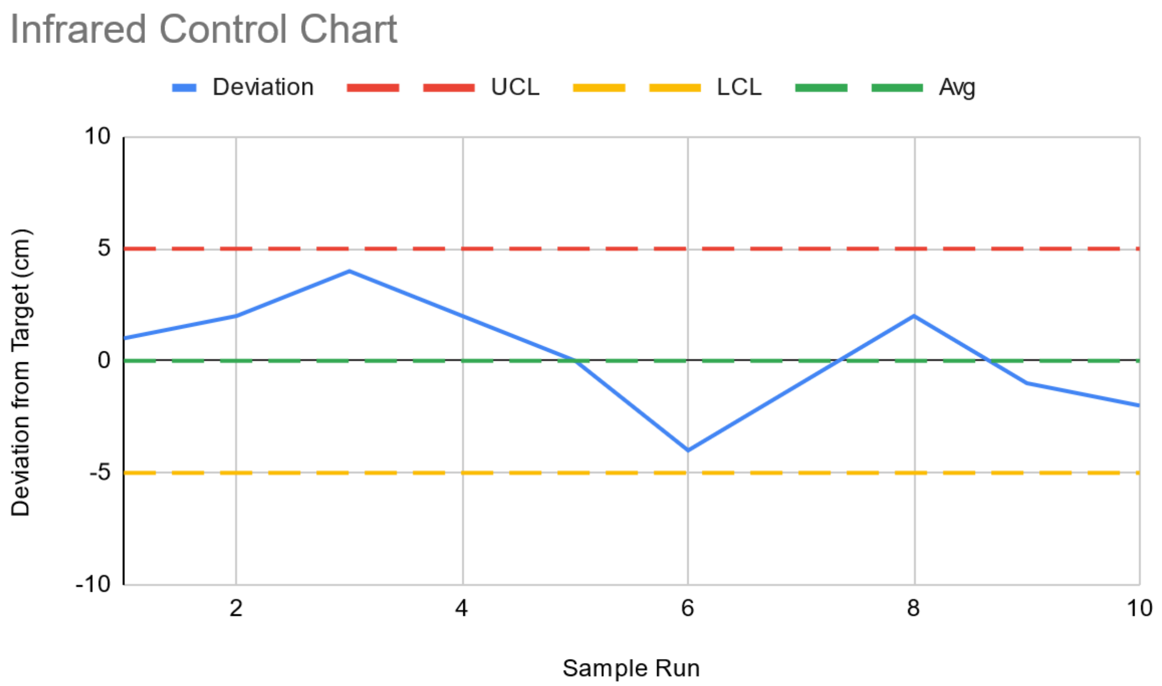


Figure 3: Infrared Sensor Control Chart

The Ultrasonic sensor is not reliable as the other sensors used in the project , 10 sample runs were done to check its performance out of which the 3 times the sensor failed to detect any obstacles which are denoted by 0 for the graphs purposes, The sensor also caused interferences with the motors where the motors would stop working on detecting an obstacle or vice versa . The sensor would detect something out of bounds because of the wide viewing area of 17 degrees causing problems and detecting random objects not within the frame of reference.

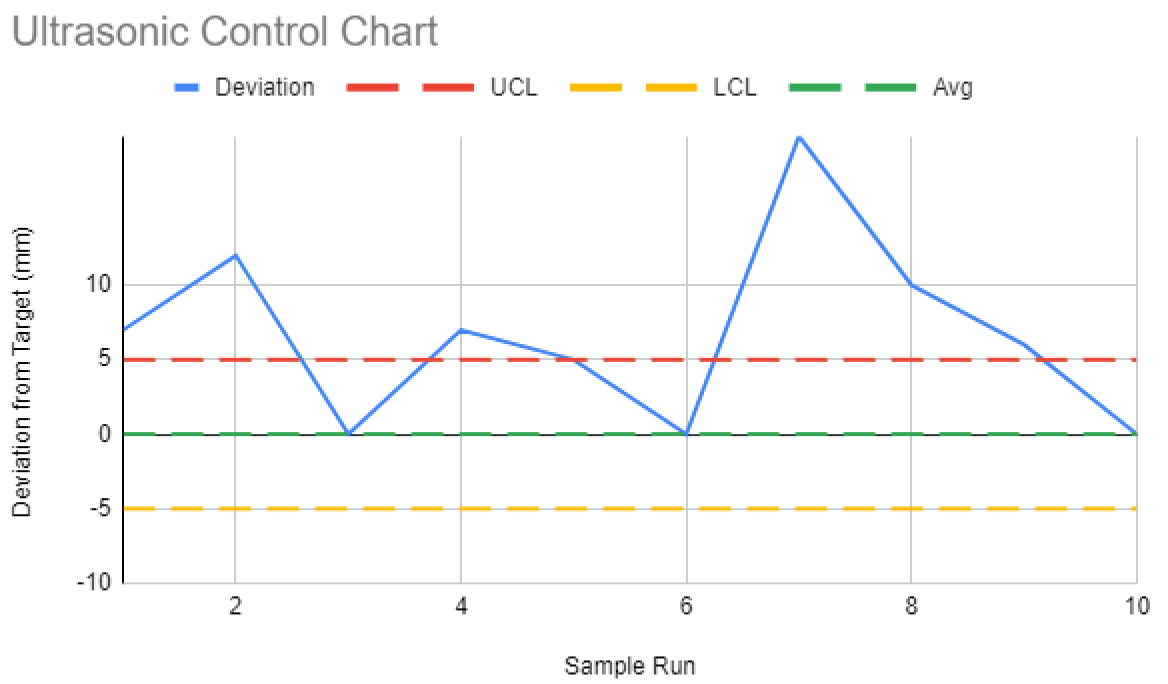


Figure 4: Ultrasonic Control Chart

2.3. Speed Constraint

Figure 5 displays the control chart for the speed constraint of the robot's motors. The motors are set to not reach a speed greater than 30 cm/s, as stated in the project proposal as one of our requirements. The method used to test was measuring out 30 centimeters, and timing the robot to ensure it does not travel the 30 centimeters faster than one second. The chart outlines the deviation of the speed from the target value for each run. Our target value is 30 cm, we have set the upper control limit to be an additional 2 centimeters and the lower control limit to be 2 centimeters lower than our target value. We performed 7 sample runs, where we timed the robot as it traveled the 30 centimeters. There was no noticeable deviation. Thus, the speed constraints of the motors were deemed to be persistent and dependable.

Figure 5: Motors Speed Constraint Control Chart

2.4. Snow Removal

Figure 6 represents the accuracy of the snow removal control chart. Within our proposal we made it a requirement that the snow must be removed from the area within 1cm without crossing over the line. The below chart represents the distance that the vehicle left the blocks when testing 10 different sample runs. The average 0 mark represents the maximum of 1cm over the line. As seen in the control chart the snow was removed no more or less than 1cm away from the maximum of 1cm away from the line. Therefore, we deemed our snow removal tests as successful.

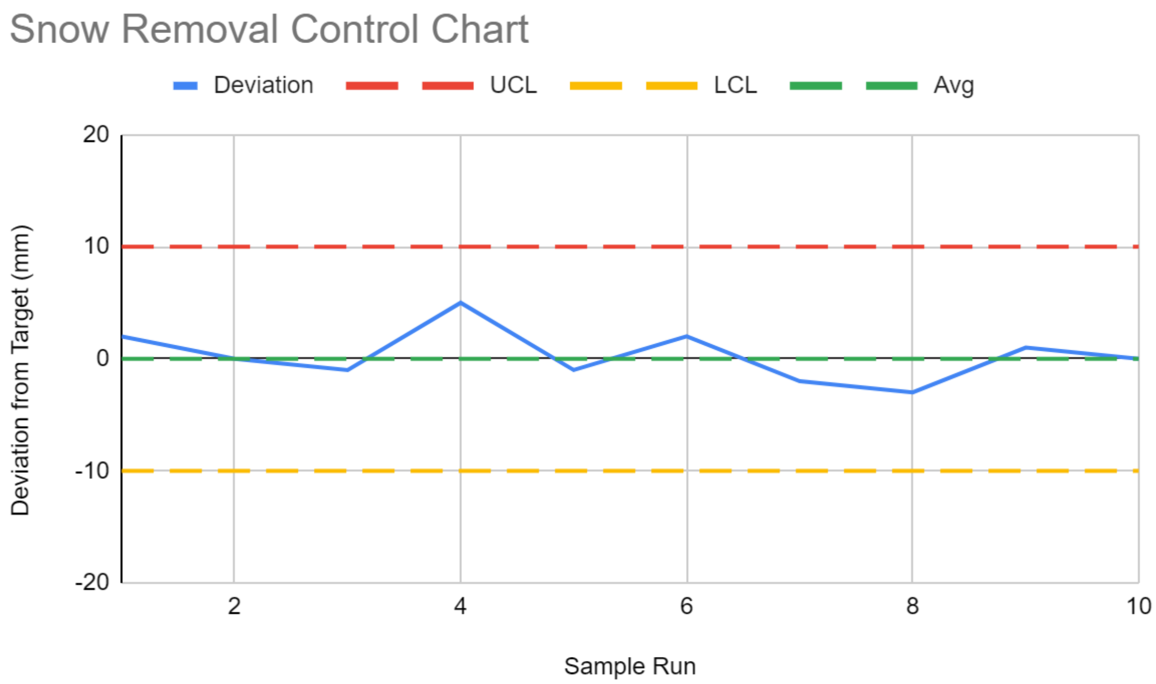


Figure 6: Snow Removal Control Chart

3. System Testing

3.1. Unit Testing

3.1.1. Time of Flight Sensor

The time-of-flight sensor testing resulted in observing poor performance during motor operation, however the readings met the desired accuracy while the motors were off. This was verified by performing two sets of tests, one with the motors and one without. The sensor readings were compared to the actual distance manually with a ruler. The results showed that the sensor could not meet the accuracy criteria while the motors were on and was therefore unreliable.

3.1.2. Ultrasonic Sensor

The Ultrasonic sensor was placed on the sides of the robot chassis initially but testing showed unreliable performance during operations, The readings were off by a significant margin which resulted in scrapping of the sensor, Sometimes the sensor would stop detecting objects and would only work after resetting , The sensor would otherwise detect obstacles not placed in the test area, which would give us false readings, This was also caused by the wide angle view of the sensor which would lead the sensor obtain false readings.The tests confirmed that it would be better to use the IR sensor instead of the ultrasonic sensor.

3.1.3. Line Following Sensor

This sensor was tested by placing it on the left, in the center, and on the right of the boundary and the success scenario was that the sensor would be able to detect if it had crossed the threshold. The acceptable performance criteria was that the device was accurate in detecting a line within 1 cm on either side of the sensor. The test results showed that the sensor was completely successful in meeting the criteria. The sensor detected the line accurately and consistently in all three positions. The device stopped and turned away from the line as soon as the sensor crossed the threshold. The sensor data showed that the deviation from the line was within the 1 cm range in all cases. The test results confirmed that the line following sensor was reliable and effective in restricting the device to the simulation environment.

3.1.4. Motors

The motors were tested by measuring 30 centimeters, and ensuring the robot did not complete the 30 centimeter distance before 1 second, as the maximum speed of the robot should be limited to 30 cm/s. The acceptance criteria for the motor speed was being able to accurately adjust the speed of the device to within 1 cm/s, and in our case we have met the criteria. The robot did not travel 30 centimeters faster than a second, but rather the robot is configured to travel at a considerably slower rate so as to not pull as much power from the batteries and to give time for other sensors to accurately read and gather values.

3.2. Integration Testing

3.2.1. Object Avoidance

We conducted integration testing to examine the behavior of our device when it encountered objects of various sizes in the testing environment. Our project proposal specified four different scenarios for object placement: left, right, front, and back. The results of the object avoidance testing were inconclusive for the scenarios where the object was placed on the left, right, or back of the plow. This was because we changed our original proposal to only detect objects in front of the vehicle, using an infrared light sensor. For the scenario where the object was placed in front of the plow, we successfully avoided the object by stopping and turning right as the object came close. However, we noticed that if the motors were moving too fast, we could stop too late and collide with the object. Therefore, we adjusted the maximum speed of the plow to prevent this from happening.

3.2.2. Path Finding

We performed the path finding testing as described in our project proposal. The test involved two scenarios: one where there were no objects present, and another where specific stationary objects were strategically placed to assess the expected outcomes of the path finding algorithm. The test results showed that the path finding algorithm was mostly successful in both scenarios.

1. Scenario 1: No Objects (Baseline Test)
    a. The robot followed the expected path and completed the snow removal task as specified by the algorithm.

    b. The robot's behavior, sensor data, and algorithm execution matched the anticipated outcomes. It was capable of avoiding driving over the black line.
    c. The robot did not encounter any issues or deviations in this scenario.

2. Scenario 2: Objects in Precise Locations
    a. The robot detected and avoided the objects by altering its path and performing the snow removal around them.
    b. The robot's behavior, sensor data, and algorithm execution mostly matched the expected outcomes. However, there were some minor discrepancies in the robot's response to some objects. For example, the robot sometimes stopped too close or too far from the objects, or took a longer or shorter route than expected. These discrepancies were likely due to the variations in electrical power to the sensors and the motors. We adjusted the parameters of the algorithm and the sensors to minimize these discrepancies and improve the accuracy and efficiency of the robot's navigation.

Overall, the path finding testing was mostly successful, as the robot accurately followed the path finding algorithm under both scenarios: with no objects and when objects were placed. The algorithm effectively responded to obstacles and navigated as expected. We documented and analyzed the test results, highlighting any issues or discrepancies. We also suggested some possible improvements and recommendations for future work.

4. Customer Testing

The customer testing demonstration presented several challenges. Initially, the device was experiencing a hardware failure and did not power on. This was solved by replacing the batteries as they were found to not have enough voltage. With newly recharged batteries, the device became operational. This resulted in a slight delay to the first round of the demonstration. Once the device was used in the demonstration, it quickly became apparent that there was another unresolved issue as it continued to rotate in place indefinitely. This resulted in a failed first demonstration. Before the second round of the demo, the issue was identified. There was a hardware failure on one of the line detector probes, causing the device to think there was a border when there was none. This was fixed in time for the second demo, which resulted in a mostly successful second round of testing. There were still some stuttering issues due to the interference of the motors but more importantly, the plow design was shown to be ineffective. The plow was originally meant to be 3D printed in time for the demo but due to delays it was not ready. The substitute plow used was made of cardboard and had an outer brace style which had a horizontal beam attached in front of the main surface. This was intended to allow the cubes to be pushed further outside the testing area borders but instead resulted in issues. The cubes were able to go under the brace and got stuck between it and the main surface of the plow. Due to the success criteria of the demo, these cubes were counted as they did go past the border, however they became stuck in the plow and required manual removal.

Overall, the project was mostly successful as the device was able to clear a reasonable amount of cubes from the testing area. However, this showed many of the challenges and issues with the device. Given further development time, more work should be done to resolve outstanding issues. For example, using CAN bus instead of I2C would be preferable as it is more resistant to noise, and using properly regulating the voltage across the components with capacitors would improve system stability.

5. Appendix A

1. Project Charter

1.1. Objective

The objective of this project is to develop an autonomous snow removal device. This device is to function based on a strict set of criteria which is defined in the list of requirements, and must be capable of identifying and clearing snow from a designated area. Additional limitations are imposed on the device's physical size, speed, and a time limit for task completion. The goal of the project is to implement hardware and software solutions to achieve the desired functionality. This includes developing solutions for object avoidance, boundary detection, maneuverability challenges, and the snow removal process. The success of the project will be evaluated by using a controlled environment with wooden cubes as a substitute for snow.

1.2. Deliverables

The project deliverables are listed in the list of deliverables. They include but are not limited to a progress report, a final report, and a live demonstration of the device in a simulated testing environment. Major milestones consist of the key stages of development such as object detection and avoidance, line sensing capabilities, and effective motor control software. The project will make use of GitHub to store and share software among the team members.

2. Scope

2.1. List of Requirements

2.1.1. Border Line Detection

During the snowplow's navigation, the snowplow's control and guidance system must continuously monitor the ground to detect the presence of a black line. When the line detector sensor detects that the snowplow is within 5mm of the black line, the snowplow must come to an immediate stop and then execute a safe turning maneuver to move away from the detected black line, ensuring it remains within the defined area

2.1.2. Object Avoidance

During the snowplow's navigation, the snowplow's obstacle avoidance system must employ motion sensors to detect the presence of objects. When motion sensors identify an object within a 1m the snowplow will react by slowing down. Once the snowplow has reached the critical distance of 5cm from the plow the snowplow must stop and turn to avoid collision with the object.

2.1.3. Speed Constraint

When the snowplow is moving, the control system must monitor the speed of the snowplow. If the snowplow's speed exceeds 30 cm/s then the snowplow must not increase the speed of the motors any further.

2.1.4. Snow Removal

When the snowplow is moving, the snowplow's snow removal mechanism must be capable of moving the snow blocks just 1cm outside of the black border line without the snowplow crossing the line.

2.1.5. Plow Blade

The plow blade must be designed to effectively handle wooden cubes with a 20mm side length. The plow's width must be 200mm to ensure that a maximum of 10 cubes lined up can be moved by the snowplow and that the plow blade is larger than the snowplow itself.

2.1.6. Testing

Conduct thorough testing to evaluate the performance of sensors in the wooden cube handling plow, ensuring that the plow meets the following criteria:

- Measure and verify that the plow maintains a maximum speed of 30 centimeters per second during operation.
- Validate the plow's ability to detect objects at a distance of up to 2 meters.
- Confirm that the plow can detect objects as close as 5 centimeters.
- Test and ensure that the plow's sensors prevent it from crossing the outer border lines by more than 5mm

2.2. List of Deliverables

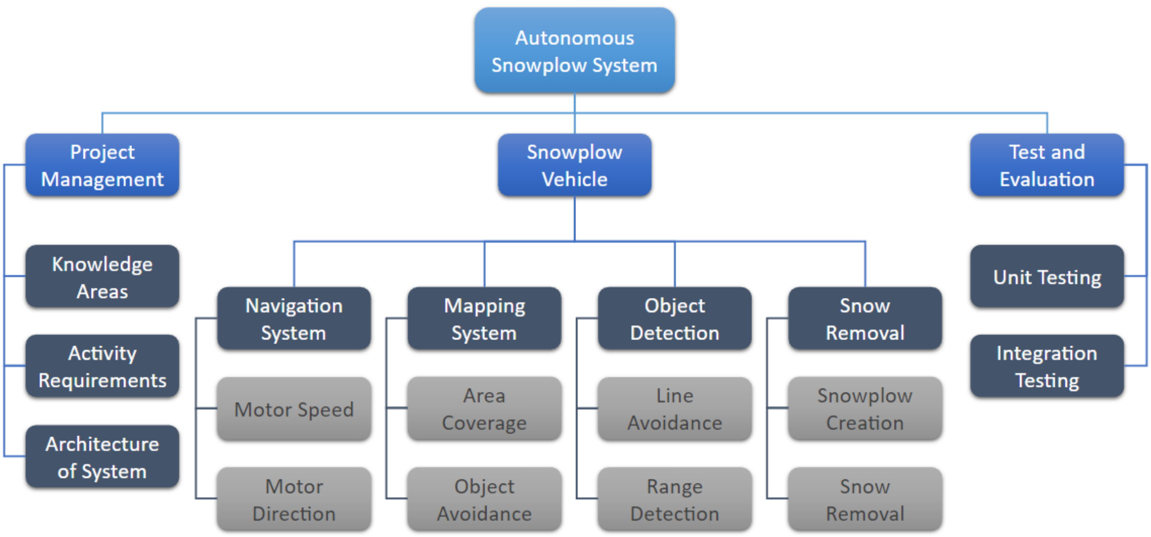


Figure 1: Work Breakdown Structure

2.3. Testing Plan

2.3.1. Unit Testing

2.3.1.1. Time of Flight Sensor

The goal of this test plan is to verify the accuracy and consistency of the time-of-flight sensor's distance measurements. We will conduct tests at specified distances ranging from 1cm to 100cm, including 1cm, 10cm, 25cm, 50cm, 75cm, and 100cm. For each distance, we will place an object, manually measure the actual distance, and compare it to the sensor's detected distance. The test results will be documented and analyzed to assess the sensor's performance, with success defined as the sensor accurately measuring distances within an acceptable tolerance of ±1cm.

2.3.1.2. Ultrasonic Sensor

This test plan aims to verify the precision and reliability of the ultrasonic sensor in measuring distances within a shorter range, specifically from 1mm to 10cm, with increments of 1mm, 5mm, 1cm, 2cm, 5cm, and 10cm. For each distance point, an object will be placed, and the actual distance will be manually measured. Subsequently, we will compare the manually measured distance to the sensor's detected distance. The test results will be documented and evaluated to assess the sensor's performance, with success defined as the sensor accurately measuring distances within an acceptable tolerance of ±1mm for close-range applications

2.3.1.3. Line Following Sensor

This sensor will be used to restrict the movements of the device to the designated simulation environment. It will be tested by placing it on the left, in the center, and on the right of the boundary and the success scenario is that the sensor will be able to detect if it has crossed the threshold. Acceptable performance criteria is that the device is accurate in detecting a line within 1 cm on either side of the sensor.

2.3.1.4. Motors

The motors will be evaluated to determine the effective output range with a goal of ensuring a maximum speed of 30 cm/s. The motors will be tested in both forward and reverse from 10% to 100% power in increments of 10%. The speed will be measured manually by timing how long the device takes to cross a specific distance. Success will be defined as being able to accurately adjust the speed of the device to within 1 cm/s.

2.3.2. Integration Testing

2.3.2.1. Object Avoidance

Object avoidance will be tested by setting up objects of various sizes inside a testing environment and examining the behaviour of the device. Data from the time of flight sensor, ultrasonic sensor, line follower sensor, and the motors will be collected throughout the test to help in determining any possible issues. Furthermore, objects will be placed to the left, right, front, and back of the device so as to construct a labelled dataset for each scenario. The success scenario is defined as being able to develop an algorithm that will continuously detect objects in each direction and report the current state of the device’s surroundings.

The output of this algorithm will be manually compared with the actual placement of the objects to determine its effectiveness.

2.3.2.2. Path Finding

The primary goal of this test plan is to evaluate the integration of all sensors in the path finding process to ensure successful navigation of the snow removal area. The test will involve two scenarios: one where there are no objects present, and another where specific stationary objects are strategically placed to assess the expected outcomes of the path finding algorithm.

1. Scenario 1: No Objects (Baseline Test)
    a. In a controlled environment with no obstacles, initiate the path finding algorithm and monitor the robot's navigation.
    b. Confirm that the robot follows the expected path and completes the snow removal task as specified by the algorithm.
    c. Document the performance and ensure that the algorithm executes as anticipated.

2. Scenario 2: Objects in Precise Locations
    a. Introduce stationary objects in predefined and strategic locations within the snow removal area.
    b. Activate the path finding algorithm and observe the robot's response to the objects.
    c. Verify that the robot detects and avoids the placed objects by altering its path and performing the specified snow removal around them.
    d. Document and compare the actual outcomes with the expected outcomes to assess the effectiveness of the path finding algorithm in response to obstacles.

Record the robot's behavior, sensor data, and any deviations from expected outcomes during both scenarios. Analyze the recorded data to evaluate the success of the path finding algorithm in navigating with and without objects. Document the test results, highlighting any issues or discrepancies. The test will be considered successful if the robot accurately follows the path finding algorithm under both scenarios: with no objects and when objects are placed. The algorithm should effectively respond to obstacles and navigate as expected

3. Schedule

3.1. List of Activities

Table 1 summarizes the tasks for creating our Autonomous Snowplow robot. These tasks are based on the requirements listed in section 2.1. The table shows milestones with a breakdown of tasks it consists of. As a group of 4, we are expected to have a total of 20 tasks between all milestones.

Table 1: List of Activities

Milestone	Task

Object Detection	Configure and Connect Time of Flight Sensor to robot
	Configure and Connect Ultrasonic Sensor to robot
	Write handler code for Flight Sensor
	Write handler code for Ultrasonic Sensor
	Integrate motor code for object avoidance
Line Avoidance	Configure and Connect Line Following Sensor to robot
	Write handler code for Line Following Sensor
Robot Motors	Configure and Connect Robot’s 4 motors
	Write and Configure motor speed to move at a max of 30 cm/s
	Write Handler code to change motor direction given turn angle
	Write Handler code to change motor speed when approaching object
Path Finding	Create algorithm to move in the most efficient path for snow removal
	Create algorithm to ensure snow removal around obstacles
Snow Plow	Create Snow Plow
	Attach Snow Plow to Robot
Finishing Up Robot	Configure Watchdog timer to avoid infinite loops
	End to end testing
Documents and Presentation	Progress Report
	Final Report
	In-Lecture Presentation

3.2. Schedule Network Diagram

The following diagram illustrates our schedule network diagram, starting from the project start date all the way to project completion. The diagram is also available for viewing at [this link](). Given that there is no path with total floats being 0's, we do not have a critical path.

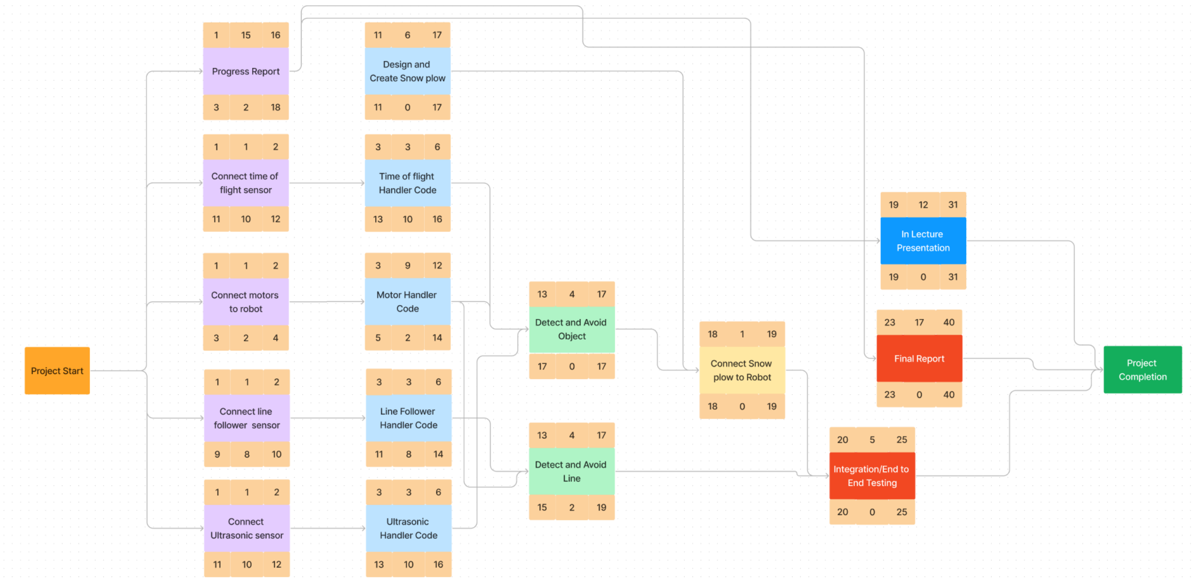


Figure 2: Schedule Network Diagram

3.3. Gantt Chart

The following Gantt Chart displays project milestones and deadlines. The project start date will occur after reading week, which would be October 30th. The project end date is set to December 5th as it is when the final report is due, however earlier milestones will have target dates according to corresponding deliverables. The chart is also available for viewing at [this link]().

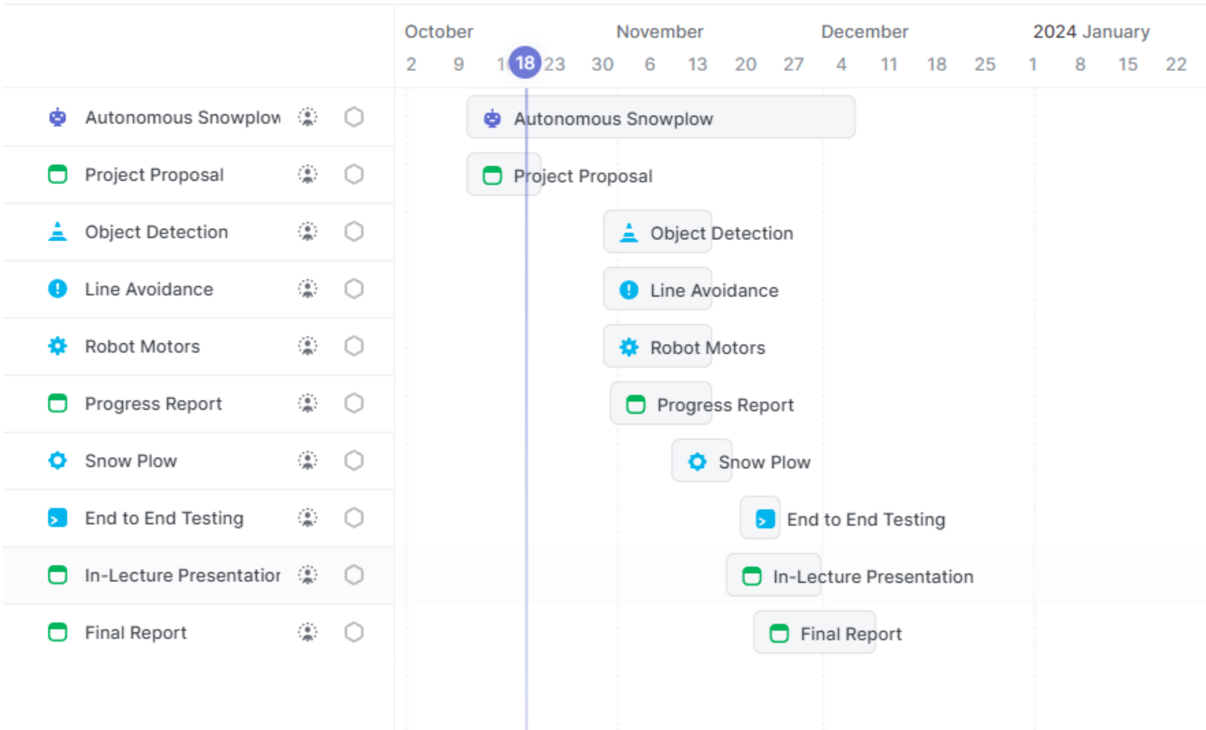


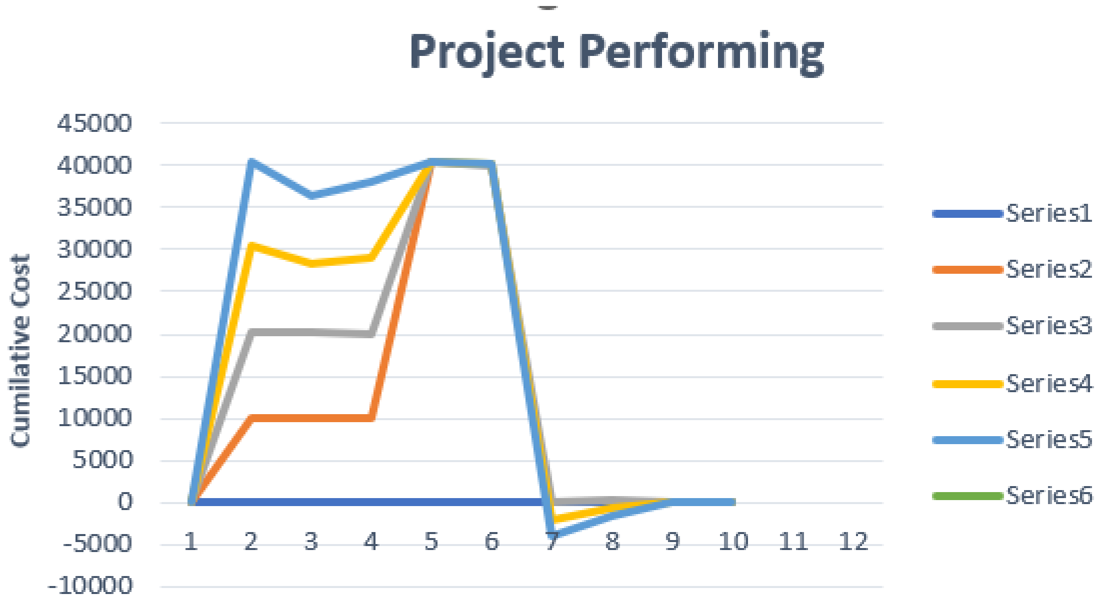
Figure 3: Project Timeline Gantt Chart

4. Cost

Assuming our project was planned to be completed in 4 weeks (160 hours) with a budget of $40,500 at $50/hr per developer

If We're at the end of week 2 (80 hours), The EVA metrics would be

- **Planned Value (PV)**: Our PV would be 50% of the Budget at Completion (BAC), which is $20,250.
- **Earned Value (EV)**: If we completed 60% of the project, my EV would be 60% of the BAC, which is $24,300.
- **Actual Cost (AC)**: If we;ve actually spent $22,000 by the end of week 2, then my AC is $22,000.
- **Budget At Completion (BAC)**: Our overall authorized budget. BAC = sum of PV = $40,500
- **Estimate At Completion (EAC)**: Our EAC = BAC + (AC – EV) = $40,500 + ($22,000 - $24,300) = $38,200
- **Schedule Variance (SV)**: Our SV = EV – PV = $24,300 - $20,250 = $4,050
- **Cost Variance (CV)**: Our CV = EV - AC = $24,300 - $22,000 = $2,300
- **Schedule Performance Index (SPI)**: Our SPI = EV/PV = $24,300 / $20,250 = 1.2
- **Cost Performance Index (CPI)**: Our CPI = EV/AC = $24,300 / $22,000 = 1.1



5. Human Resource

Table 2: Responsibility Assignment Matrix

Milestone	Task	Assignee	Approver
Object Detection	Configure and Connect Time of Flight Sensor to robot	Eric	Josh
	Configure and Connect Ultrasonic Sensor to robot	Zakariyya	Josh
	Write handler code for Time of Flight Sensor	Eric	Josh
	Write handler code for Ultrasonic Sensor	Aaryan	Josh
	Integrate motor code for object avoidance	Aaryan	Eric
Line Avoidance	Configure and Connect Line Following Sensor to robot	Josh	Zakariyya

	Write handler code for Line Following Sensor	Josh	Zakariyya
Robot Motors	Configure and Connect Robot’s 4 motors	Zakariyya	Josh
	Write and Configure motor speed to move at a max of 30 cm/s	Aaryan	Josh
	Write Handler code to change motor direction given turn angle	Zakariyya	Aaryan
	Write Handler code to change motor speed when approaching object	Zakariyya	Aaryan
Path Finding	Create algorithm to move in the most efficient path for snow removal	Josh	Zakariyya
	Create algorithm to ensure snow removal around obstacles	Josh	Eric
	Attach and Configure Gyroscope	Josh	Aaryan
	Write Handler code for Gyroscope	Zakariyya	Josh
Snow Plow	Create Snow Plow	Aaryan	Eric
	Attach Snow Plow to Robot	Eric	Aaryan
Finishing Up Robot	Configure Watchdog timer to avoid infinite loops	Eric	Everyone
	End to end testing	Eric	Everyone
Documents and Presentation	Progress Report	Everyone	Everyone
	Final Report	Everyone	Everyone

	In-Lecture Presentation	Everyone	Everyone