Abstract

This report presents the Phase A design of an autonomous delivery robot required to carry a preloaded package across astroturf, woodchips, gravel, and stone, place it inside the lobby boundary, and stop within 150 seconds without human intervention. Requirements were translated into measurable engineering specifications and evaluated through a House of Quality, functional decomposition, and weighted concept selection. Based on the evaluated criteria, concepts were proposed with Computer-Aided-Design (CAD), design-for-manufacture review, and Failure Mode and Effects Analysis. The selected architecture is a four-wheel skid steer platform with 65 mm wheels, a 150 mm x 110 mm x 28 mm printed PLA chassis, four TT gearmotors, an Arduino Due, three HC-SR04 ultrasonic sensors, two L298N drive modules, and a 28BYJ-48/ULN2003 rack and pinion pusher. The decision matrix ranked this concept at 86%, above a track-based concept at 73% and rear wheel drive concept at 60%.

The robot chassis and designed subcomponents are 3D printed from PLA and fastened by select M series threads and bolts. During informal testing, the robot displayed success in terrain navigation and package delivery via pusher. On the day of the demonstration, the robot's DC motor stalled when performing a turn on the astroturf. This was a result of an uncharged battery which could not supply adequate power to the motors. The robot will be redesigned to improve reliability, to reduce the length and width of the chassis, to have the ability to pick up a package, and simplify the autonomous navigation system.

Table of Contents

Abstract 2

Table of Contents 3

List of Figures 6

List of Tables 6

1.0 Introduction 8

1.1 Project Background 8

1.2 Phase A Problem Statement and Objective 9

1.3 Scope and Boundaries 10

2.0 Background Search 10

3.0 Requirements and Engineering Specifications 11

3.1 Engineering Specifications and Acceptance Criteria 13

3.2 House of Quality 14

4.0 Design Plan and Project Management 15

4.1 Work Breakdown and Dependencies 15

4.2 Schedule Control 16

5.0 Concept Generation and Selection 16

5.1 Functional Decomposition 16

5.2 Brainstorming 17

5.2.1 Terrain Mobility 17

5.2.2 Package Maneuvering 19

5.3 Integration of Conceptualized Subsystems 20

5.4 Concept Screening and Selection 21

5.5 Selected Concept and Rationale 21

5.6 Components and Materials Identification and Selection 22

5.6.1 Sensor Requirement 22

5.6.2 Actuator Requirement....22
5.6.3 Sensors of Consideration....22
5.6.4 Actuators of Consideration....22
5.6.5 Materials of Consideration....23
6.0 Detailed Design and Engineering Analysis....24
6.1 Mechanical Design....25
6.2 Electrical and Power Design....25
7.0 Robot Functionality and Automation Algorithm....27
7.1 Operating Sequence....27
7.2 Sensors, State Estimation, and Decision Logic....28
7.3 Motion and Control Strategy....29
7.3.1 Drive....29
7.3.2 Turning....30
7.3.3 Stopping Logic....30
7.4 Pseudocode and Software Structure....30
8.0 Manufacturing Assembly Economy and Sustainability....31
8.1 Design for Manufacturing....31
8.2 Design for Assembly and Disassembly....32
9.0 Safety and Risk Analysis....32
9.1 Failure Modes and Effects Analysis (FMEA)....33
10.0 Engineering Documentation....34
11.0 Results Discussion and Product Validation....36
11.1 Component and Subsystem Results....36
11.2 Full Phase A Mission Results....38
11.3 Phase A Demo Outcome....38
11.4 Requirement Compliance Matrix....39
11.5 Discussion Limitations and Lessons Learned....41
12.0 Phase B Development Plan....41

12.1 Required Design Changes 42
12.2 Phase B Tasks and Delegates 42
13.0 Conclusion 43
References 44
Appendices 46
APPENDIX A Gantt Chart 46
APPENDIX B Engineering Drawing Set 47
APPENDIX C Approval Records 49
APPENDIX D Front Sensor Code Block (AI ASSISTED) 50

List of Figures

Figure 1. Phase A terrain course and lobby zone [2] 9
Figure 2. Lunar Rover Optimization Platform [4] 11
Figure 3. House of Quality linking Phase A requirements to engineering specifications 14
Figure 4. Functional Decomposition 17
Figure 5. Tracked terrain mobility concept 17
Figure 6. Rear wheel drive terrain mobility concept 18
Figure 7. Four-wheel drive terrain mobility concept 18
Figure 8. Two degree-of-freedom claw with rotational lift 19
Figure 9. Two degree-of-freedom claw with linear lift 19
Figure 10. Rack and pinion package pusher concept 20
Figure 11. Final Architecture 25
Figure 12. Phase A Circuit Schematic 27
Figure 13. Detailed vehicle flowchart (DISCLAIMER: AI ASSISTED IMAGE USING TABLE 9) 28
Figure 14. Software Structure logic (DISCLAIMER: AI ASSISTED USING TABLE 9) 31
Figure 15. Exploded view and subassembly identification 35
Figure 16. Rendered 3D View 36

List of Tables

Table 1. Scope and Boundaries 10
Table 2. Project Requirements 12
Table 3. Phase A Constraints 13
Table 4. Engineering Specifications and Acceptance Criteria 13
Table 5. Work Breakdown 15
Table 6. Integrated Robot Concepts 20
Table 7. Weighted Decision Matrix 21
Table 8. Selected Components and Materials 23
Table 9. Logic Thresholds 29
Table 10. Design for Manufacture Decisions 32
Table 11. Phase A FMEA 33
Table 12. Phase A Bill of Materials 34
Table 13. Component Tests 37

Table 14. Subsystem tests 37
Table 15. Demo Criterion 38
Table 16. Compliance Matrix 39
Table 17. Phase B General Task Delegation 42

1.0 Introduction

The purpose of this project is to develop a robot capable of autonomous navigation, package delivery, package pick-up and package transportation. The project has been undertaken within the “Mechatronics Design” course at Ontario Tech University and aims to develop systems capable of autonomously demonstrating solutions to present day challenges in logistics and distribution. The project began in September 2026 and will conclude in December 2026, the project aims to demonstrate concurrent engineering principles, a physical prototype, and relevant engineering documentation.

The project is split into three phases; Phase A, Phase B, and Phase C. Phase A requires the robot to autonomously transport a package over a varying terrain and drop the package off at the prespecified location. Phase B requires the robot to navigate a maze, pick up a package, and maneuver the package out of the maze. Phase C is a combination of Phase A and Phase B, the robot must navigate the varying terrain, access the maze, retrieve the package, and return to the varying terrain starting location. This report will follow the development of the robot specifically for Phase A, while considering the future development for Phase B and Phase C.

1.1 Project Background

The accelerating growth of e-commerce has pressured the distribution and logistics industry to maximize efficiency with how products are shipped, handled, and delivered. Online shoppers can order physical products virtually and receive them within days, which shortens the time available to move a product from the distributor to the customer. Much of this pressure falls on the "last mile", where a large portion of total logistics cost is found. The “last mile” refers to the final stage of delivery, in which a product is moved from a distribution center to the customer's location [1].

To meet this growing logistics demand, autonomous robots are being developed to perform these tasks with greater efficiency. Such robots may receive packages from a distributor, track and manage package locations within a distribution depot, or complete the last mile of the logistics chain by delivering packages directly to customers. Autonomous robots rely on a wide array of sensors and actuators for obstacle avoidance, localization, and physical manipulation, as well a strong interface between mechanical, electrical, and software systems to ensure both logistics efficiency and public safety.

The operating environment of these robots varies with the application. Notable environments include sidewalks and roadways, distribution depot floors, and gravel or dirt pathways in rural areas. As terrain varies, a single autonomous system capable of traversing all these surfaces is of particular interest. The robot may also operate in public settings or alongside distribution center workers. Ensuring controlled operation over a varying terrain and maintaining human safety are key design considerations for such a system.

The challenges faced by these autonomous robots motivate continued development of robust autonomous navigation, reliable obstacle detection, efficient energy management, and user-friendly interaction. These elements form the foundation of the autonomous delivery robot which will be presented in this project.

1.2 Phase A Problem Statement and Objective

Phase A of this project relies on the development of a robotic system capable of autonomously navigating a varying terrain, while delivering a pre-loaded package to a specific drop-off location. The testing area for this phase is a rectangular box with the flooring surfaces of astroturf, woodchips, gravel, and stone. The rectangular box has approximate dimensions of 120 cm in length and 80 cm in width; the package drop-off zone being on the diagonal of the starting corner location. The time between the operation of the robot and the drop-off of the package shall not exceed a time limit of 150 seconds.



Figure 1. Phase A terrain course and lobby zone [2]

1.3 Scope and Boundaries

The scope and boundaries presented in table 1 are to clarify the differences between Phase A and Phases B and C; this is presented to avoid overcomplication and ensure a clear design basis.

Table 1. Scope and Boundaries

Included in Phase A	Outside Phase A or reserved for later phases
Autonomous terrain traversal from the specified start point	Basement maze navigation
Package carried at the start and autonomously placed at the lobby	Initial package pickup and stacked-package retrieval
No obstacles and no direction constraint	Unknown obstacle avoidance
Performance within 150 seconds	Phase B and C mission time limits

2.0 Background Search

Existing autonomous delivery robot designs combine a wheeled chassis, electric propulsion, onboard computing, environmental sensors, and a cargo handling system. Sibley's delivery system patent describes a robot that autonomously travels between locations while carrying an object in a container and using a computer-controlled mechanism to deliver it [3]. For Phase A, this can be simplified into four main subsystems; mobility, sensing and control, package retention, and package release. A lightweight holding mechanism with a controlled release mechanism is much more appropriate for the scale of the project than a large, enclosed compartment.

Terrain mobility depends strongly on wheel dimensions, traction, ground clearance, and wheel loading. NASA's Lunar Rover Optimization Platform was developed to study variables including slope ascent, obstacle traversal, drawbar pull, wheel loading, and center of gravity positioning [4]. This research shows that mobility components should not be selected independently. Larger wheels and greater ground clearance can improve the robot's ability to cross terrain, but they can also increase mass, rotational inertia, and motor torque requirements. Therefore, chassis geometry, wheel design, motor gearing, and component placement should be evaluated together using calculations and physical terrain testing.



Figure 2. Lunar Rover Optimization Platform [4]

A differential drive system is suitable for small prototypes because it provides simple steering and allows the robot to turn within a small area. However, navigation based only on wheel rotation can accumulate position and heading error. Borenstein and Feng identified uncertainty in the effective wheelbase and unequal wheel diameters as two major sources of systematic error [5]. Their results demonstrate the importance of experimental calibration. For Phase A robot, commanded travel distances and turning should be calibrated through repeated tests, while sensor feedback should be used to correct the error.

3.0 Requirements and Engineering Specifications

The project guidelines and Phase A demonstration requirements were first separated into individual, atomic statements. Table 2 identifies the robot's functional and physical requirements, mandatory project constraints, and potential design opportunities. Each item is assigned an identifier, so it can be traced throughout the engineering specifications, concept selection, design analysis, and testing stages. The source and verification method are also provided to demonstrate how each requirement was confirmed.

Table 2. Project Requirements

ID	Type	Atomic requirement	Priority	Source	Verification
F1	Functional	The robot shall navigate autonomously from the specified Phase A start point to the lobby zone.	Necessity	Project guide 3.1; demo brief 1.2	System test
F2	Functional	The robot shall traverse every terrain included in the Phase A course.	Necessity	Project guide 3.1; demo rubric	System test
F3	Functional	The robot shall carry the package provided during terrain traversal.	Necessity	Project guide 3.1	Payload and system test
F4	Functional	The robot shall autonomously place the package inside the accepted lobby boundary.	Necessity	Demo brief 1.2, 1.3 and 6	Placement test
F5	Functional	The robot shall complete the full Phase A mission in no more than 150 seconds.	Necessity	Project guide 3.1 and 5.1	Timed system test
F6	Functional	The robot shall complete Phase A without a save move or human intervention after starting.	Necessity	Demo brief 1.3 and 5A	Observation and video
F7	Functional	The robot shall stop safely after releasing the package.	Necessity	Task interpretation	System test
P1	Physical	The payload interface shall accommodate the course package envelope of up to 0.25 kg and 50 mm per side, and the measured Phase A cube.	Necessity	Project guide 3.1; demo brief 4	Inspection and load test
C1	Constraint	The team shall not use a prefabricated robot chassis.	Necessity	Project guide 4.1 and FAQ Q2	Design inspection
C2	Constraint	Any added controllers, sensors, motors, or materials shall be submitted with pricing and specifications and approved before Demo A.	Necessity	Project guide 4.1; demo brief 5B	Approval record
O1	Opportunity	The design should improve robustness, manufacturability, assembly, economy, usability, and sustainability where feasible.	Luxury / differentiator	Project guide section 3	Design review

While table 2 defines what must be accomplished, table 3 shows the practical limitations involving the operating environment, available space, and project cost. The operating environment was a mixture of astroturf, woodchips, gravel, and stones. All of which were 120 cm by 80 cm space. Each group was provided with a kit to use, with cost being required to be minimal to ensure all

groups have an equal opportunity to perform. Table 3 also shows the impact each constraint had on key design decisions and what the verification process for each was.

Table 3. Phase A Constraints

ID	Constraint	Design impact	Verification
A1	Terrain	Wheel size, ground clearance, torque	Manual inspection
A2	Dimensions of the enclosure	Chassis dimensions and layout	Manual measurement using tape measure
A3	Cost	Outsourced parts used	Getting approval from Professor Seo

3.1 Engineering Specifications and Acceptance Criteria

The atomic requirements in table 2 were converted into measurable specifications in table 4 so that the same identifiers can be carried through concept selection, calculations, and testing. ES1-ES5 are direct acceptance criteria from the course and demonstration briefs [9], [2]; ES6-ES8 are team determined design targets that add quantitative margins for traction, positioning, and release. A specification is considered verified only when the method produces recorded results, not CAD in appearance or a single informal observation.

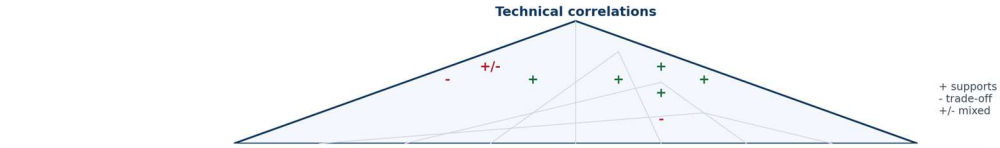
Table 4. Engineering Specifications and Acceptance Criteria

Spec ID	Related req.	Metric and unit	Target	Verification method	Basis
ES1	F5	Total mission time (s)	<=150 s	Timed end-to-end trials	Mandatory [9], [2]
ES2	F2	Terrain regions completed	100% of presented terrain	Video-supported system trials	Mandatory [9], [2]
ES3	F4	Package final position	Entire package inside lobby boundary	Overhead image and boundary check	Mandatory [2]
ES4	F6	Human interventions / save moves	0	Observer log and continuous video	Mandatory [2]
ES5	P1	Payload envelope	<=0.25 kg; <=50 mm per side	Scale and caliper inspection	Mandatory [9]
ES6	F2	Available / required wheel-torque ratio	>=1.5	Calculation plus worst terrain pull test	Team-selected margin

Spec ID	Related req.	Metric and unit	Target	Verification method	Basis
ES7	F1, F4	Stopping / placement error (mm)	<=50 mm	Ten approach-and-stop trials	Team-selected target
ES8	F4	Successful releases	>=9 of 10 trials	Repeated loaded release test	Team-selected target

3.2 House of Quality

To determine the weighted priorities of the engineering specifications, a House of Quality, as seen in figure 4, was used to map the customer requirements from table 2 to ES1-ES8. Relationship scores 9, 3, and 1 denote strong, moderate, and weak influence respectively. Weighted priorities show that positioning accuracy, release reliability, and intervention free autonomy dominate the design response. These results support the use of three directional sensors, a controlled stepper driven pusher, and a four-wheel platform with a traction margin target rather than selection of components from speed alone.



Requirement (WHAT)	Importance	ES1	ES2	ES3	ES4	ES5	ES6	ES7	ES8
F1 Autonomous navigation	5	3	3	3	9		3	9	1
F2 Traverse terrain	5	3	9		3		9	3	
F3 Carry payload	4		3		3	9	3	1	3
F4 Place in lobby	5	1		9	3	3		9	9
F5 <=150 s	5	9	1		3		3	1	3
F6 Zero intervention	5	3	3	3	9		3	9	9
F7 Safe stop	4	1			3			9	3
P1 Payload envelope	4		1	1		9	3		9
Target	-	**<=150 s**	**All terrain**	**Inside zone**	**0**	**0.25 kg / 50 mm**	**>=1.5**	**<=50 mm**	**>=9/10**
Weighted priority	-	**9.9%**	**9.6%**	**7.9%**	**15.9%**	**8.7%**	**11.4%**	**19.5%**	**17.0%**

Relationship: 9 = strong, 3 = moderate, 1 = weak, blank = none

Figure 3. House of Quality linking Phase A requirements to engineering specifications

The correlation roof also exposes tradeoffs. A large payload envelope can increase mass and reduce both acceleration and torque margins, while higher gearing can improve terrain capability at

the expense of top speed. The selected targets therefore require measured mass, motor-current, traction, and placement data before the design can be declared compliant.

4.0 Design Plan and Project Management

4.1 Work Breakdown and Dependencies

The purpose of this section is to identify the different tasks which comprise phase A of the project, as well as the subtasks which construct each main task. Table 5 summarizes the work breakdown.

Table 5. Work Breakdown

Task ID	Task	Subtask	Subtask ID	Dependencies
1	Project Initialization	Concept Generation	1.A	N/A
		Concept Selection	1.B	1.A
		Detailed Design of Concept	1.C	1.B
2	Design	Component Selection	2.A	1.C
		Design of Chassis	2.B	
		Purchasing Components	2.C	2.A and 2.B
		Constructing Chassis and Subparts	2.D	
		Programming and Troubleshooting	2.E	2.C and 2.D
3	Testing and Validation	Sensor Calibration and Validation	3.A	2.D
		Verify Autonomous Navigation System	3.B	

		Verify Package Manipulation System	3.C	

4.2 Schedule Control

During the early stages of the project, a Gantt chart was generated within Microsoft Excel. This Gantt chart depicts the deadlines for each subtask and project milestone, with the demonstration and report for Phase A concluding on October 6, 2026. The key bottlenecks which may affect the outcome of the project are the chassis construction and the component purchasing (purchasing and receiving). A one-week time slot has been allocated to ensure these topics have sufficient time to be constructed and shipped. The project aims to be constructed by September 27, 2026, with programming and troubleshooting occurring from September 28, 2026 - October 1, 2026. Appendix A contains a snapshot from September 23, 2026, and indicates the group is on a successful path to meet the October 6, 2026, deadline.

5.0 Concept Generation and Selection

5.1 Functional Decomposition

Functional decomposition was used to separate what the robot needs to do from component choice. Figure 5 shows the energy, information, and material flows between sensing, decision making, locomotion, structural support, payload retention, and package drop. This makes it possible to spot interface failures early on; the controller needs conditioned sensor information; each actuator needs a compatible power path, and the payload must remain mechanically supported until the release is done.

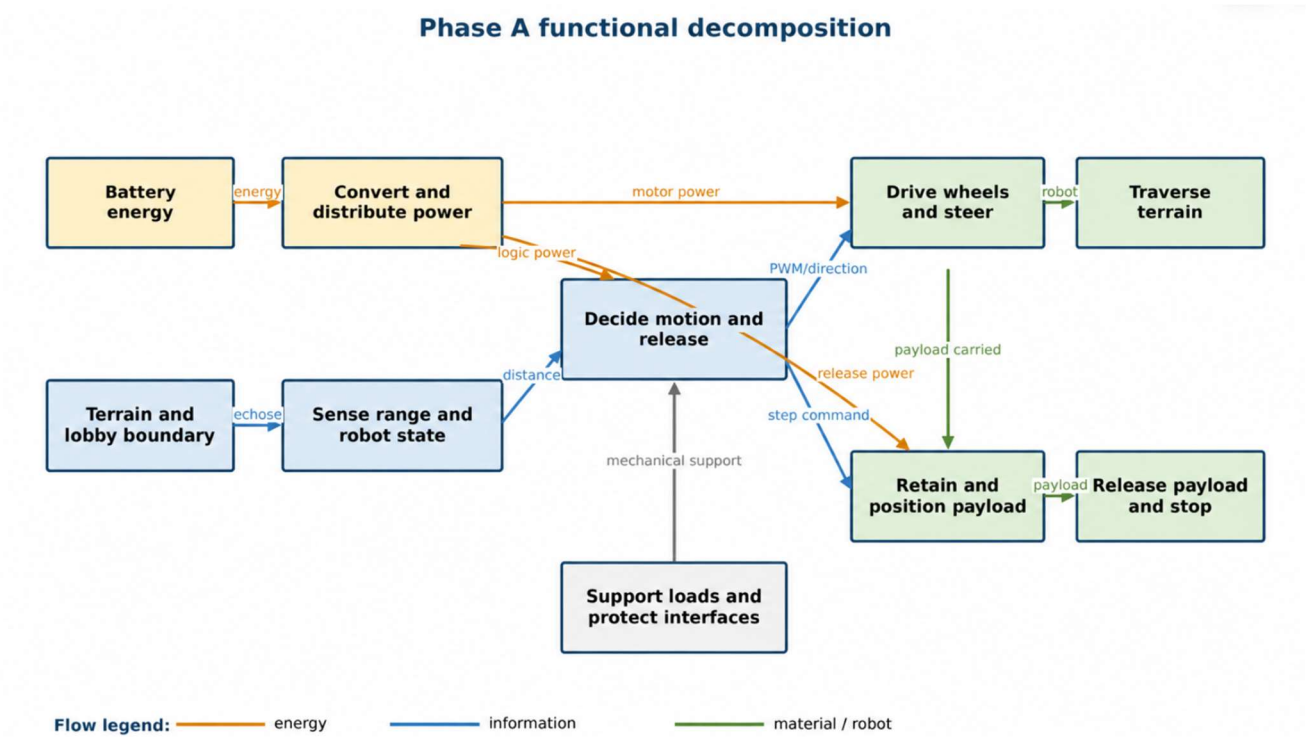


Figure 4. Functional Decomposition

5.2 Brainstorming

5.2.1 Terrain Mobility

As Phase A revolves around traversing a varying terrain, the first concept explored was a tank-like system. This system would feature two parallel tracks operating as a typical tank; both tracks moving one direction forward and reverse and opposing directions for turning. An open platform above the chassis between the tracks would be available to implement circuitry and systems.

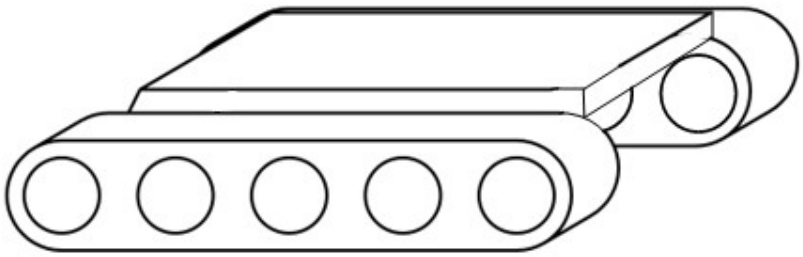


Figure 5. Tracked terrain mobility concept

The next design conceptualized was a two-wheel drive robot, two front wheels, and two driven rear wheels. The chassis would support the wheels in the typical four corner configuration, while accommodating circuitry and systems above the chassis between the wheels. The robot would function like a typical rear-wheel drive vehicle, with a steering mechanism angling the front wheels to turn.

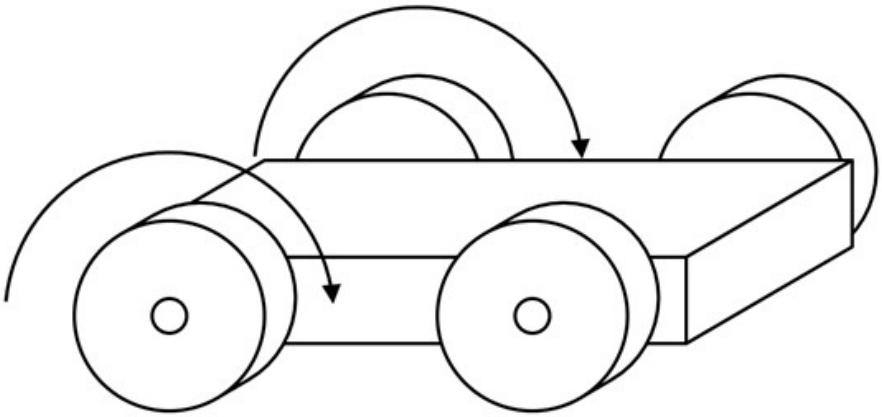


Figure 6. Rear wheel drive terrain mobility concept

The final concept for terrain mobility was a four-wheel drive robot, built like the previously discussed rear-wheel robot; however, the front and rear wheels would be driven. Space would be allocated for circuitry and systems above the chassis and between the wheels alike the previous idea. Steering may be accomplished by angling the front/rear wheels or opposing the side directions like the tank method.

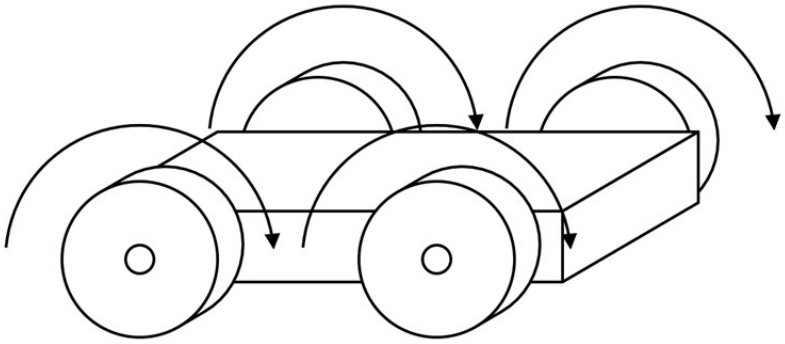


Figure 7. Four-wheel drive terrain mobility concept

5.2.2 Package Maneuvering

The first consideration for package maneuvering is a 2 Degree-of-Freedom claw-based mechanism. Such a mechanism would have an actuated claw end effector capable of closing to withhold a package and opening to release a package. The claw itself would then be vertically actuated to lift the package from the ground to be supported in the air. Two variations are conceptualized, both falling under this same description. One lifting the claw by a rotational joint and the other lifting the claw with a linear motion. The actuators would then support the package while the autonomous robotic system maneuvers the robot.

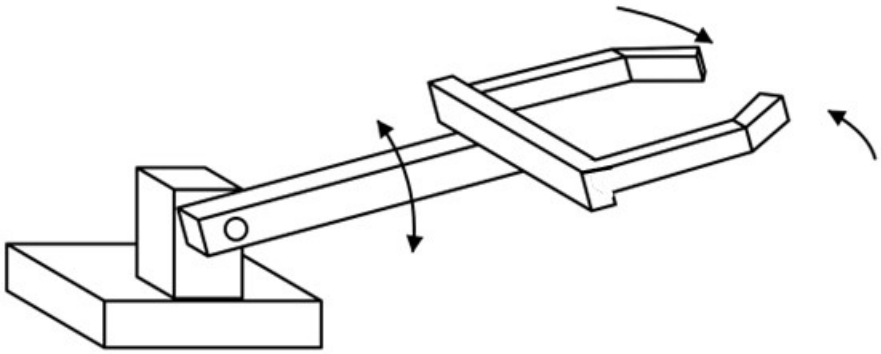


Figure 8. Two degree-of-freedom claw with rotational lift

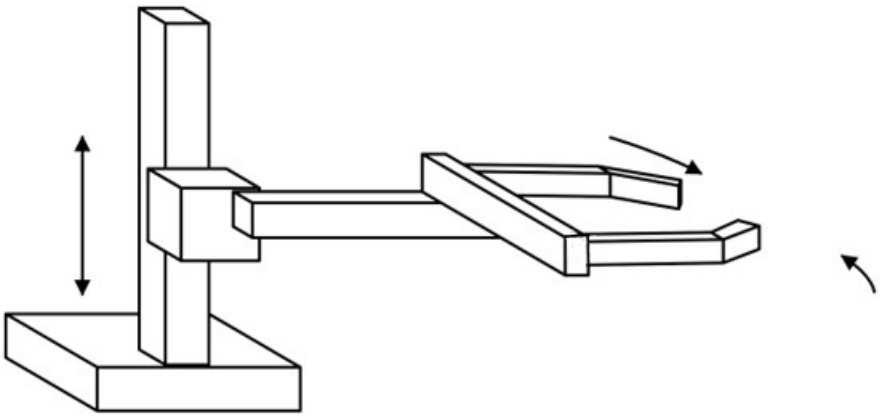


Figure 9. Two degree-of-freedom claw with linear lift

The next idea conceptualized for package maneuvering was specific to Phase A of the project, that being only to release the package. A pusher mechanism would push the package off the robot

when instructed. An actuator would activate to move the package; no means would be provided to retrieve a package.

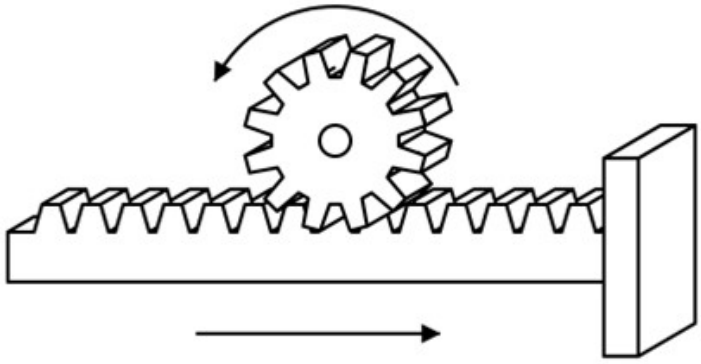


Figure 10. Rack and pinion package pusher concept

5.3 Integration of Conceptualized Subsystems

The independent mobility and package handling ideas were combined into three complete concepts so that each candidate could be evaluated at a system level. Table 6 deliberately pairs one mobility option with one package-handling option and shows the benefit and system level drawback. This avoids comparing isolated mechanisms that cannot complete the full mission.

Table 6. Integrated Robot Concepts

Concept	Mobility subsystem	Package subsystem	Primary strengths	Primary limitations
1	Dual tracked drive	Rotational-lift two-DOF claw	High contact area and obstacle conformity; positive package restraint	Highest part count, mass, control complexity, and fabrication effort
2	Rear-wheel drive with two passive front wheels	Linear-lift two-DOF claw	Low drive-part count and controlled placement	Limited loose-terrain traction; claw adds actuators and collision envelope
3	Four-wheel skid-steer drive	Rack-and-pinion pusher	Four driven contact patches, compact turning, simple one-axis release	Open-loop wheel slip; pusher must be aligned and travel-limited

5.4 Concept Screening and Selection

Each concept was rated from 1 to 5 against ES1-ES8, with them being unacceptable and excellent respectively. Weights sum to 100% and emphasize mandatory mission time, terrain traversal, placement, and autonomy. Weighted entries are the ratings multiplied by the percentage weight. Concept 3 ranks first at 86%, compared with 73% for concept 1 and 60% for concept 2.

Table 7. Weighted Decision Matrix

Engineering specification	Weight	Concept 1 rating	Concept 1 weighted	Concept 2 rating	Concept 2 weighted	Concept 3 rating	Concept 3 weighted
ES1 Mission time	20%	3	60	3	60	4	80
ES2 Terrain completion	20%	5	100	2	40	4	80
ES3 Lobby placement	15%	3	45	4	60	5	75
ES4 Zero intervention	15%	3	45	3	45	4	60
ES5 Payload envelope	5%	4	20	4	20	5	25
ES6 Torque / traction	10%	5	50	2	20	4	40
ES7 Position accuracy	5%	3	15	3	15	4	20
ES8 Release reliability	10%	3	30	4	40	5	50
Total	100%	-	365	-	300	-	430
Normalized score	-	-	73%	-	60%	-	86%

5.5 Selected Concept and Rationale

Concept 3 was selected because it provides the best balance of terrain capability, compactness, placement control, and manufacturability. Four driven wheels distribute tractive effort and retain a small turning radius while the rack and pinion pusher eliminates the extra joints and control states of a claw. The concept also fits with the available hardware: a printed chassis, four gearmotors, three ultrasonic sensors, and a stepper driven pusher. The CAD assembly confirms that the subsystems can fit on the 150 mm x 110 mm chassis without extending the package past the wheel. The electronics tower and removable sensor holders support quick access, while the pusher occupies the front left region beside the package.

5.6 Components and Materials Identification and Selection

5.6.1 Sensor Requirement

A primary requirement of this project is to have the delivery robot move autonomously; this will be achieved with sensor feedback. Sensors will be required to communicate information about the surrounding environment, ensuring the robot will not collide with an obstacle or wall. The sensors are required to transmit the depth of the surrounding environment.

5.6.2 Actuator Requirement

The actuators involved in this project include those to move about the terrain, and those to maneuver the package. Based on the selected concept for phase A, four motors are required for terrain mobility, and one actuator is required to operate the package pusher mechanism.

5.6.3 Sensors of Consideration

The primary requirement for the autonomous navigation sensors is depth control; the robot shall be able to detect the distance from surrounding objects and walls to avoid collision.

The first technique explored was using Infrared (IR) sensors capable of emitting and receiving IR signals, notably the GP2Y0E02A IR Range Sensor as it was provided in a course design kit. This sensor operates by emitting an IR wavelength and receiving the reflected wavelength at a deflected angle; the sensor uses the deflection angle to determine the distance the light was reflected at. The sensor can detect ranges from 4cm to 50cm, requires 2.7 volts for operation, and has a current consumption of ~26mA [10] [11].

The next technique explored was using an ultrasound sensor capable of emitting and receiving an ultrasonic soundwave; notably the SR04 Ultrasonic Ranging Module was of interest. This sensor operates by emitting an 8-cycle burst of ultrasound at 40 kHz [6], an adjacent "echo" pin will then "listen" for this ultrasound to be reflected and sent back towards the sensor module. The distance can be determined based on the time interval between frequency emission and reception. The sensor is capable of detecting distances of 2 cm to 400 cm, requires 5.0 volts, and has a working current of 15mA [7].

5.6.4 Actuators of Consideration

The first actuators explored for mobility were 6V 210:1 75RPM micro metal gearmotor with an encoder; these were provided in the course design kit. This actuator is a brushed DC motor with a

no-load output speed of 75RPM rated for 1.4 Kg-cm of torque; the motor operates with 6.0 volts and is rated for 170 mA [12].

The next actuators explored for traversing were TT 200RPM 3-6V DC Gearbox Motors. These actuators are simplistic brushless DC motors with a 1:48 gear box; they produce 0.15Nm ~0.60Nm with a maximum speed of 200RPM [13]. The motor operates between 3.0 volts and 6.0 volts and has a continuous no-load current of 150 mA [13].

The actuators of consideration to operate the pusher mechanism are an SG90 servo motor and a 28BYJ-48 5.0V stepper motor. The SG90 servo motor was included in the course design kit while the stepper motor was readily available from one of the group members. Both actuators operate at 5.0 volts; the SG90 has a 180° maximum turning radius and stalls at 1.2 kgcm, while the 28BYJ-48 stepper motor has no turning constraint and has a pull in torque of 2.9 N-cm.

5.6.5 Materials of Consideration

This prototype will make use of rapid manufacturing techniques, notably 3D printing. The readily available materials include: Polylactic acid (PLA), Acrylonitrile Butadiene Styrene (ABS), and thermoplastic polyurethane (TPU). Table 8 shows the selected components and their materials, along with the basis for the selections.

Table 8. Selected Components and Materials

Component/ Subsystem	Material	Actuator	Sensor	Basis
Chassis	PLA	N/A	N/A	PLA is cost-efficient, readily available, and reliable for the robot chassis
Sensor and Actuator Casings	PLA	N/A	N/A	PLA is cost-efficient, readily available, and reliable for the sensor casings
Package Pusher	PLA	N/A	N/A	PLA is cost-efficient, readily available, and reliable for the package pusher

Component/ Subsystem	Material	Actuator	Sensor	Basis
Navigation Sensors	N/A	N/A	HC-SR04 ultrasonic ranging module	The HC-SR04 range covers the Phase A enclosure; the GP2Y0E02A infrared option has a shorter useful range.
Mobility Motors	N/A	TT 200 rpm 3-6 V DC gearmotors	N/A	Four matched units were readily available; the two encoder gearmotors in the kit were insufficient for four-wheel drive.
Package Pusher Actuator	N/A	28BYJ-48 5 V stepper motor	N/A	Continuous step command and the available driver suit the required rack travel better than the SG90 servo's limited angular stroke.

6.0 Detailed Design and Engineering Analysis

Figure 12 presents the final physical and logical architecture. The Arduino Due is the controller, reading three HC-SR04 sensors and issuing directional commands to two L298N drive modules as well as step commands to the ULN2003 driver. The 9V source supplies the controller and sensors while the 7.4 V lithium ions supply the drive motors and stepper. All grounds are common, and the HC-SR04 lines are divided to protect the Due's 3.3 V only input [6] [7] [14] [15].

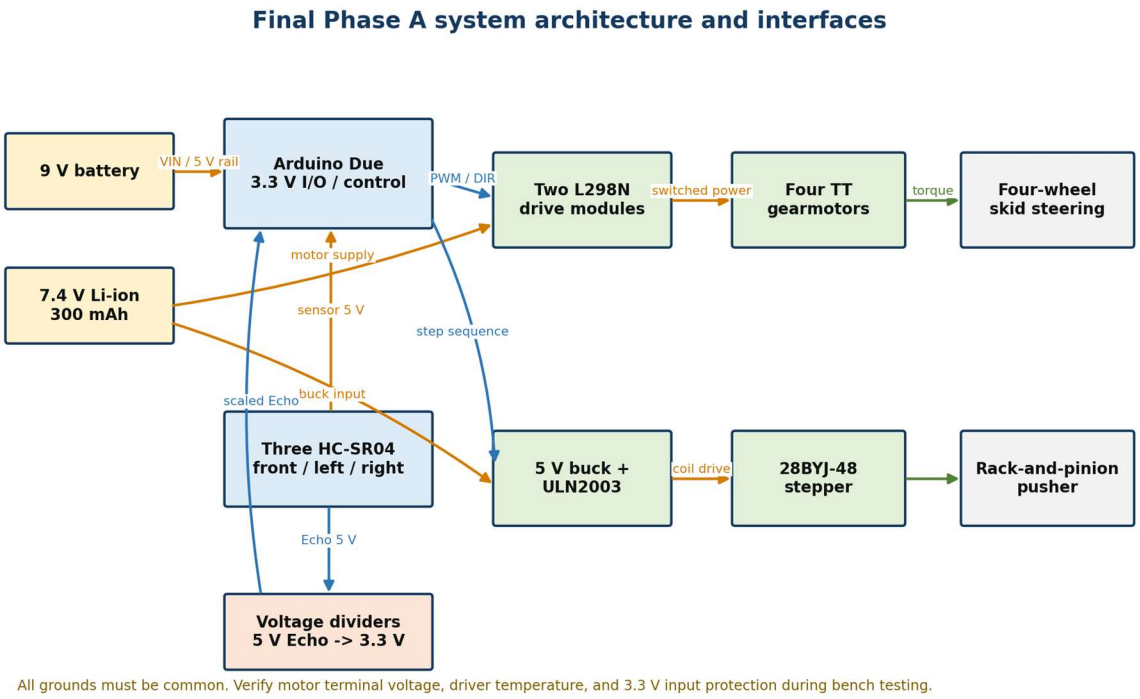


Figure 11. Final Architecture

6.1 Mechanical Design

The delivery vehicle is designed with an integral chassis to support all robot subsystems. The chassis is 150mm long x 110mm wide x 28mm high, all constructed with a 3D printer using PLA filament. The left and right sides of the chassis feature casings to support each of the four drive motors. Two holes on the top of the left motor casings and two behind the right have been placed for routing wiring. Both the left and right motor casing sides are designed to be enclosed with a panel, secured with 4 M2 threaded screws in the corners. The top of the chassis features cut outs displaying where each of the electrical subsystem components reside, each being a 2mm deep cut on the 3mm deep deck. The motor drivers and the microcontroller are given custom holding plates to ensure they do not shift while the robot is active. The front left of the top deck features a cut-out and ramp fit for holding and transferring an already placed package with the package pusher module. Two holes have been placed along the front, back, and sides; these hold the ultrasonic sensor's casings, all secured with a mix of M2 and M2.5 threads.

6.2 Electrical and Power Design

The robot is powered with a 9.0 V DC battery and a 7.4 V DC Li-ion 3000mAh rechargeable battery pack, the 7.4V battery to power the actuators and the 9.0V to power the sensors and

controller. A 5.0 V DC rail is generated from the microcontroller produced from the 9.0V DC battery at the controller barrel jack.

The four TT 3-6 V DC motors pull power directly from the 7.4V Li-ion battery through two L298N motor drivers; each side of the wheel is controlled via its own motor controller. The 7.4 volts battery is accepted to power the motors as the voltage over the motor driver is expected to drop 1.5-2.0V [14]. It was selected to use two motor drivers as the L298N driver has a maximum current rating of 2A, which can easily be surpassed by a stall current as the rating for one motor exceeds 1.0A [13], and the ease of access. The L298N motor driver logic is generated from the 5.0 V DC rail on the controller and is given H-bridge commands from the digital pins.

The ultrasonic sensors were originally specific to monitor the front, left, right, and back of the robot; however, this was later switched to only the front, left, and right due to the back sensor being damaged in delivery transit. The ultrasonic pins are powered from the 5.0 V DC rail of the controller and are connected to two digital I/O pins; a trig and an echo. The trig sends a signal to the sensor to release an 8-cycled ultrasonic wave of 40kHz, and searches for this wave reflection. The echo pins act as a receiver to collect and transmit this signal. However, the echo pin generates a 5.0 V DC output, exceeding the controller's 3.3 V digital pin input limit. To ensure the safety of the controller, a voltage divider was introduced per ultrasonic sensor to reduce the voltage to 2/3 of the original signal (5.0V to 3.33V). The resistors used for the divider were 1 kΩ and 2 kΩ, with the sensor's output received in parallel to the 2 kΩ while the 1 kΩ did the division.

The stepper motor is power from the 7.4 V Li-ion battery, the battery was buck converted from 7.4 V to 5.0 V with a step-down DC to DC convertor. The motor is controlled with a ULN2003 stepper motor driver which receives stepping configuration commands from the controllers' digital pins. Figure 13 shows the circuit diagram for the Phase A prototype. Due to limitations with the software, a four-pack AAA battery holder was used as a substitute for the 7.4 V Li-ion and buck converter module.

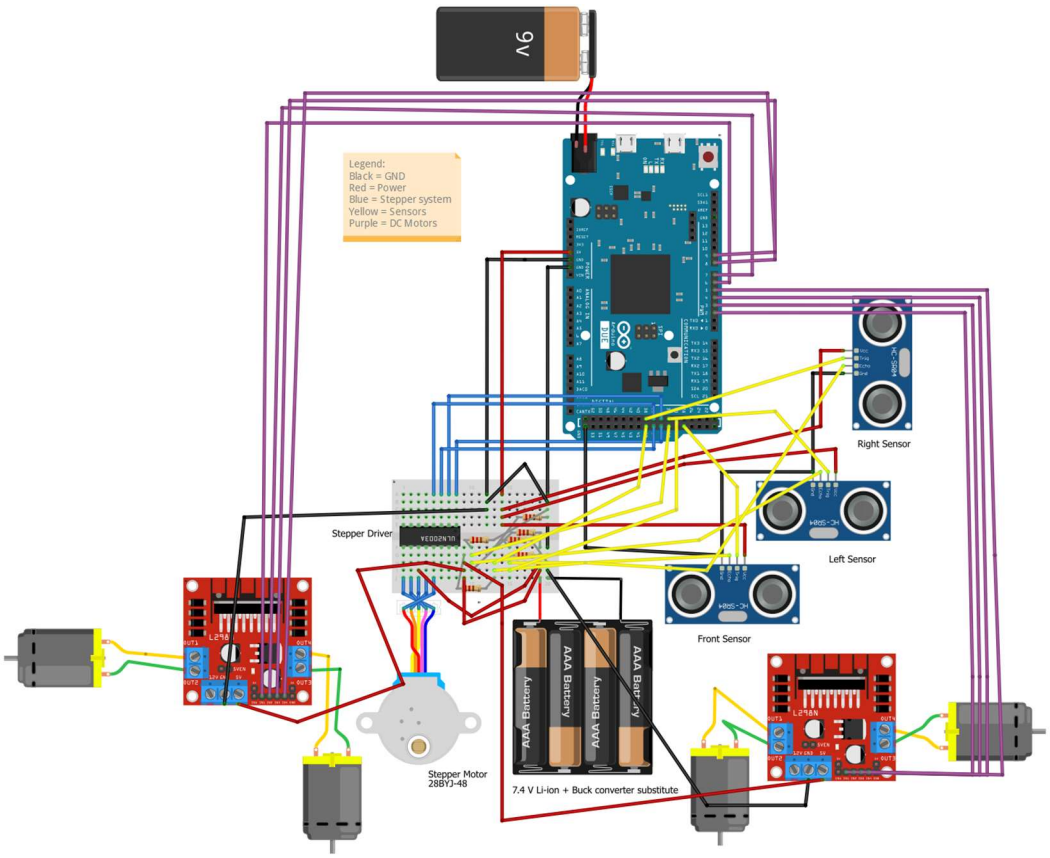


Figure 12. Phase A Circuit Schematic

7.0 Robot Functionality and Automation Algorithm

7.1 Operating Sequence

The Phase A mission prioritizes an autonomous, single sequence drive from the start to the end position on the maze as highlighted in figure 1. The Arduino Due begins the vehicles navigation as soon as its setup is complete in the starting position, requiring no operator involvement beyond pushing the start/reset button built into the microcontroller. Figure 13 shows a detailed flowchart of the vehicle's operating sequence.

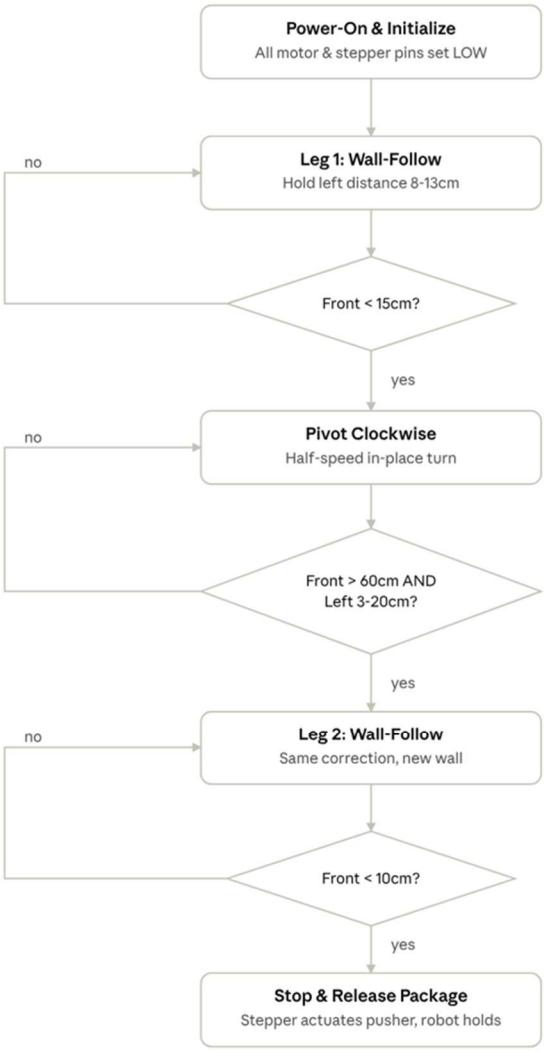


Figure 13. Detailed vehicle flowchart (DISCLAIMER: AI ASSISTED IMAGE USING TABLE 9)

7.2 Sensors, State Estimation, and Decision Logic

The vehicle's state estimation is dependent on two HC-SR04 ultrasonic sensors, connected on the front and the left, with a right sensor retained to ensure object orientation. The front sensor is primarily responsible for signaling the robot when to turn, once it reaches a low enough value the robot understands it’s getting closer to the left wall, and must now turn until it reaches a high enough distance that logically, the robot is facing the target location. The left sensor ensures the robot remains between 8-13 cm away from the left facing wall. If the robot senses less than 8 cm away from the left wall, it turns clockwise until it meets the specified range, and vice versa with the set upper limit. This ensures the robot retains wall tracking and path following, with right sensor data feeding in as a backup source for distance data.

Each sensor is pinged to sense on an independent timer every 35ms and incorporated into a 3-sample rolling average before the robot can utilize sensor data for logic decisions. A ping that returns 0 or provides a “No Echo” response is discarded rather than considered, ensuring the robot does not react to transient signals. Moreover, through testing, it was realized that utilizing raw ultrasonic sensing values was inconsistent with the robot's operation; the HC-SR04 sensors are very sensitive to noise and physical disturbances. Hence, the NewPing library was utilized to resolve these inconsistencies and validate them through bench testing.

Table 9. Logic Thresholds

Reading	Threshold	Action
Left distance < 8cm	Bot too close to wall	Arc right (reduce right PWM)
Left distance > 13cm	Bot too far from wall	Arc left (reduce left PWM)
Left distance 8-13cm	In limit	Drive straight
Front distance < 15cm	Corner reached	Stop, bot begin pivot
Pivot end condition	Front > 60 cm && left 3-20cms	Stop pivot, resume straight operation
Front distance < 10cm	Delivery location reached	Stop, activate stepper motor
Reading	Threshold	Action
Left distance < 8cm	Bot too close to wall	Arc right (reduce right PWM)
Left distance > 13cm	Bot too far from wall	Arc left (reduce left PWM)

7.3 Motion and Control Strategy

Due to the lack of encoders on the motors, the robot utilizes open loop steering control by sending direct PWM signals to the driver without receiving wheel rotation information back from the motors. Moreover, the robot utilizes a closed-loop control strategy for orientation and wall-correction by sending back ultrasonic sensor data to continuously align with the logic highlighted in Table 9.

7.3.1 Drive

All 4 wheels are commanded using analogWrite() PWM at a max drive speed of 255. Drifting towards/away from the left wall is corrected by reducing one sides PWM by 80, so the motion of the robot stays on track without requiring full stops to re aim its alignment, saving on the 150 seconds limit.

7.3.2 Turning

Initially, a 90-degree open loop turning logic was utilized for the robot to face the end position after reaching the first wall. However, since this integration did not utilize any sensor information, the actual turn was dependent on the terrain the bot was standing on along with the tuned turn numbers in the code. For a more robust design, this logic was switched to a pivot logic where the vehicle begin turning after hitting the first wall until it detects a distance greater than 60 cm on its front sensor. This value was tuned accordingly to bench testing on the terrain for Phase A.

7.3.3 Stopping Logic

All 4 motors are commanded to stop by sending a 0 PWM signal for each wall state transition, ensuring the robot arrives at a controlled stop rather than coasting off-track.

This strategy was sufficient for the scope of Phase A due to the path being constrained to a 120cmx80cm track, and the required path is a straightforward, two-legged route with no direct obstacles. Utilizing reactive sensor limits, rather than planned trajectory or PID controllers, was in alignment with this scope and mitigated further burden to the project timeline. The main risk of this strategy is an unbounded pivot if the sensor exit condition is never satisfied, which was already bench tested for in the specific Phase A track.

7.4 Pseudocode and Software Structure

The software structure operates in sequential logic in accordance with the track, where the current state is stopped, and the next state is given prioritization, with the sensor code running on loop and actively providing data back to the microcontroller. Figure 14 shows the software structure logic used. The file will be included in the submission.

```
loop():
    updateLeft()
    updateFront()
    updateRight()

    switch state:
        LEG1:
            left = effectiveLeft()   // leftAvg, or right-sensor-derived fallback
            wallFollowDrive(left)
            if frontAvg valid and frontAvg < FRONT_TURN_CM:
                stopAll(); state = TURN1

        TURN1:
            pivotCW()
            left = effectiveLeft()
            leftOK  = TURN_LEFT_MIN_CM <= left <= TURN_LEFT_MAX_CM
            frontOK = frontAvg valid and frontAvg > TURN_FRONT_MIN_CM
            if leftOK and frontOK:
                stopAll(); state = LEG2

        LEG2:
            left = effectiveLeft()
            wallFollowDrive(left)
            if frontAvg valid and frontAvg < FRONT_STOP_CM:
                stopAll(); state = DONE

        DONE:
            if not stepperDone:
                runStepperSequence()
                stepperDone = true

effectiveLeft():
    if leftAvg valid: return leftAvg
    if rightAvg valid:
        roomCross = ROOM_NS (legs 1) or ROOM_EW (leg 2)
        derived = roomCross - CAR_WIDTH - rightAvg
        if derived >= 0: return derived
    return invalid   // neither sensor usable -> drive straight
```

Figure 14. Software Structure logic (DISCLAIMER: AI ASSISTED USING TABLE 9)

8.0 Manufacturing Assembly Economy and Sustainability

8.1 Design for Manufacturing

The custom chassis, sensor housings, battery retainers, electronics covers, and pusher parts are intended for fused filament fabrication in PLA. Broad faces are oriented on the build plate to minimize support material, while openings and fastener access are kept on vertical or upward facing surfaces where possible. Load paths around motor mounts, wheel openings, and the pusher base require continuous fillets rather than thin unsupported tabs. FDM parts are directional-dependent because interlayer bonding is weaker, although the cited study tested ABS. Its anisotropy finding supports orienting highly loaded features, so their principal stress does not separate layer planes [16].

A practical phase A print standard is to keep overhangs near 45 degrees or less, provide local clearance at mating and sliding interfaces, and reserve threaded fasteners for components likely to be removed. The rack and pinion require the most careful fit: tooth engagement must be free enough to avoid step loss. The decisions can be seen in table 9.

Table 10. Design for Manufacture Decisions

Part / feature	Process / material	Manufacturing constraint	Design response	Inspection
Chassis and motor bays	FDM PLA	Large footprint; motor loads; long print	Flat build orientation, filleted bays, continuous walls, accessible fasteners	Overall dimensions, flatness, motor fit
Sensor housings / tall holder	FDM PLA	Small clips and thin walls can split across layers	Orient clips in-plane; use bolts for retention; avoid unsupported bridges	Sensor fit, bolt clearance, line of sight
Rack, pinion, and slide	FDM PLA	Tooth finish, backlash, and sliding friction	Print tooth faces cleanly; provide clearance; deburr without changing pitch	Full-stroke hand test, tooth engagement
Battery and electronics cover	FDM PLA	Repeated service and cable routing	Removable M2/M2.5 fasteners; lead-in chamfers; cable openings	Retention, no wire pinch, tool access

8.2 Design for Assembly and Disassembly

The robotic assembly is designed with a modular open concept idea. The four drive motors are installed in the chassis side bays before the wheel covers and wheels are fitted; the three ultrasonic modules are then secured in removable printed housings, followed by the battery retainers, Arduino Due, motor driver tower, and pusher. M2 and M2.5 bolts are used rather than permanent adhesive where service access is required. Removable top and battery covers allow the controller, drivers, and batteries to be reached without dismantling the wheel train, while the pusher can be removed as one functional unit.

9.0 Safety and Risk Analysis

Safety governs manufacturing, charging, bench work, autonomous motion, and package release. Printing is performed only on an intact printer with guarded hot surfaces and ventilation appropriate to the material. Electrical work begins with both batteries disconnected, and common ground connections are checked before applying power. The Li-ion pack is charged only with a compatible charger on a non-combustible surface and is removed from service if swollen, damaged or overheated. During powered testing, the robot is supported before wheel tests; hands are kept clear of the wheels and rack and pinion pinch point, and the course is kept clear of bystanders. A clearly accessible master power disconnect is required, so autonomous motion can be stopped without

reaching into moving components. 3D printing avoids many common manufacturing hazards, jumper wires were press fit wires, so no soldering was implemented for phase A.

9.1 Failure Modes and Effects Analysis (FMEA)

The FMEA uses 1-5 scales. Severity (S) ranges from negligible effect to injury, fire, or total mission loss; occurrence (O) ranges from remote to frequent; and detection (D) ranges from readily detected before the effect to unlikely to be detected. The risk priority number (RPN) is S x O x D. This can be found in table 10.

Table 11. Phase A FMEA

Subsystem	Failure mode and effect	Initial S/O/D = RPN	Design / process control	Revised S/O/D = RPN
Drive / control	Unexpected motion; collision or pinch	4/2/3 = 24	Power cable disconnect; guarded wheel test; start delay; clear test zone	4/1/1 = 4
Drive	Motor stall; hot driver, wiring, or battery	4/3/3 = 36	PWM limit; stall timeout; current/temperature test; wheel clearance	4/2/2 = 16
Power	Li-ion short, damage, or incorrect charging; fire	5/2/4 = 40	Compatible charger; fused/insulated leads; secure enclosure; inspection	5/1/2 = 10
Sensing	False or missing echo; boundary collision	3/3/3 = 27	Range validation; timeout; median/filter check; stop on invalid data	3/2/2 = 12
Release	Finger caught at rack/pinion or pusher	3/2/2 = 12	Guard/cover; hands-off operation; software travel limit; power-off service	3/1/1 = 3
Mechanical	Lose wheel or fastener; loss of steering	3/3/3 = 27	Pre-run torque check; retention feature; inspection marks	3/1/2 = 6
Wiring	Jumper disconnect/reversal; loss of control	3/3/2 = 18	Keyed/labeled connectors; strain relief; continuity check	3/2/1 = 6
Stability	Tip or payload shift on rough terrain	3/2/3 = 18	Low/central payload; slow turns; incline and disturbance tests	3/1/2 = 6
Printed parts	Layer crack at motor or sensor mount	3/2/3 = 18	Load-aligned print orientation; fillets; pre-run crack inspection	3/1/2 = 6

The highest initial risks are the Li-ion power system, motor stall/overcurrent, and sensing failure or a loose wheel. The controls therefore prioritize battery containment and charging discipline, a tested power isolation method, motor current limits, logic, and a pre-run inspection.

10.0 Engineering Documentation

The SolidWorks package included with the submission contains the final robot assembly along with all part files. The assembly combines the four-wheel chassis, sensing modules, two level electronics tower, battery retention, and rack and pinion pusher. Table 11 is a report level bill of materials; purchased parts are identified separately from custom printed parts.

Table 12. Phase A Bill of Materials

Item	Part / subassembly	Qty.	Make / buy	Material / model	Function / evidence
1	Chassis	1	Make	PLA	Primary structure; CAD part
2	Wheel D65x25	4	Buy / model	65 mm wheel	Mobility; CAD model
3	TT DC gearbox motor	4	Buy	3-6 V, 1:48 [13]	Four-wheel propulsion
4	Wheel cover / motor retainer	4	Make	PLA	Motor and wheel retention
5	Arduino Due	1	Buy / kit	SAM3X8E board [8]	Supervisory control
6	L298N motor-driver module	2	Buy	Dual H-bridge [14]	Drive switching
7	HC-SR04 ultrasonic module	3	Buy	5 V ranging module [6], [7]	Front/left/right sensing
8	Ultrasonic case / tall holder	3 sets	Make	PLA	Sensor orientation and protection
9	Li-ion battery pack	1	Buy	7.4 V, 3000 mAh	Actuator supply
10	9 V battery and holder	1	Buy + make	9 V + PLA holder	Controller/sensor supply
11	Battery covers / retainers	2 sets	Make	PLA	Battery retention and service access
12	28BYJ-48 stepper motor	1	Buy / available	5 V	Pusher actuation
13	ULN2003 stepper-driver module	1	Buy	Darlington array [15]	Stepper phase switching
14	Rack-and-pinion pusher assembly	1 set	Make	PLA	Payload release
15	Motor-driver tower and top case	1 set	Make	PLA	Electronics mounting/protection

Item	Part / subassembly	Qty.	Make / buy	Material / model	Function / evidence
16	M2 / M2.5 fasteners and wiring	A/R	Buy	Steel / copper	Assembly and electrical interconnect

Figure 14 shows the exploded view and identifies the four serviceable subassemblies: (A) mobility and chassis, (B) control and power tower, (C) ultrasonic sensing, and (D) package release. The mobility module establishes the wheel, motor, and chassis interfaces; the tower contains the controller, motor drivers, batteries, covers, and cable paths. Each sensing module consists of an HC-SR04 and a printed holder. The release module contains the stepper, ULN2003 driver, pinion, rack, slide, and pusher. These interfaces correspond to the functional architecture in Figure 12 and the item numbers in Table 11. The chassis is the dimension setting custom part because it controls motor location, sensor placement, battery fit, pusher alignment, and electronics mounting. Records of the approval of all bought components can be found in Appendix C.

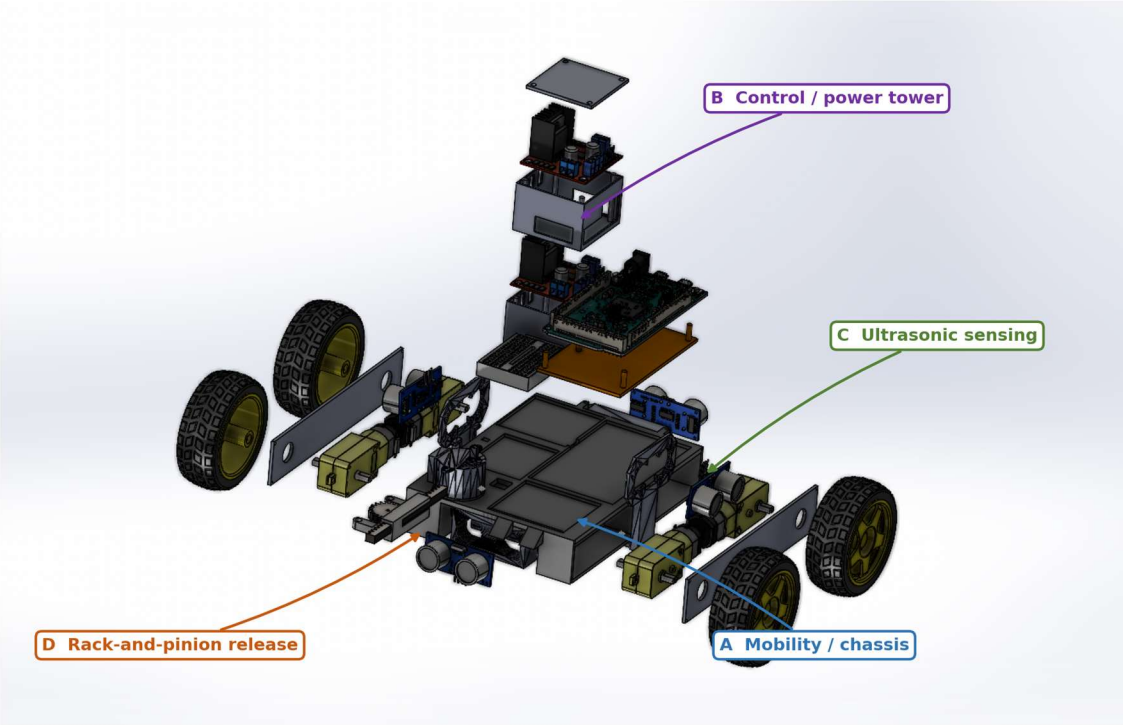


Figure 15. Exploded view and subassembly identification

Figure 15 shows the assembled Phase A design with the package storage location on the front left surface in front of the Arduino Due. The view shows the four-wheel stance, sensor coverage, pusher placement, Arduino Due location, and protected electronics tower. The package is positioned

inside the wheelbase to reduce tipping sensitivity while front and side ultrasonic modules keep clear fields of view. All engineering drawings can be found in Appendix B.

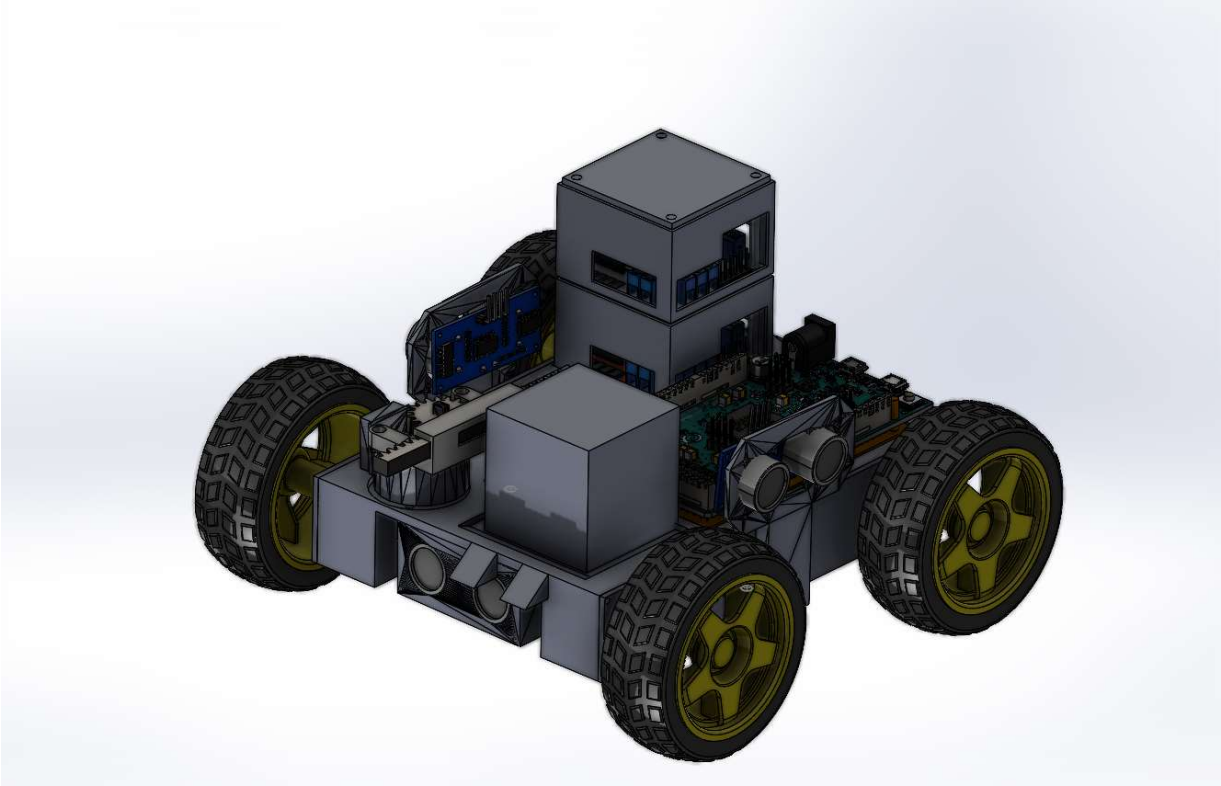


Figure 16. Rendered 3D View

11.0 Results Discussion and Product Validation

Though the vehicle was successful in mockup scenarios and bench testing, multiple uncertainties were introduced once both the stepper, and the driving operation were integrated together. Moreover, component fatigue, breadboard corruption, and complexity of the electrical circuitry introduced unprecedented variables that introduced further hurdles during product demonstration.

11.1 Component and Subsystem Results

Individual component tests were first conducted before integration into the full mechanical assembly to ensure operation validation. Each of the HC-SR04 sensors (front/left/right) were individually tested with the NewPing library to validate sensed data across 10cm intervals on a tape measure. Moreover, the speed, and polarity of the motors was tested with reference to the motor drivers to ensure straight drive during operation, mitigating occurrences of stray motor lag. Along

with that, the stepper motor was tested in isolation for consistent forward and reversed operation. The outcomes can be seen in table 13.

Table 13. Component Tests

Component	Test	Result
Each HC-SRO4	NewPing sensor sketch. Verified sensor data using a tape measure at 10 cm intervals to 120 cm.	Front and left sensor confirmed stable after switching from pulse logic to the NewPing library in the code. Right sensor also confirmed stable and used as a backup for orientation.
Motor Drivers	One wheel at a time, forward only pulse tests.	Confirmed forward polarity on all motors. Front left and front right were wired with reversed polarity which was mitigated in the code.
Stepper motor	Standalone full-step rotation tests, forward and reversed.	Confirmed function with rake and pinion setup highlighted in Figure 10.

The three main subsystems were tested in isolation first before combined integration. Wall following was first tested on both a mockup track and the Phase A track and tuned appropriately to ensure enough physical leverage remained for safe travel. The corner pivot upon reaching the left wall was tested and tuned for various exit conditions, eventually leading to a 60cm exit condition on the Phase A track after repeated testing. Lastly, the package release system was tested by connecting the stepper motor with the rake and pinion mechanism showcased in Figure 10 with the package in place to validate the torque and package release requirements. The outputs are highlighted in table 14.

Table 14. Subsystem tests

Subsystem	**Test**	**Result**
Wall following	Robot running in a straight line along the bottom wall of the track, left sensor distance logged over time	Confirmed wall tracking behavior against sensor output
Corner pivot	Robot driven and responded to corner trigger point, pivot exit condition observed	Using a 60cm pivot exit condition, confirmed that the robot reliably exits the pivot without indefinite spin, looking straight ahead at the exit position.
Package Release	Repeated release cycle using the rake and pinion	Confirmed stepper motor drive can facilitate package release

11.2 Full Phase A Mission Results

During testing, Phase A mission results were successful; the robot was successfully able to navigate the terrain and accurately make the turn to face the exit position at the top left of the track. However, once it came to test the wall tracking with the stepper, the complexity of the circuit had increased and the breadboard malfunctioned, leading to uncertainties with the subsystems working in tandem.

11.3 Phase A Demo Outcome

When it came time for the Phase A demonstration, the LiPo battery had its charge reduced to approximately 6V due to continued testing, causing the robot to be unable to turn on the pivot point, and the stepper motor to stall during package release due to the high torque/voltage requirement. Hence, upon a second demonstration, straight angled drive with only utilizing front sensor logic was used with an end condition to stop 20cm away from the front wall, as highlighted in Appendix D.

Table 15. Demo Criterion

Demo criterion	Official outcome/evidence	Technical explanation
Required terrain	Successful navigation through different types of terrain	The grip on the wheels and the high driver torque was successful in pathing the robot along different kinds of terrain. However, due to a lack of battery power during the demo, the turning was unsuccessful since the battery was unable to facilitate enough turning torque to the motors.
Package placement	Outside Boundary	Though the package was successfully released in testing, the lack of battery power during demonstration meant the stepper motor would start stalling when the robot met its end condition. If the 7.1 V LiPo battery was fully charged, package release and placement would be successful.
Time limit	Measured official time <150s	The official time was well within the 150 seconds limit.
Autonomy/save moves	0 interventions required for compliance	0 saves were utilized in the demonstration; the robot was fully autonomous.

11.4 Requirement Compliance Matrix

Most of the requirements highlighted in Table 2 were met as highlighted by Table 16. The custom 3d printed chassis facilitated all required hardware safely; the vehicle was able to drive across the different terrains highlighted in Figure 1, whilst also operating completely autonomously and stopping safely at the end zone. However, multiple requirements that were met during the testing phase failed during the official demonstration, particularly the dual-sensing wall pathfinding integration, and the package release. These failures were primarily due to unrecognized risk of lack of battery charge, high circuit complexity leading to electrical noises, and lack of pivot integration on different types of terrain.

Table 16. Compliance Matrix

Req. ID	Spec/test evidence	Result	Status	Gap or corrective action
F1	Full course trails from the specified starting position, robot must reach lobby zone on every recorded trial.	Robot successfully reached end zone on every test trial but was unsuccessful during the demo due to a lack of battery charge.	Unmet	Ensure battery charge is adequate before official demo and conduct/tune pathfinding for varying terrains
F2	System travel across each terrain on the path without stall, high centering or wheel slippage.	System successfully navigated across all terrain on the track without showing signs of reluctancy.	Met	Ensure battery charge available before official demo for correct pivot and wall following objectives, less reliance on straight path.
F3	Payload loaded before operation start on every test run, confirmed package still seated upon arrival at delivery.	Package successfully stayed afloat and balanced atop the robot chassis	Met	N/A
F4	Payload successfully released from robot upon practiced standalone testing.	Package unsuccessfully released from robot on official demonstration due to lack of battery charge causing stepper motor to stall.	Unmet	Ensure battery charge available before official demo

Req. ID	Spec/test evidence	Result	Status	Gap or corrective action
F5	Timed system tests using stopwatch and system reached endzone well before 150 seconds.	System successfully traversed through the terrain in the 150 second's budget.	Met	N/A
F6	No save moves or human intervention utilized in any of the test trial runs. System performed independently	System successfully performed autonomously during official demonstration without any save positions	Met	N/A
F7	System end states held all the drive motors at 0 by deenergizing the motor coils to ensure safety.	System successfully stopped safely at end condition without drifting.	Met	N/A
P1	Robot payload interface successfully accommodated the payloads physical attributes	The payload interface safely kept the payloads physical integrity during official demo.	Met	N/A
C1	Prefabricated chassis not utilized in any of the test phases	Custom 3D printed chassis proved successfully during official demo	Met	N/A
C2	All added materials approved before initiating tested phase.	All materials utilized in demo pre-approved as highlighted in Appendix C	Met	N/A

Req. ID	Spec/test evidence	Result	Status	Gap or corrective action
O1	The design of the robot encouraged robustness due to autonomous wall tracking by utilizing dual sensor data instead of single drive straight at an angle to the end zone. This approach proved successful during the test phase.	Battery charge proved too insufficient to facilitate robot turning during official demo, leading to forced straight drive to meet the end criterion.	Unmet	Ensure battery charge sufficient before future official demos and tune wall tracking integration along different terrains.

11.5 Discussion Limitations and Lessons Learned

Main lesson learned from Phase A is ensuring that battery capacity stays sufficient during each testing phase to avoid unprecedented conditions. Moreover, nearing demonstration day, it was realized the sensor position on the chassis made it so the ultrasonic sensors would read data that was interfered by the terrain, leading to incorrect data flows and robot logic. Along with that, as the complexity of the circuit increased, so did the electrical uncertainties, as halfway through testing it was realized the utilized breadboard was extremely fatigued, and the open resistors were interreacting with each other, leading to flawed/no sensor data flow to the DUE. Ensuring a cleaner, more robust circuit design, and more terrain versatile hardware and software structure is essential for future phases.

12.0 Phase B Development Plan

Phase A was a success in determining the strengths and weaknesses of the robot. While the exact engineering specifications for phase B have not yet been determined, the general task involves autonomously navigating down a ramp of approximately 17cm in width, entering a maze, navigating the maze to retrieve a package of known location, and returning the package to the top of the ramp. The robot will undergo significant upgrades and alterations to improve reliability and to accommodate the requirements of future project phases.

12.1 Required Design Changes

The major design changes included redesigning the chassis to hold the same autonomous system, as well as electrical component upgrades. The chassis length and width will be shrunken by increasing the height, this to fit future project phase requirements. The same motor style will be used; the gear ratio will be upgraded from a 1:48 plastic gear ratio to a 1:98 metal ratio which increases the torque and reduces the risk of stalling. The dual L298N motor drivers will be replaced with a single 4 channel motor driver; the L293D motor driver is of interest. The Arduino Due will be changed to support 5.0V input, this to support the ultrasonic sensor output without the need for a voltage divider. The chassis will have integral holding locations for the ultrasonic sensors, to ensure the sensors do not drift when mobile.

Phase B requires the package to be picked up and maneuvered autonomously; thus, the package pusher will no longer suffice. The conceptualized ideas for package maneuvering presented above will be revisited to view compliance with the new phase requirements.

12.2 Phase B Tasks and Delegates

The redesign of the car for future phases has begun as of this writing; the group will meet weekly to discuss updates. The following table describes the general task delegation between group members for the upgrades.

Table 17. Phase B General Task Delegation

Task	Delegate	Timeline
Chassis Redesign	Connor Armstrong	October 6 – Oct 20
Electrical Component Upgrade	Hani Hussain	October 10 – Oct 20
New Package Maneuver System	Ibraham Ahmed	October 6 – Oct 20
Model Construction and Validation	All members	Oct 20 – Oct 31

13.0 Conclusion

This report describes the design and development of an autonomously navigated robot which is capable of delivering a pre-loaded package. Multiple variants of terrain navigation and package manipulation are conceptualized and considered for development to meet the requirements of this project phase; a four-wheel drive vehicle with a rack and pinion pusher mechanism was deemed most suitable.

The surrounding environment of the robot was fed back with HC-SR04 ultrasonic sensors; this was the direct control aspect to move the 3-6 V TT DC motors navigating the terrain. The prototype was constructed with PLA components and displayed good results in preliminary testing; the sensor feedback was unreliable which was fixed by obtaining a median of a set of sample soundwave pulses. The primary code constructed was designed to follow along a wall of the phase A testing box, perform a 90-degree turn, follow the left wall until the end of the box (the package drop zone), and activate the rack and pinion to push the package off the chassis.

When the system was tested on demonstration day, the DC motor ended up stalling on the astroturf during the 90-degree turn. This was a result of an uncharged battery; the 7.4V Li-ion battery read a voltage of 6.0V after the first demonstration, which was dropped to 4.0-4.5V after the motor driver. Ultimately, the program was switched to display the all-terrain navigation system; the robot would cross all the terrains, stop when the front ultrasonic sensor detects a wall is 10cm away, and activate the stepper motor to drop the package. The robot was successfully able to navigate the terrain; however, the weak battery did not activate the stepper. A major lesson learned is to ensure the power source is always charged and ready to supply power.

References

[1] World Economic Forum, The Future of the Last-Mile Ecosystem, Jan. 2020. [Online]. Available: https://www3.weforum.org/docs/WEF_Future_of_the_last_mile_ecosystem.pdf. Accessed: Oct. 10, 2026.

[2] Ontario Tech University, "Phase A Demo Details," MECE 4100U demonstration brief, Sept. 13, 2026.

[3] G. T. Sibley, "Robot delivery system," U.S. Patent 10 459 450 B2, Oct. 29, 2019. [Online]. Available: https://patents.google.com/patent/US10459450B2/en

[4] S. Gerdts, J. Breckenridge, and K. Johnson, "Lunar Rover Optimization Platform for Wheel Traction Studies," in Earth and Space 2021, pp. 381-390, 2021, doi: 10.1061/9780784483374.038.

[5] J. Borenstein and L. Feng, "Measurement and correction of systematic odometry errors in mobile robots," IEEE Trans. Robot. Autom., vol. 12, no. 6, pp. 869-880, Dec. 1996, doi: 10.1109/70.544770.

[6] Adafruit Industries, "Ultrasonic Distance Sensor - 5V (HC-SR04)," Product 3942. [Online]. Available: https://www.adafruit.com/product/3942. Accessed: Oct. 10, 2026.

[7] Elec Freaks, "HC-SR04 Ultrasonic Ranging Module," datasheet. [Online]. Available: https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf. Accessed: Oct. 10, 2026.

[8] Arduino, "Arduino Due," hardware documentation. [Online]. Available: https://docs.arduino.cc/hardware/due. Accessed: Oct. 10, 2026.

[9] Ontario Tech University, "Mechatronics Design Project - Fall 2026," METE/MECE 4100U course project brief, Sept. 13, 2026.

[10] L. Llamas, "Measure distance with Arduino and GP2Y0E03 sensor," Luis Llamas, Mar. 9, 2018. [Online]. Available: https://www.luisllamas.es/en/measure-distance-arduino-gp2y0e03-sensor/. Accessed: Oct. 10, 2026.

[11] Sharp Corp., "GP2Y0E03 distance measuring sensor," product information. [Online]. Available: https://global.sharp/products/device/lineup/selection/opto/haca/diagram2.html. Accessed: Oct. 10, 2026.

[12] RobotShop, "6V 210:1 75RPM Micro Metal Gearmotor with Encoder," product page. [Online]. Available: https://ca.robotshop.com/products/6v-2101-75rpm-micro-metal-gearmotor-encoder. Accessed: Oct. 10, 2026.

[13] Adafruit Industries, "DC Gearbox Motor - TT Motor - 200RPM - 3 to 6VDC," Product 3777, datasheet. [Online]. Available: https://media.digikey.com/pdf/Data%20Sheets/Adafruit%20PDFs/3777_Web.pdf. Accessed: Oct. 10, 2026.

[14] STMicroelectronics, "L298 Dual Full-Bridge Driver," datasheet. [Online]. Available: https://www.st.com/resource/en/datasheet/l298.pdf. Accessed: Oct. 10, 2026.

[15] Texas Instruments, "ULN200x, ULQ200x High-Voltage, High-Current Darlington Transistor Arrays," datasheet, Rev. T, Mar. 2025. [Online]. Available: https://www.ti.com/lit/gpn/uln2003a. Accessed: Oct. 10, 2026.

[16] S.-H. Ahn, M. Montero, D. Odell, S. Roundy, and P. K. Wright, "Anisotropic material properties of fused deposition modeling ABS," Rapid Prototype. J., vol. 8, no. 4, pp. 248-257, 2002, doi: 10.1108/13552540210441166.

Appendices

APPENDIX A Gantt Chart

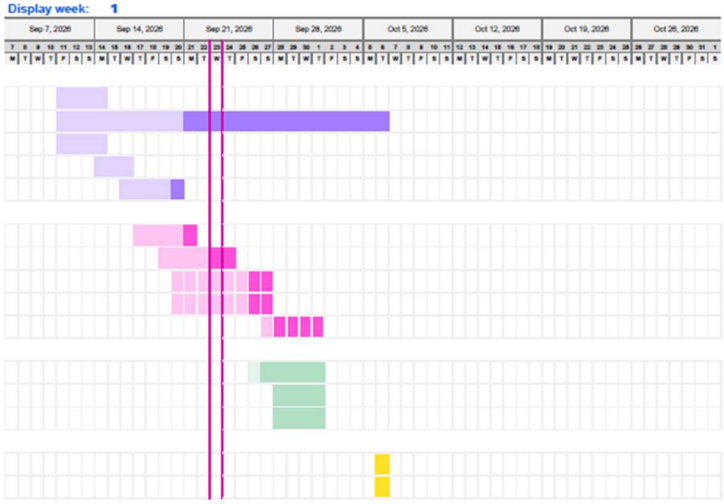
Mechatronics
Design Project

Phase A Gantt Chart

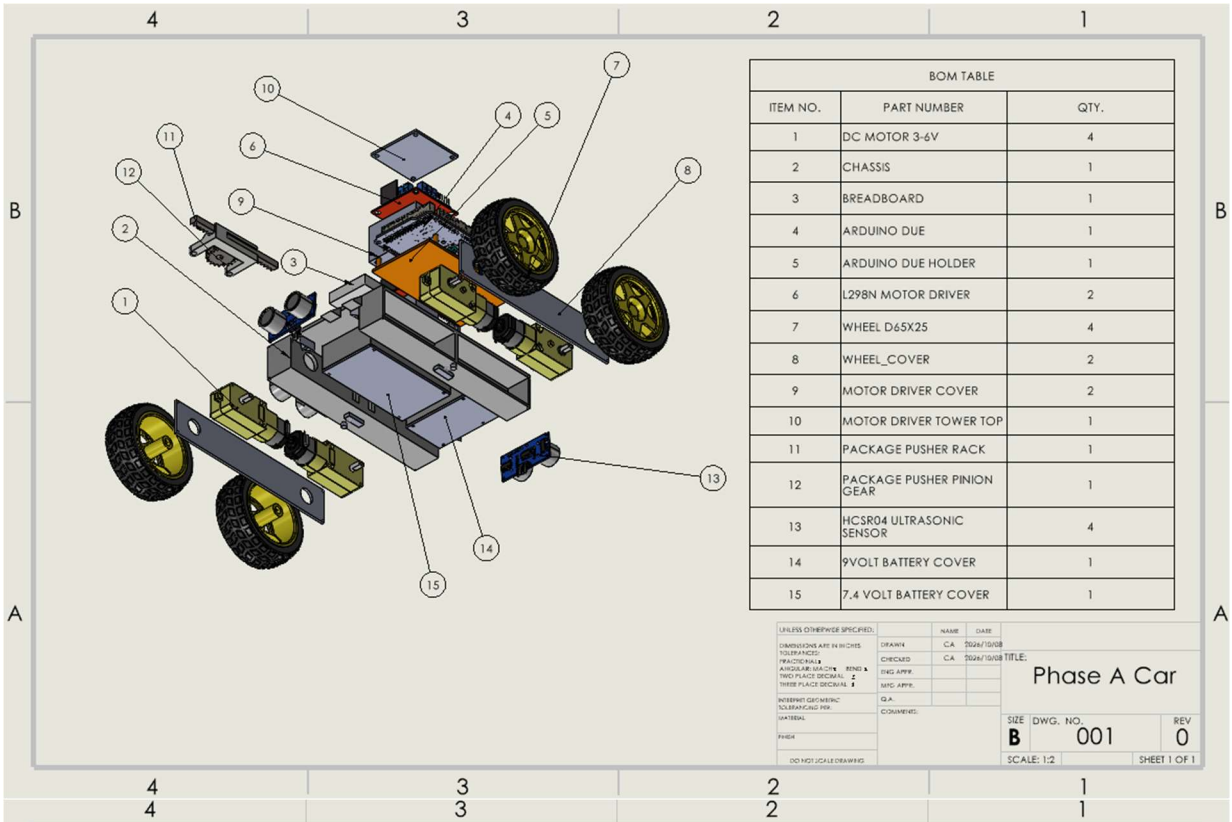
Project start: **Fri, 9-11-2026**

Display week: **1**

TASK	ASSIGNED TO	PROGRESS	START	END
Project Initiation				
Background Research	All Members	100%	9-11-26	9-14-26
Begin Engineering Report	All Members	40%	9-11-26	10-6-26
Initial Concept Generation	All Members	100%	9-11-26	9-14-26
Concept Selection	All Members	100%	9-14-26	9-16-26
Detailed Design of Concept	All Members	90%	9-16-26	9-20-26
Planning and design				
Circuit Design and Component Selection	Ibraham Ahmed and Hani Hussain	80%	9-17-26	9-21-26
Design of Chassis	Connor Armstrong	70%	9-19-26	9-24-26
Purchasing of Components	Ibraham Ahmed and Hani Hussain	80%	9-20-26	9-27-26
Construction of Chassis	Connor Armstrong	80%	9-20-26	9-27-26
Programming and Trouble Shooting	All Members	20%	9-27-26	10-1-26
Testing and Validation				
Sensor Calibration and Verification	Hani Hussain	20%	9-26-26	10-1-26
Trial Run to Verify Traversing Functionality	All Members	0%	9-28-26	10-1-26
Trial Run to Verify Package Release Functionality	All Members	0%	9-28-26	10-1-26
Evaluation				
Demonstration of Phase A Robotics Capabilities	All Members	0%	10-6-26	10-6-26
Completion of Phase A Report	All Members	0%	10-6-26	10-6-26



APPENDIX B Engineering Drawing Set



BOM TABLE		
ITEM NO.	PART NUMBER	QTY.
1	DC MOTOR 3-6V	4
2	CHASSIS	1
3	BREADBOARD	1
4	ARDUINO DUE	1
5	ARDUINO DUE HOLDER	1
6	L298N MOTOR DRIVER	2
7	WHEEL D65X25	4
8	WHEEL_COVER	2
9	MOTOR DRIVER COVER	2
10	MOTOR DRIVER TOWER TOP	1
11	PACKAGE PUSHER RACK	1
12	PACKAGE PUSHER PINION GEAR	1
13	HCSR04 ULTRASONIC SENSOR	4
14	9VOLT BATTERY COVER	1
15	7.4 VOLT BATTERY COVER	1

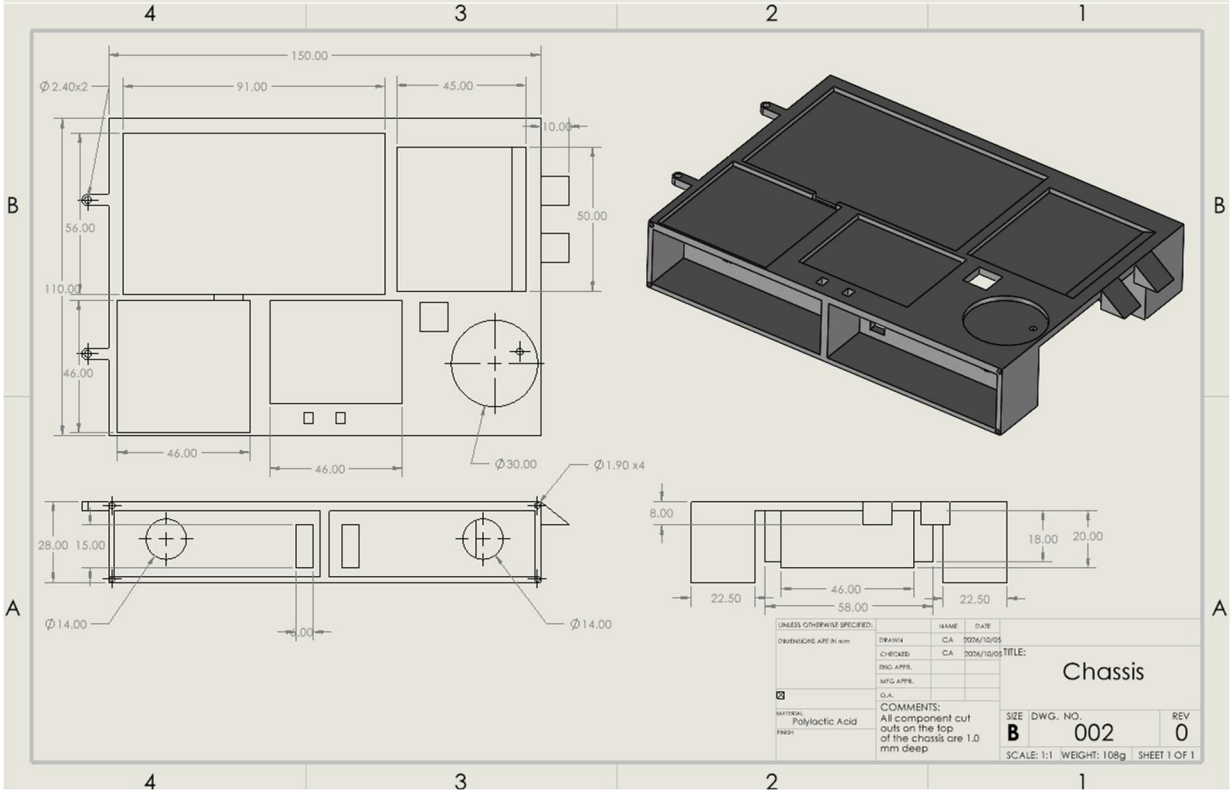
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN INCHES
TOLERANCES:
FRACTIONAL
ANGULAR: MACH
TWO PLACE DECIMAL
THREE PLACE DECIMAL
INTERPRET GEOMETRIC TOLERANCING PER:
MATERIAL
FINISH
DO NOT SCALE DRAWING

	NAME	DATE
DRAWN	CA	2026/10/08
CHECKED	CA	2026/10/08
ENG APPR.		
MFG APPR.		
Q.A.		
COMMENTS:		

TITLE: Phase A Car

SIZE	DWG. NO.	REV
B	001	0

SCALE: 1:2 SHEET 1 OF 1



UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN mm
MATERIAL Polylactic Acid
FINISH

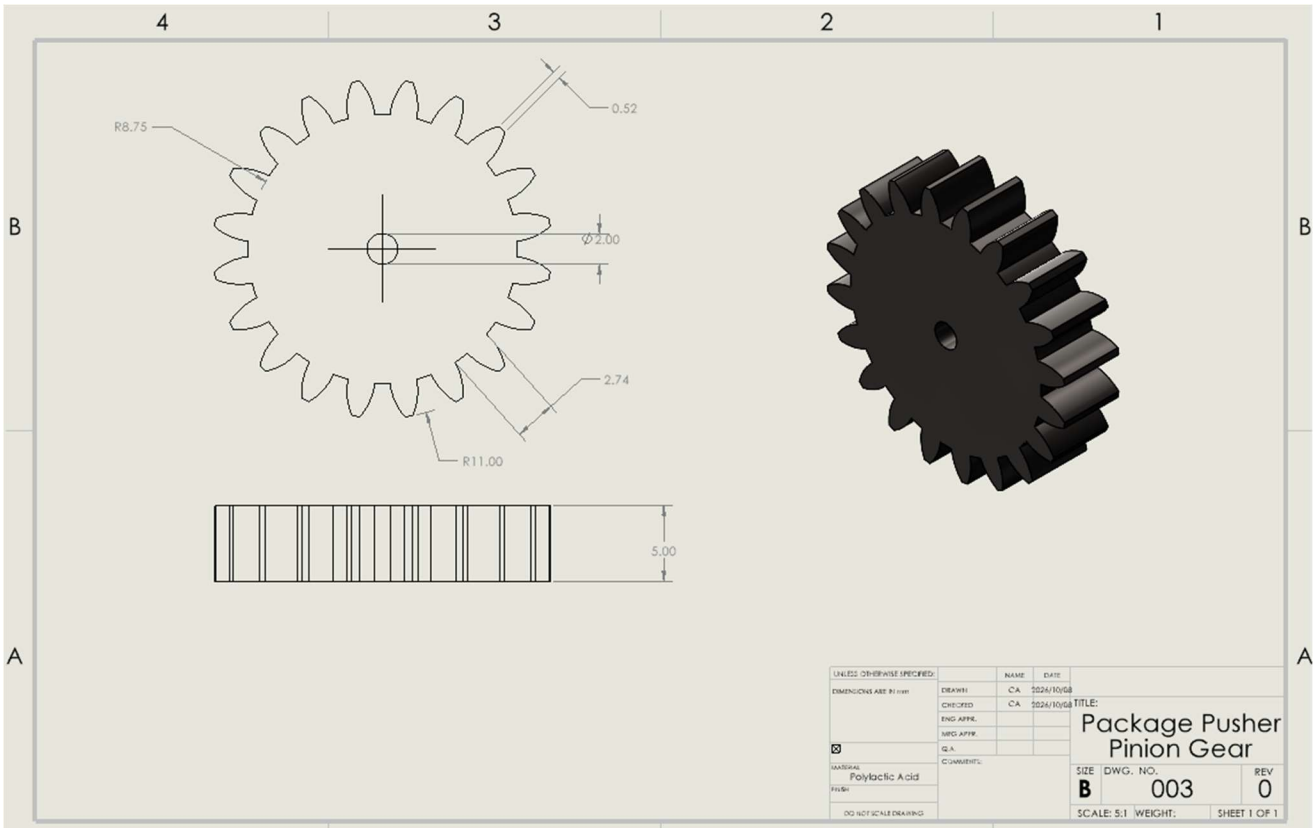
	NAME	DATE
DRAWN	CA	2026/10/05
CHECKED	CA	2026/10/05
ENG APPR.		
MFG APPR.		
Q.A.		

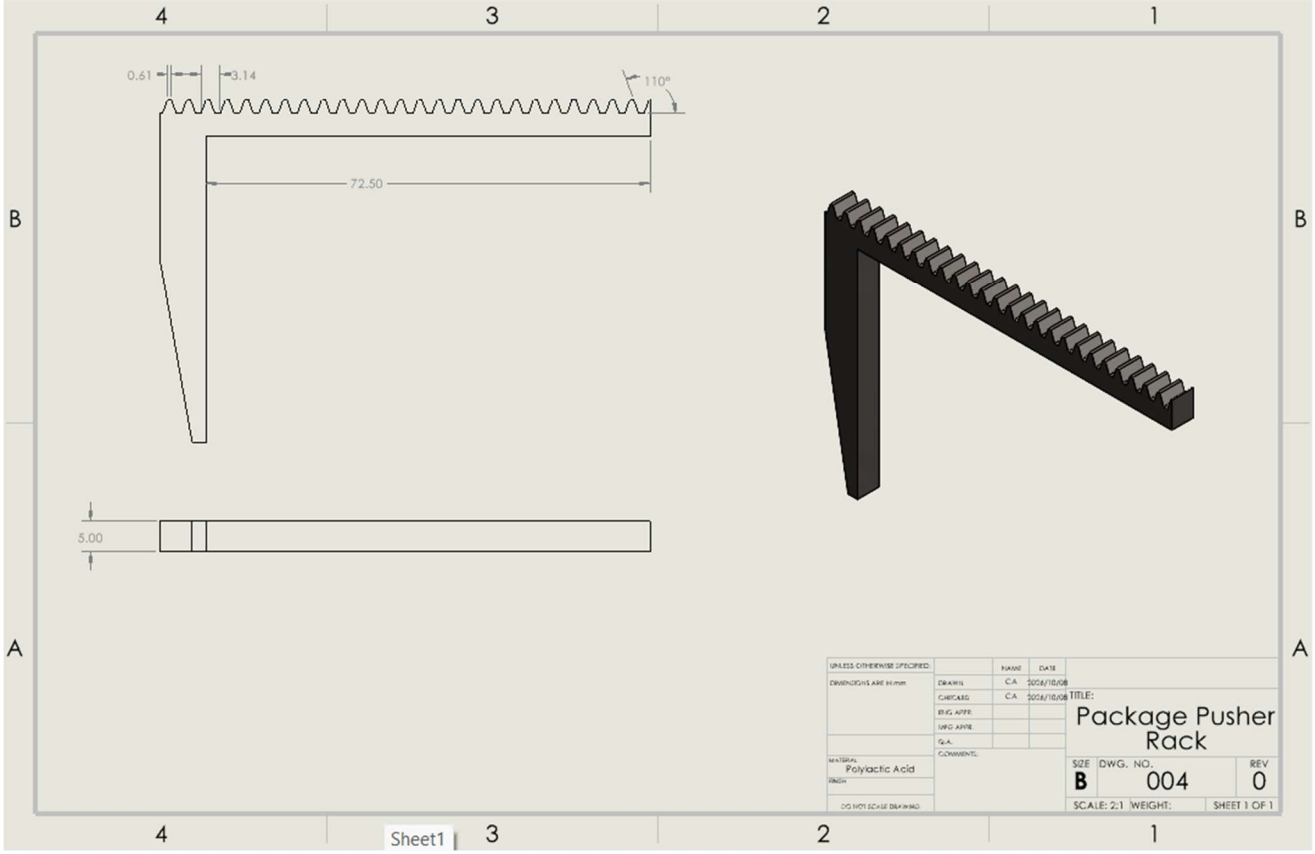
COMMENTS:
All component cut outs on the top of the chassis are 1.0 mm deep

TITLE: Chassis

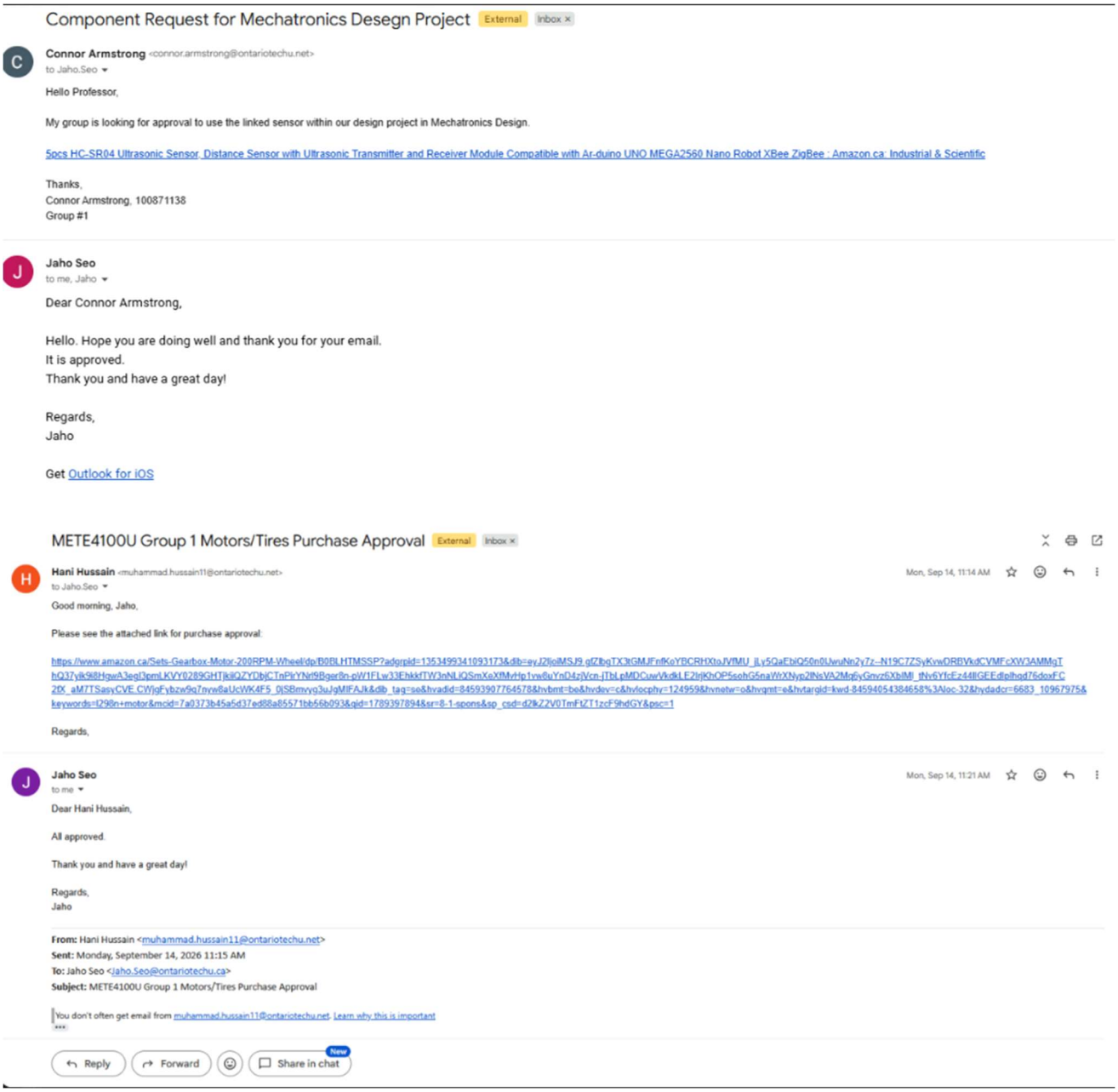
SIZE	DWG. NO.	REV
B	002	0

SCALE: 1:1 WEIGHT: 108g SHEET 1 OF 1


4
3
2
1
B
A
R8.75
0.52
Ø2.00
2.74
R11.00
5.00
Polylactic Acid
CA
2026/10/08
Package Pusher
Pinion Gear
SIZE
B
DWG. NO.
003
REV
0
SCALE: 5:1
WEIGHT:
SHEET 1 OF 1


4
3
2
1
B
A
0.61
3.14
110°
72.50
5.00
Polylactic Acid
CA
2026/10/08
Package Pusher
Rack
SIZE
B
DWG. NO.
004
REV
0
SCALE: 2:1
WEIGHT:
SHEET 1 OF 1
Sheet1

APPENDIX C Approval Records



Component Request for Mechatronics Desegn Project External Inbox ×

Connor Armstrong <connor.armstrong@ontariotechu.net>
to Jaho.Seo

Hello Professor,

My group is looking for approval to use the linked sensor within our design project in Mechatronics Design.

5pcs HC-SR04 Ultrasonic Sensor, Distance Sensor with Ultrasonic Transmitter and Receiver Module Compatible with Ar-duino UNO MEGA2560 Nano Robot XBee ZigBee : Amazon.ca: Industrial & Scientific

Thanks,
Connor Armstrong, 100871138
Group #1

Jaho Seo
to me, Jaho

Dear Connor Armstrong,

Hello. Hope you are doing well and thank you for your email.
It is approved.
Thank you and have a great day!

Regards,
Jaho

Get Outlook for iOS

METE4100U Group 1 Motors/Tires Purchase Approval External Inbox ×

Hani Hussain <muhammad.hussain11@ontariotechu.net>
to Jaho.Seo
Mon, Sep 14, 11:14 AM

Good morning, Jaho,

Please see the attached link for purchase approval:

https://www.amazon.ca/Sets-Gearbox-Motor-200RPM-Wheel/dp/B0BLHTMSSP?adgrpid=135349934109317 3&dib=eyJ2IjoiMSJ9.gfZbqTX3tGMJFnfKoYBCRHXtoJVfMU_jLy5QaEbiQ50n0UwuNn2y7z--N19C7ZSyKvwDRBVkdCVMFcXW3AMMgThQJ7ylk9k8HgwA3egI3pmLKVY0289GHTjkiiQZYDbjCTnPkYNrl9Bger8n-pW1FLw33EhkkfTW3nNLiQSmXeXfMvHp1vw8uYnD4zjVcn-jTbLpMDCuwVkdkLE2brjKhOP5sohG5naWrXNyp2lNsVA2MqfjyGnvz6XbIMI_tNv6YfcEz44lIGEEdlplhqd75doxFC2fX_aM7TSasyCVE.CWjgFybzw9q7nyw8aUcWK4F5_0jSBmvyq3uJgMIFAJk&dib_tag=se&hvadid=84593907764578&hvbmt=be&hvdev=c&hvlocphy=124959&hvnetw=o&hvqmt=e&hvtargid=kwd-845940543846 58%3Aloc-32&hydadcr=6683_10967975&keywords=t298n+motor&mcid=7a0373b45a5d37ed88a85571bb56b093&qid=1789397894&sr=8-1-spons&sp_csd=d2lkZ2V0TmFtZT1zcF9hdGY&psc=1

Regards,

Jaho Seo
to me
Mon, Sep 14, 11:21 AM

Dear Hani Hussain,

All approved.

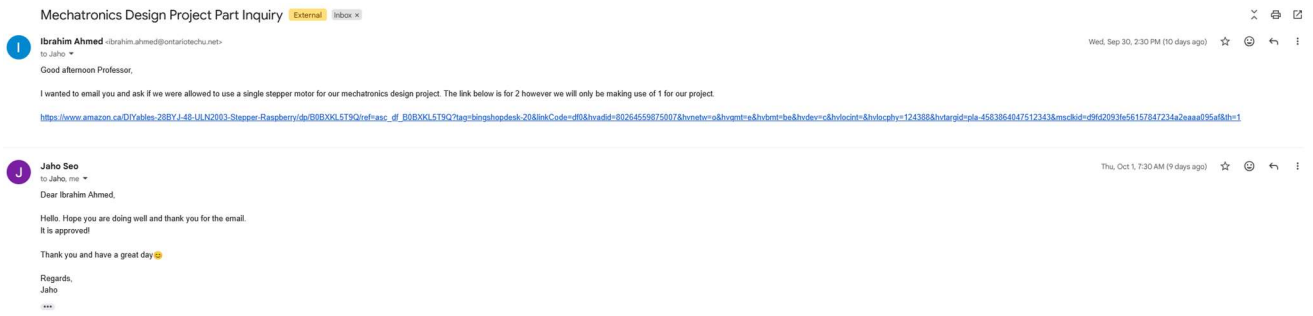
Thank you and have a great day!

Regards,
Jaho

From: Hani Hussain <muhammad.hussain11@ontariotechu.net>
Sent: Monday, September 14, 2026 11:15 AM
To: Jaho Seo <Jaho.Seo@ontariotechu.ca>
Subject: METE4100U Group 1 Motors/Tires Purchase Approval

You don't often get email from muhammad.hussain11@ontariotechu.net. Learn why this is important

Reply Forward Share in chat New



Mechatronics Design Project Part Inquiry External Inbox ×

Ibrahim Ahmed <ibrahim.ahmed@ontariotechu.net>
to Jaho
Wed, Sep 30, 2:30 PM (10 days ago)

Good afternoon Professor,

I wanted to email you and ask if we were allowed to use a single stepper motor for our mechatronics design project. The link below is for 2 however we will only be making use of 1 for our project.

https://www.amazon.ca/DIYables-28BYJ-48-ULN2003-Stepper-Raspberry/dp/B0BXKL5T9Q/ref=asc_df_B0BXKL5T9Q?tag=bingshopdesk-20&linkCode=df0&hvadid=80264559875007&hvnetw=o&hvqmt=e&hvbmt=be&hvdev=c&hvlocint=&hvlocphy=124388&hvtargid=pla-458386404751234 3&mscklid=d9fd2093fe5615784723 4a2eaaa095af&th=1

Jaho Seo
to Jaho, me
Thu, Oct 1, 7:30 AM (9 days ago)

Dear Ibrahim Ahmed,

Hello. Hope you are doing well and thank you for the email.
It is approved!

Thank you and have a great day😊

Regards,
Jaho

APPENDIX D Front Sensor Code Block (AI ASSISTED)

```
#include <NewPing.h>

//DISCLAIMER AI ASSISTED CODE
// ---------- Front sensor ----------
#define FRONT_TRIG 30
#define FRONT_ECHO 29
#define MAX_DISTANCE 200
NewPing sonarFront(FRONT_TRIG, FRONT_ECHO, MAX_DISTANCE);

// ---------- Motor pins ----------
#define FL_FWD 2
#define FL_REV 3
#define RL_FWD 5
#define RL_REV 4
#define FR_FWD 7
#define FR_REV 6
#define RR_FWD 8
#define RR_REV 9

// ---------- Stepper pins ----------
const int STEPPER_IN1 = 35;
const int STEPPER_IN2 = 34;
const int STEPPER_IN3 = 37;
const int STEPPER_IN4 = 36;

const int STEPS_PER_REV_STEPPER = 2048; // 360 deg
const int STEP_DELAY_MS = 3;            // raise if the stepper stalls

const int fullStepSeq[4][4] = {
  {1,0,0,0},
  {0,1,0,0},
  {0,0,1,0},
  {0,0,0,1}
};

// ---------- Tunables ----------
const int   DRIVE_SPEED  = 255;
const float STOP_CM      = 20.0;
const unsigned long PING_INTERVAL = 35;
const int   ROLL_N       = 3;

// ---------- Rolling average ----------
float frontBuf[ROLL_N] = {0};
int frontCount = 0, frontIdx = 0;
unsigned long lastFrontPing = 0;
float frontAvg = -1;   // -1 = no valid reading yet
```

```
bool stopped = false;
bool stepperDone = false;

// ---------- Motors ----------
void setMotor(int pinFwd, int pinRev, int speed) {
  speed = constrain(speed, -255, 255);
  if (speed >= 0) { analogWrite(pinFwd, speed); digitalWrite(pinRev, LOW); }
  else            { analogWrite(pinRev, -speed); digitalWrite(pinFwd, LOW); }
}

void driveAll(int leftSpeed, int rightSpeed) {
  setMotor(FL_FWD, FL_REV, leftSpeed);
  setMotor(RL_FWD, RL_REV, leftSpeed);
  setMotor(FR_FWD, FR_REV, rightSpeed);
  setMotor(RR_FWD, RR_REV, rightSpeed);
}

void stopAll() { driveAll(0, 0); }

// ---------- Stepper ----------
void rotateStepper(int steps) {
  int dir = (steps >= 0) ? 1 : -1;
  int totalSteps = abs(steps);
  static int seqIndex = 0;
  for (int i = 0; i < totalSteps; i++) {
    seqIndex = (seqIndex + dir + 4) % 4;
    digitalWrite(STEPPER_IN1, fullStepSeq[seqIndex][0]);
    digitalWrite(STEPPER_IN2, fullStepSeq[seqIndex][1]);
    digitalWrite(STEPPER_IN3, fullStepSeq[seqIndex][2]);
    digitalWrite(STEPPER_IN4, fullStepSeq[seqIndex][3]);
    delay(STEP_DELAY_MS);
  }
}

void runStepperSequence() {
  rotateStepper(-STEPS_PER_REV_STEPPER);
  delay(500);
  rotateStepper(STEPS_PER_REV_STEPPER);
  digitalWrite(STEPPER_IN1, LOW);
  digitalWrite(STEPPER_IN2, LOW);
  digitalWrite(STEPPER_IN3, LOW);
  digitalWrite(STEPPER_IN4, LOW);
}
```

```
  lastFrontPing = millis();

  unsigned int cm = sonarFront.ping_cm();
  if (cm == 0) return; // ignore no-echo

  frontBuf[frontIdx] = cm;
  frontIdx = (frontIdx + 1) % ROLL_N;
  if (frontCount < ROLL_N) frontCount++;

  float sum = 0;
  for (int i = 0; i < frontCount; i++) sum += frontBuf[i];
  frontAvg = sum / frontCount;

  Serial.print("Front: ");
  Serial.print(frontAvg);
  Serial.println(" cm");
}

void setup() {
  for (int p = 2; p <= 9; p++) { pinMode(p, OUTPUT); digitalWrite(p, LOW); }
  pinMode(STEPPER_IN1, OUTPUT);
  pinMode(STEPPER_IN2, OUTPUT);
  pinMode(STEPPER_IN3, OUTPUT);
  pinMode(STEPPER_IN4, OUTPUT);

  Serial.begin(115200);
  delay(500);
  Serial.println("Driving forward...");
}

void loop() {
  if (!stopped) {
    updateFront();
    driveAll(DRIVE_SPEED, DRIVE_SPEED);

    if (frontAvg >= 0 && frontAvg <= STOP_CM) {
      stopAll();
      stopped = true;
      Serial.println("Reached 20 cm -> stopping");
    }
  } else {
    stopAll();
    if (!stepperDone) {
      Serial.println("Running stepper sequence");
      runStepperSequence();
      stepperDone = true;
      Serial.println("Stepper sequence complete");
```