Smart RC Rescue Rover

A Senior Project Report

Presented to

Department of Electronics and Computer Engineering Technology

Indiana State University

Terre Haute, Indiana

In Partial Fulfillment

of the Requirements for the Degree

B.S. Computer Engineering Technology

by

Lea Tice, Edgar Fernandez, & Gavin Schumaker

May 2026

Keywords: Autonomous, Rover, Search & Rescue, Unmanned Ground Vehicle, Thermal Imaging

ABSTRACT

This project addresses the safety risks faced by first responders when entering structurally unstable or hazardous environments. The primary objective is to design, engineer, and prototype a "Smart RC Rescue Rover", a semi-autonomous, remote vehicle capable of acting as a "first-in" scout to provide real-time situational awareness, thereby reducing human exposure to immediate threats. This project contributes to the field of applied robotics by demonstrating a cost-effective rover that integrates thermal sensing logic, low-latency video transmission, and active payload deployment within a customizable platform for specific first responder duties and tasks. The methodology employs an iterative engineering design process divided into parallel hardware and software development tracks. This means we aren't just building the whole rover in one go and hoping it works at the end but completing the different sub-systems and integrating them into one full system. We recently completed the propulsion system which addresses the challenge of static inertia using high-torque, 1:90 ratio metal-gear DC motors driven by a differential steering algorithm (tank drive). Expected results include a fully functional prototype capable of navigating uneven terrain, transmitting real-time visual and thermal feedback to a handheld operator unit, and successfully actuating a payload depending on the task (e.g. fire suppression, search and rescue, etc.). Performance validation will involve field testing the rover’s torque efficiency on inclines, communication latency in non-line-of-sight conditions, and autonomous obstacle avoidance reliability. These findings will evaluate the feasibility of deploying low-cost, high-torque robotic platforms to enhance operational safety in dynamic rescue scenarios.

PREFACE

This report documents the ongoing development of a smart RC rescue rover, a semi-autonomous unmanned vehicle designed to assist first responders in hazardous environments. This project will be conducted over the course of the Fall 2025 and Spring 2026 semester. The goal of this project is to integrate theoretical knowledge of embedded systems, software engineering and robotics into a useful in the real-world functional platform.

The completion of this project relies on an iterative engineering design model which allows us all to work simultaneously on different tasks at different times. This required a multidisciplinary approach that forced us to rely on each individual members' talents.

Division of Responsibility:

- **Lea Tice** focuses on circuit design, firmware programming, and sensor integration.
- **Edgar Fernandez** focuses on circuit installation.
- **Gavin Schumaker** focuses on mechanical chassis design and 3D printing.

ACKNOWLEDGMENTS

This project is being completed with the support of the Department of Electronics and Computer Engineering at Indiana State University. I would like to sincerely thank Dr. Javaid for her invaluable guidance and robotics expertise throughout this project. Her feedback and knowledge on the subject were driving forces in what shaped this project. I would also like to thank Dr. McLeod and Dr. Li for their mentorship and classes on C programming and microcontrollers, which are essentially the core of this project.

TABLE OF CONTENTS

0

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

BACKGROUND

The basis of this project is to provide a new alternative to help first responders combat potentially risky or hazardous situations. Most of the time, first responders are unable to enter these environments because of the unstable or risky conditions that the problem might be in such as intense fires or damaged structures. The solution that we came up with is a smart RC rescue rover that could help with these intense scenarios. The rover will be able to extinguish fires at a safe distance and be able to traverse in different terrains to complete its task. Some of the other features for this rover will include stuff like autonomous driving, thermal detection, and live camera feed.

CHAPTER 2

INTRODUCTION

The members of this Rover project are Lea Tice, Edgar Fernandez, and Gavin Schumaker. Lea Tice, the Project Manager and Lead Embedded Systems Engineer, is responsible for keeping the group and project on track; she is also responsible for engineering the embedded systems of the Rover. Steering, and other systems like it. Edgar is responsible for installing the components of the Rover. Gavin is responsible for modeling and printing any necessary parts for the Rover. This Rover's purpose is currently a prototype for an autonomous drone that's able to drive around hazardous environments; eventually this Rover should be able to be used for emergency situations.

Table 1

Student Project Team Composition.

Team Member	Roles
Lea Tice	Project Manager & Lead Embedded Systems Engineer. Responsible for firmware development and circuit design.
Edgar Fernandez	Hardware Integration. Focused on component installation.

Gavin Schumaker	Mechanical Design Lead. Responsible for CAD modeling and 3D printing custom chassis and sensor housing.

Deliverables

To validate the success of the Smart RC Rescue Rover project, we have defined a comprehensive set of physical and digital deliverables. These items encompass the functional electromechanical hardware and the technical documentation necessary for operation and future replication. The project deliverables are categorized into three primary components.

Smart Rescue RC Rover & Controller

This deliverable consists of the fully assembled functional rover. The custom-designed vehicle will feature a 3D printed chassis, two high-torque DC motors, and integrated sensor housing. The controller will be a handheld remote unit that will feature two analog joysticks for speed and direction control, a button to release the payload, a thermal feedback visual cue, and the live feed.

Technical Documentation

This will be a comprehensive set of engineering documents that detail the full design process of the rover. This will include all the source code, electronic design schematics, CAD files, bill of materials and any included reports of significance.

User Manual

The user manual will be designed for the end-user (first responders) to operate the rover safely and effectively. The manual should include two main sections, which are the operations and maintenance procedures. The operations section will include step-by-step instructions on the basic operation of the rover. The maintenance section will include tier 1 instructions on maintenance and troubleshooting as well as next steps if issues are beyond tier 1 and repair must be completed by tier 2 personnel (project subject matter experts).

Features & Functions

The Smart RC Rescue Rover features distinct capabilities designed to enhance operational safety and efficiency in the field.

Driving Capabilities (Wheeled)

The rover's propulsion system is engineered to navigate unpredictable and extreme terrains. It utilizes two gearbox motors with a 1:90 gear ratio. This high-torque gearing allows the rover to overcome static inertia without stalling. This decision was made because the rover's main concern is not speed but maneuverability. The wheels feature a bead-lock design for secure tire seating, and the tires are 60 mm rubber tires. The decision to choose rubber ensures maximum traction across a variety of surfaces, including concrete, tile, and uneven debris.

Autonomous Driving

The rover is primarily operated by the controller however the user will have an option to allow for autonomous driving. This feature allows the operator to switch the rover to "scout" mode, freeing up personnel to focus on other critical tasks during emergency situations. The rover utilizes ultrasonic sensors to detect and avoid obstacles within a calibrated range. When an obstacle is detected, the collision avoidance logic halts motor operation and executes a redirection maneuver to prevent impact.

Thermal Sensing

A critical safety feature will include integrated thermal sensing logic. This system will utilize a thermocouple instead of a thermistor. This thermocouple should detect temperature differentials that are invisible to the naked eye, particularly in environments with low visibility due to smoke density. The onboard logic will be able to identify different heat signatures such as a survivor's natural body heat or a hazardous heat source that could be invisible from thick smoke. This capability is essential for identifying "flashover" conditions, such as intense heat radiating from behind a closed door, alerting first responders to potential danger before they breach a room.

Payload

The rover will be equipped with a modular payload adaptable to the rover's specific area of operations such as firefighting, search & rescue, police & military operations, etc. In this project, we are primarily focusing on firefighting support where our payload will feature a water discharge system for small-scale fire suppression. This function will be able to be actively triggered remotely via a dedicated input on the controller unit.

Live-Feed Camera

To provide real-time situational awareness to the rover's operator, the rover will transmit a low-latency video feed to the controller. The first-person view will allow the operator to see exactly where the rover is and its surroundings from a safe distance without having to maintain line of sight with the rover.

Acceptance Criteria

Acceptance criteria are based on a pass/fail condition. These systems must be no-fail as they are the main functions of the rover. Without these main functions, the rover will not operate as intended with the mission in mind.

Driving Capabilities

The rover will utilize a tank-drive algorithm that will give the rover more control and maneuverability in extreme terrains. The rover must successfully go forward, reverse, and turn left & right, depending on joystick input. The propulsion system must generate enough torque to move across and climb surfaces with its intended weight and payload.

Real-time Camera Feed

Video transmission must provide continuous visual feed to the controller with a low enough latency to allow for operations. The target latency should be less than 500 milliseconds.

Payload Deployment

The payload actuation mechanism must be able to be triggered remotely from the handheld controller. The system must discharge the intended payload (water, fire suppressant, etc.). The payload system must not leak or misfire during operation.

Autonomous Obstacle Detection & Navigation

When placed in its ‘autonomous mode’, the rover must be able to successfully detect an obstacle and go around it. It should make use of its thermal sensor as well to detect heat signals above a certain threshold to simulate a human body or fire.

Controller Communication with Rover

The handheld controller must maintain a stable connection with the rover. Line of sight should not be a factor in reliable communications as the rover is intended to navigate hard to reach and hazardous areas that pose too much of a risk to human life.

CHAPTER 3

PROJECT MANAGEMENT

Effective project management is essential to the success of complex engineering tasks, particularly when integrating mechanical, electrical, and software sub-systems. This chapter outlines the organizational strategies, financial planning, and scheduling methodologies employed by the team to ensure the Smart RC Rescue Rover is completed within the academic year. The project operates under strict time constraints, necessitating a structured approach to resource allocation, risk mitigation, and milestone tracking.

Methodology

To ensure the timely completion of the Smart RC Rescue Rover, the team has adopted an iterative engineering design process [5]. This methodology facilitates a highly efficient, parallel workflow, allowing hardware construction and software development to occur simultaneously rather than sequentially. This approach minimizes "downtime" where one member is waiting for another to finish a task.

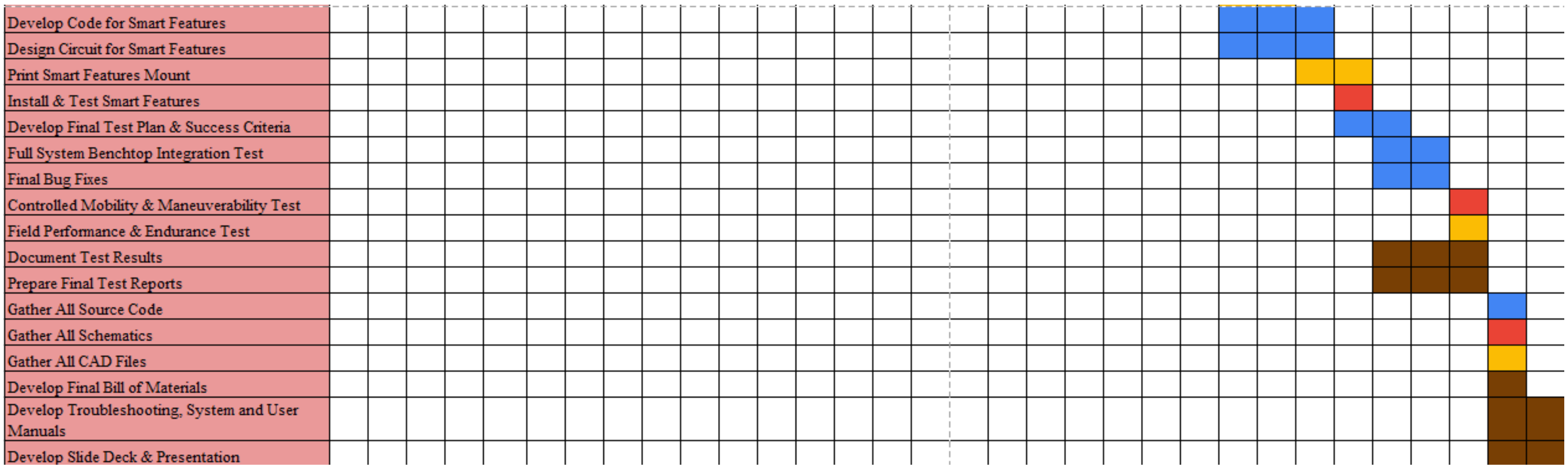
As the Lead Embedded Systems Engineer, Lea Tice focuses on the initial design and firmware coding for individual sub-systems (e.g., the propulsion logic or thermal sensing logic). Once a sub-system's circuit design is validated, Edgar Fernandez immediately proceeds with component installation and physical wiring. Simultaneously, Gavin Schumaker manages the CAD modeling and manufacturing (3D printing) of custom chassis components. This ensures that critical elements, such as sensor mounts and the controller housing, are fabricated "just-in-time" for final assembly. By testing and refining each sub-system independently before full-scale integration, the team can identify and resolve technical issues early in the prototyping phase.

Development Tools & Resources

To facilitate seamless development and design collaboration, the team utilizes the Microsoft 365 ecosystem. Microsoft Teams serves as the primary hub for daily communication, while the suite's cloud integration allows for real-time collaborative editing of project documentation. For software version control, GitHub is used to manage the C firmware repository. This enables the safe implementation of new algorithms through branching, ensuring that updates can be tested without disrupting existing logic. Autodesk Fusion 360 was used in modeling the main body and payload of the rover. The design engineer has opted to use another software to model the controller. Arduino IDE serves as the primary development environment, chosen for its native compatibility with the project's microcontroller architecture.

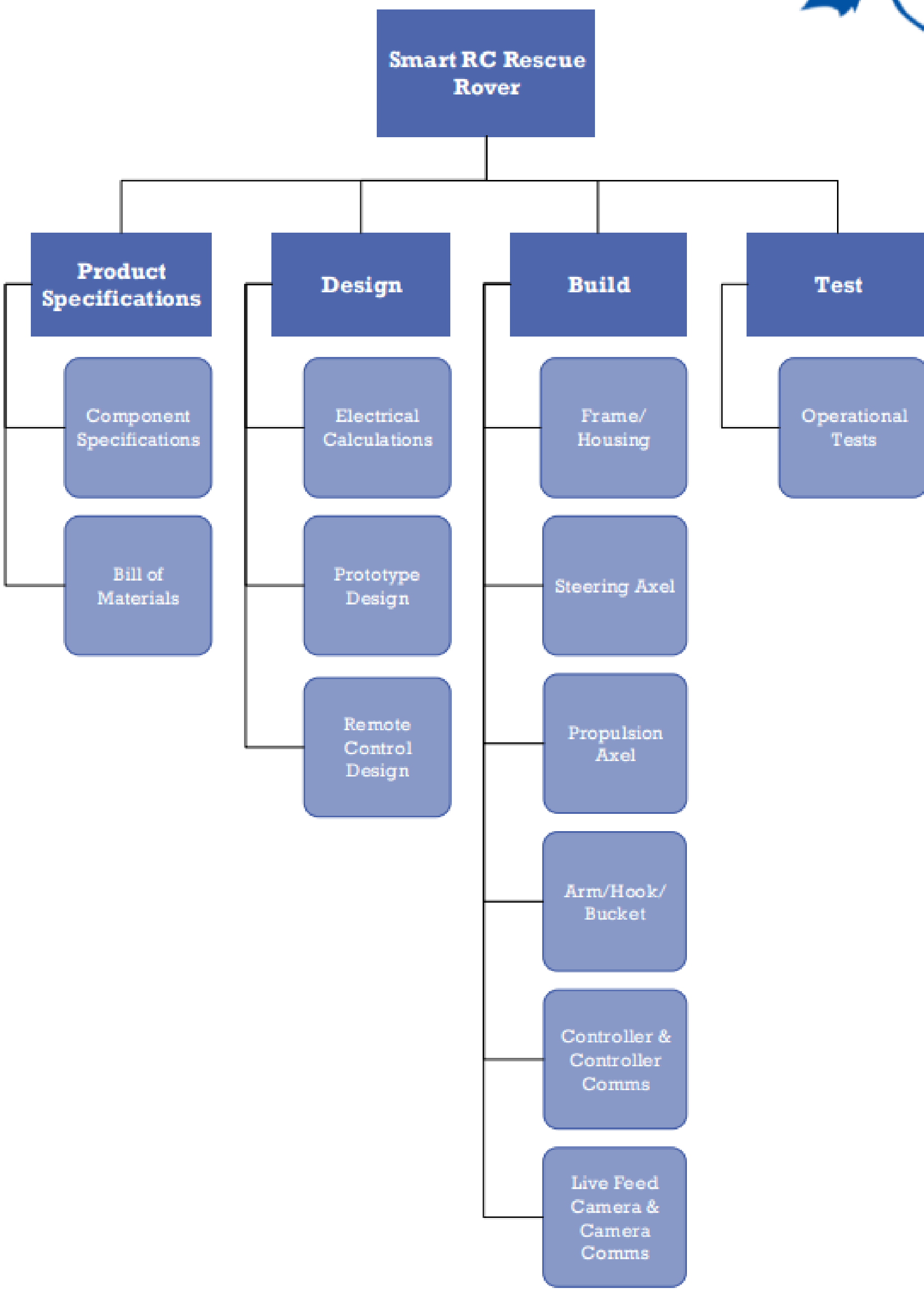
Gantt Chart


Smart RC Rescue Rover
Semester 1
Semester 2
Task
September
October
November
December
January
February
March
April
Precedent Research
Component Specifications
Design Prototype Rover
Design Prototype Controller
Define Electrical Needs
Obtain Component Measurements
Define Mounting Pts & Weight Distribution
Model Main Body in CAD Software
Develop Code for L. Propulsion System
Design Circuit for L. Propulsion System
Print Main Body
Install & Test L. Propulsion System
Model Payload in CAD Software
Develop Code for R. Propulsion System
Design Circuit for R. Propulsion System
Print Payload
Install & Test R. Propulsion System
Develop Code for Fire Extinguishing Payload
Design Circuit for Fire Extinguishing Payload
Model Controller in CAD Software
Install & Test Payload
Develop Code for Heat Sensing System
Design Circuit for Heat Sensing System
Install & Test Heat Sensing System
Develop Code for Communication Link System
Design Circuit for Communication Link System
Install & Test Communication System
Print Controller
Develop Code for Rover Live Feed
Design Circuit for Rover Live Feed
Install & Test Live Feed
Design Smart Features Mount for Rover
Model Mount in CAD Software



Key	Color
Lea Tice	
Edgar Fernandez	
Gavin Schumaker	
Lea/ Edgar	
Edgar/ Gavin	
Lea/ Gavin	
All Three Members	

WBS Chart



Smart RC Rescue Rover
Product Specifications
Design
Build
Test
Component Specifications
Bill of Materials
Electrical Calculations
Prototype Design
Remote Control Design
Frame/ Housing
Steering Axel
Propulsion Axel
Arm/Hook/ Bucket
Controller & Controller Comms
Live Feed Camera & Camera Comms
Operational Tests

Schedule & Timeline

The schedule and timeline are based on three phases for every sub-system to be designed. These systems include the propulsion system, autonomous system, payload system, thermal sensing system, live-feed system, and communication system between the rover and controller. The three phases each system must go through are Phase 1 - research & design, Phase 2 – prototyping, & Phase 3 – integration & testing.

Propulsion System

The development of the propulsion system has been completed. Phase 1 consisted of research into what motors to use to drive the rover. The focus was torque in this case. We wanted to ensure our rover could traverse uneven terrain without stalling which led to the selection of a high torque 1:90 gearbox motor. There were two options, one was a bi-metal motor, and one was an all-metal motor. The all-metal motors were selected due to the concern of gear wear. Plastic gears are notorious for lower durability than their metal counterparts.

Phase 2 focused on designing the circuit and code for the propulsion system. There were issues with the L298N motor driver and power consumption. A 6 V power supply was added to the system to supply the drivers with more voltage and current to power two motors. After the power issue was resolved, another issue presented itself. The motors were not responding as expected with the wired test controllers' joysticks. This required calibration of the PWM signals. We had to use a constrain() function in our code to prevent overflow which crashed our system. Some of the math was backwards and therefore when pressing forward on the right joystick caused the right motor to spin backwards instead of forwards. This was a simple fix in the code and after troubleshooting and identifying the issue, the system worked as expected.

Phase 3 consisted of testing the circuit on the temporary housing since the 3D model had yet to be printed. All tests were a success with no complications. Since the model of the main body of the rover had not been printed yet, integration of the propulsion system is behind as well as final systems testing.

Autonomy System

The next planned system is the autonomous system. The development of the Autonomy System was accelerated in the project schedule, taking precedence over the Payload System due to its critical role in the rover's core "scout" functionality. Phase 1 consisted of researching different sensors that could be used in the obstacle and collision avoidance logic. Due to availability and cost, the rover will be using ultrasonic sensors for this capability.

Phase 2 has begun and focuses on firmware logic but has not been completed. The firmware will require three sensors to be read from simultaneously without blocking the main control loop. An unknown factor is how well the selected sensors do in smoky or low-light conditions. While ultrasonic technology is inherently immune to low-light conditions, the sensors' reliability in dense smoke remains a variable that requires further testing.

Phase 3 will focus on the complex task of integrating this collision-avoidance logic with the existing propulsion firmware. The design engineer will also need to design a custom mount that will attach to the payload and house the three sensors without interfering with the rest of the components to be housed in the payload.

Budget & Costs

This rover is designed to be a cost-effective alternative to market-ready first-responding units, which often rely on expensive proprietary hardware and specialized service contracts. By leveraging commercial components, such as standard TT gearbox motors and Arduino microcontrollers, this project demonstrates that functional search-and-rescue capabilities can be achieved at a fraction of the industrial price point. This open-architecture approach not only reduces initial build costs but also simplifies future maintenance and repair for the end-user.

Funding

This project is self-funded by the students in the project team. Some resources were also funded by the department of electronic and computer engineering technology. Each student will contribute to the total cost of the project depending on hardware needed.

Table 2

Bill of Materials

Component	Category	Cost	Qty	Total	Funded By
TT Motor All-Metal Gearbox – 1:90 Gear Ratio	Propulsion	\$5.95	4	\$23.80	Lea
Snap-on Hub for TT Motor	Propulsion	\$1.95	4	\$7.80	Lea
Supporting Swivel Caster Wheel – 1.3” Diameter	Propulsion	\$1.95	2	\$3.90	Lea

2 x 4 AA Battery Holder w/Leads	Power	$5.98	2	$11.96	Lea
4 PCS 60 mm Rubber Tires w/Wheels	Propulsion	$8.99	1	$8.99	Lea
3D Print Filament	Chassis	X	X	X	ISU
48 AA Batteries	Power	$9.99	1	$9.99	Lea
4 x BOJACK L298N Motor Drivers	Propulsion	$9.99	1	$9.99	Lea
3 x HiLetgo MPU-6050 Accelerometer/Gyroscope Sensor	Autonomy	$10.99	1	$10.99	Lea
HiLetgo MLX90614ESF Infrared Temperature Module	Fire Suppression	$15.99	1	$15.99	Lea
ELEGOO HR-SR04 Ultrasonic Distance Sensor	Autonomy	$8.99	1	$8.99	Lea
3 x HiLetgo ESP32	Communication	$17.99	1	$17.99	Lea
2 x Hosyond ESP32-CAM	Live Feed	$15.99	1	$15.99	Lea
3x KEXIN 32GB Micro SD Card	Live Feed	$15.99	1	$15.99	Lea
5 x ACEIRMC VL53LOX ToF Distance Sensor	Autonomy	$13.99	1	$13.99	Lea
Creality K1C 3D Printer	Manufacturing	$379.00	1	$379.00	Lea
Creality Hyper PLA CF 2KG Bundle	Manufacturing	$34.00	1	$34.00	Lea

Risk Management

Every engineering project has risks. Some of the potential risks to the successful completion of the rover identified include technical, safety and scheduling risks.

Technical Risk

This primary technical risk of this project is integration of the different hardware systems and creating firmware for the different systems that communicate with each other. The hardware and software will be developed in parallel due to the functions and feedback of the sensors relying on each other.

Another risk we already tackled was the current handling of the L298N motor drivers may draw high currents during stall conditions or steep climbs that may exceed torque specifications, potentially leading to component failures. To mitigate this risk, logic in the propulsion system firmware constrains the values of the PWM pins used to prevent stalling.

A critical risk identified is the overheating of electronic components, specifically the L298N motor drivers under load. To mitigate this, the chassis features a ventilated, open-airflow architecture. This design maximizes passive convective cooling, ensuring effective heat dissipation without the need for active fans or additional power consumption.

Latency of the line-of-sight video feed is a secondary risk we are predicting. High latency would make remote operation of the rover extremely difficult and increase the likelihood of damage to the rover if tested in certain environments. The plans for mitigation are prioritizing connection stability over resolution.

Safety Risk

The rover is powered by a bank of AA batteries. In the event of a short circuit, these batteries can generate significant heat which poses a burn or fire hazard. The open-airflow design of the chassis was meant to prevent heat accumulation however, I do not know how reliant this will be once all components are assembled on the rover.

The fire suppression system that is planned for the rover will utilize water. This will introduce the risk of water coming into contact with some of our exposed circuitry which could potentially cause permanent damage to the components or a short circuit. The plan to mitigate this issue is to design a physically compartmentalized area for the fire suppression system. The fluid reservoir and pump are planned to be housed behind the circuitry on the back of the rover, and the payload delivery system (hose) will be traced along the bottom of the rover away from the components and into the payload housing.

Schedule Risk

Fabrication bottlenecks present a significant risk because this project relies heavily on 3D printed components. Since the team will be utilizing university facilities for production of the models, fabrication timelines are subject to print lab availability and student demands from other courses. To mitigate this, early coordination with the university print lab staff needs to be prioritized by the design engineer. Regular status updates will need to be maintained with the staff as well to anticipate delays. Scheduling of these timelines has been given longer timeslots than other tasks in the project to help with this issue as well.

Most components to be used in the rover will be pre-built modules such as the 1:90 gearbox motors. Variable shipping times and availability of components could throw the schedule off track. All critical hardware has been sourced during the research and design phase, but not all have been ordered. A list of approved venders has been created and maintained for component selection during development. This will ensure alternative procurement options are available should the stock of an item be out from a primary vendor.

CHAPTER 4

ROVER & CONTROLLER DESIGN

Currently we do not have a concrete design for the controller of the Rover. However, the design will be ready for the second semester of our senior project, in turn this will allow us to have the fully modeled controller ready as well. The controller's design will be modeled to allow for all of the necessary components to fit inside of it. Secondary to this, focusing on how comfortable it will be to handle will be the next step. Modeling of the Rover itself has progressed as far as two major components for the body of the prototype, with one part being 3D printed and ready for use. Pictured below is the part that has been printed and is ready for use.

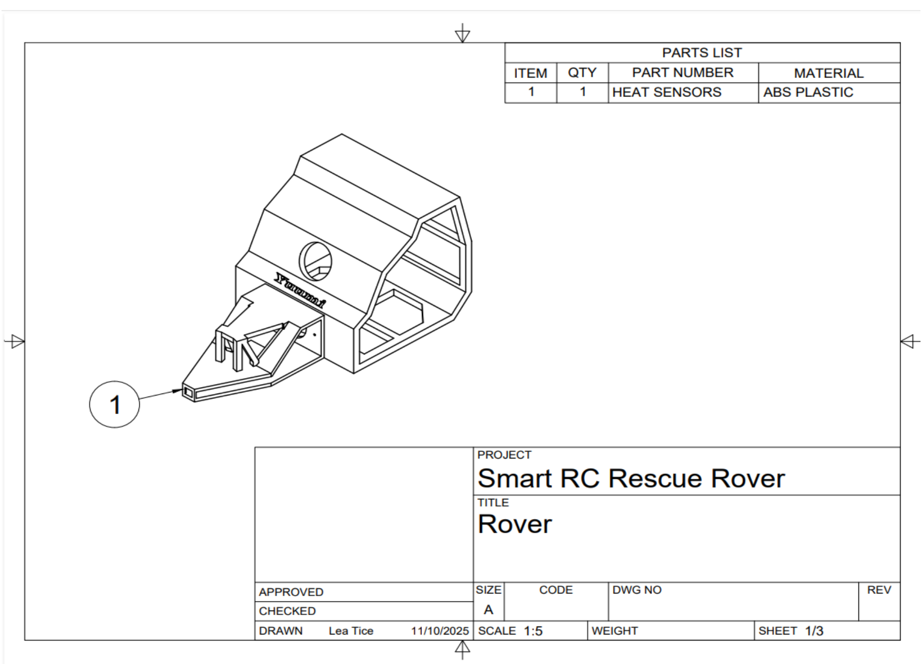


Figure 1. Isometric view of completed rover model.

CHAPTER 5

ELECTRONIC CIRCUIT DESIGN

The electronic architecture of the Smart RC Rescue Rover is centered around an Arduino microcontroller. This platform was selected for its 5V logic compatibility, extensive library support for C firmware, and sufficient General-Purpose Input/Output (GPIO) pins to handle both the propulsion and future sensor sub-systems.

Propulsion System

The system operates on a dual-voltage standard: a high-current rail for the electromechanical components and a regulated low-voltage rail for signal processing. The schematic documented in Appendix B illustrates the completed wiring for the propulsion system. The circuit utilizes a BOJACK L298N Dual H-Bridge motor driver to manage the high-current demand of the two DC gear motors.

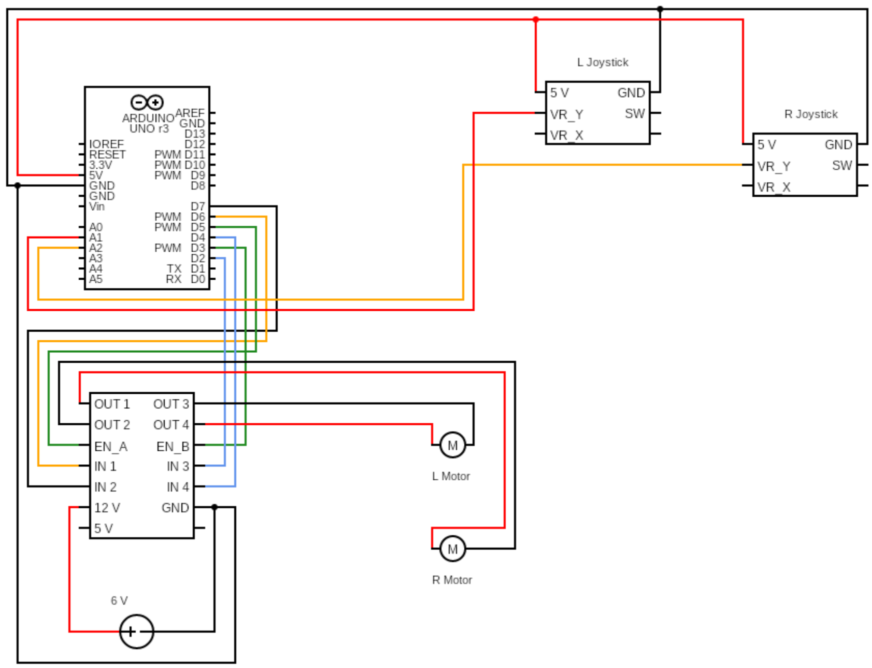


Figure 2. Wiring schematic of propulsion system.

Motor Power

A dedicated 6V DC supply (derived from the AA battery bank) is connected to the 12V input terminal of the L298N driver. This ensures the motors have direct access to the battery current.

Common Ground

A common ground (GND) reference is established between the 6V battery source, the L298N driver, and the Arduino Uno to ensure accurate signal transmission.

L298N Interface

The L298N interfaces with the Arduino via six digital control lines. The driver's Enable pins (EN A and EN B) are connected to Arduino PWM-capable pins 3 and 5, respectively. This allows the firmware to adjust the duty cycle of the voltage sent to the motors, controlling their speed. The H-Bridge logic pins determine the polarity of the motors. IN 1 and IN 2 (controlling the Left Motor) are wired to digital pins 2 and 4. IN 3 and IN 4 (controlling the Right Motor) are wired to digital pins 7 and 6.

CHAPTER 6

SOFTWARE DESIGN

The firmware for the Smart RC Rescue Rover is developed in C using the Arduino Integrated Development Environment (IDE). The software architecture utilizes standard control structures and logic loops [1]. In this design, the main loop() function sequentially calls distinct sub-routines for propulsion, sensor data acquisition, and payload management.

Propulsion System

The core mobility logic, documented in Appendix A, implements a differential steering ("tank drive") algorithm. This system translates user inputs from the handheld controller into PWM signals for the motor drivers.

Input (Data Acquisition)

The firmware reads the analog voltage from the joystick potentiometers using the analogRead() function. This returns a raw integer value between 0 and 1023, representing the physical position of the stick.

Signal Processing

The raw input data is processed to match the motor driver's requirements. The map() function is used to convert the 10-bit input range (0-1023) into an 8-bit PWM output range (0-255). To prevent "ghosting" (unwanted movement when the sticks are neutral), the code includes a conditional check: if (abs(value) > 20). This ensures that minor sensor fluctuations near the center position are ignored, and the motors remain idle until a deliberate command is input.

Output (Motor Actuation)

The firmware reads the analog voltage from the joystick potentiometers using the analogRead() function. This returns a raw integer value between 0 and 1023, representing the physical position of the stick. The processed speed and direction values are sent to the L298N driver via the digitalWrite() and analogWrite() functions. The logic splits the signal: the magnitude of the value determines the PWM duty cycle (speed), while the sign (positive/negative) toggles the H-Bridge pins to determine direction (Forward/Reverse).

CHAPTER 7

CONCLUSION

This concludes all our progress on the "Smart RC Rescue Rover" so far. The next couple of months will be used for preparation and assembly of the components for the rover. It will also include all the installation of the various components and making sure they both work individually and in sync with each other when the rover is completed. We also plan to run some tests with the final version of the rover by recreating scenarios where first responders are unable to enter. This is to ensure that this rover is fully capable of what it was built for.

REFERENCES

[1] K. N. King, *C Programming: A Modern Approach*, 2nd ed. New York, NY: W. W. Norton & Company, 2008.

[2] T. Gaddis, *Starting Out with Programming Logic and Design*, 2nd ed. Boston, MA: Pearson, 2010.

[3] R. L. Boylestad, *Introductory Circuit Analysis*, 12th ed. Upper Saddle River, NJ: Pearson, 2010.

[4] R. L. Boylestad and L. Nashelsky, *Electronic Devices and Circuit Theory*, 11th ed. Upper Saddle River, NJ: Pearson, 2013.

[5] M. N. Horenstein, *Design Concepts for Engineers*, 3rd ed. Upper Saddle River, NJ: Prentice Hall, 2010.

[6] Autodesk, "Fusion 360: Integrated CAD, CAM, and CAE software," *Autodesk.com*, 2025. [Online]. Available: https://www.autodesk.com/products/fusion-360/overview.

[7] Indiana State University, "ECT 437 Senior Project Guidelines," *Department of Electronics and Computer Engineering Technology*, 2025.

APPENDIX A: CODE

Appendix A serves as the comprehensive repository for the C++ firmware source code developed for the Smart RC Rescue Rover's embedded control systems. The software architecture is modular, with distinct codebases dedicated to each functional sub-system of the rover. As development progresses through each planned system, this appendix will be updated to include the source code for each sub-system.

Propulsion System v1

This firmware interfaces with the dual-joystick handheld controller to read analog inputs, apply dead zone correction to prevent signal drift, and map the values to Pulse Width Modulation (PWM) signals. These signals drive the L298N motor controller, allowing for precise forward, reverse, and rotational maneuvering of the high-torque DC motors.

```
/*
  Dual Joystick to L298N Motor Control Test (Tank Drive) - FIXED
  - Added 'constrain()' to prevent motor stall at full stick.
*/

// --- Joystick Pins ---
#define LEFT_JOY_Y_PIN A1
#define RIGHT_JOY_Y_PIN A2

// Default joystick center values (0-255 range)
#define JOY_CORRECTION 128

// --- LEFT Motor Pins ---
#define LEFT_ENABLE_PIN 3   // PWM
#define LEFT_IN1_PIN 2
#define LEFT_IN2_PIN 4

// --- RIGHT Motor Pins ---
#define RIGHT_ENABLE_PIN 5  // PWM
#define RIGHT_IN3_PIN 7
#define RIGHT_IN4_PIN 6

void setup() {
```

```
  Serial.begin(115200);
  Serial.println("--- ROVER READY ---");

  // Set Left Motor pins
  pinMode(LEFT_ENABLE_PIN, OUTPUT);
  pinMode(LEFT_IN1_PIN, OUTPUT);
  pinMode(LEFT_IN2_PIN, OUTPUT);

  // Set Right Motor pins
  pinMode(RIGHT_ENABLE_PIN, OUTPUT);
  pinMode(RIGHT_IN3_PIN, OUTPUT);
  pinMode(RIGHT_IN4_PIN, OUTPUT);
}

void loop() {
  // ==================================================
  // 1. READ JOYSTICKS (With Anti-Ghosting)
  // ==================================================
  int rawLeft = analogRead(LEFT_JOY_Y_PIN);
  delay(2); // Tiny delay to prevent crosstalk
  int rawRight = analogRead(RIGHT_JOY_Y_PIN);

  // ==================================================
  // 2. CONTROL LEFT MOTOR
  // ==================================================
  // Map 0-1023 input to 0-255 range
  int leftY = map(rawLeft, 0, 1023, 0, 255) - JOY_CORRECTION;
  leftY = leftY * -1; // Flip Y-axis so UP is positive

  int leftSpeed = 0;

  // Calculate Speed
  if (abs(leftY) > 20) { // Deadzone check
    if (leftY > 0) {
      leftSpeed = map(leftY, 20, 127, 0, 255);
    } else {
      leftSpeed = map(leftY, -20, -127, 0, -255);
    }
  }

  // --- THE FIX: CONSTRAIN VALUES ---
  // Ensure speed never goes above 255 or below -255
  leftSpeed = constrain(leftSpeed, -255, 255);
```

```
  // ====================================================
  // 3. CONTROL RIGHT MOTOR
  // ====================================================
  int rightY = map(rawRight, 0, 1023, 0, 255) - JOY_CORRECTION;
  rightY = rightY * -1; // Flip Y-axis so UP is positive

  int rightSpeed = 0;

  // Calculate Speed
  if (abs(rightY) > 20) { // Deadzone check
    if (rightY > 0) {
      rightSpeed = map(rightY, 20, 127, 0, 255);
    } else {
      rightSpeed = map(rightY, -20, -127, 0, -255);
    }
  }

  // --- CONSTRAIN VALUES ---
  rightSpeed = constrain(rightSpeed, -255, 255);

  // ====================================================
  // 4. SEND COMMANDS
  // ====================================================
  // Serial.print("L: "); Serial.print(leftSpeed);
  // Serial.print(" | R: "); Serial.println(rightSpeed);

  controlLeftMotor(leftSpeed);
  controlRightMotor(rightSpeed);

  delay(50);
}

// --- Helper Functions ---

void controlLeftMotor(int speed) {
  if (speed > 0) {
    digitalWrite(LEFT_IN1_PIN, HIGH);
    digitalWrite(LEFT_IN2_PIN, LOW);
    analogWrite(LEFT_ENABLE_PIN, speed);
  } else if (speed < 0) {
    digitalWrite(LEFT_IN1_PIN, LOW);
    digitalWrite(LEFT_IN2_PIN, HIGH);
    analogWrite(LEFT_ENABLE_PIN, abs(speed));
  } else {
```

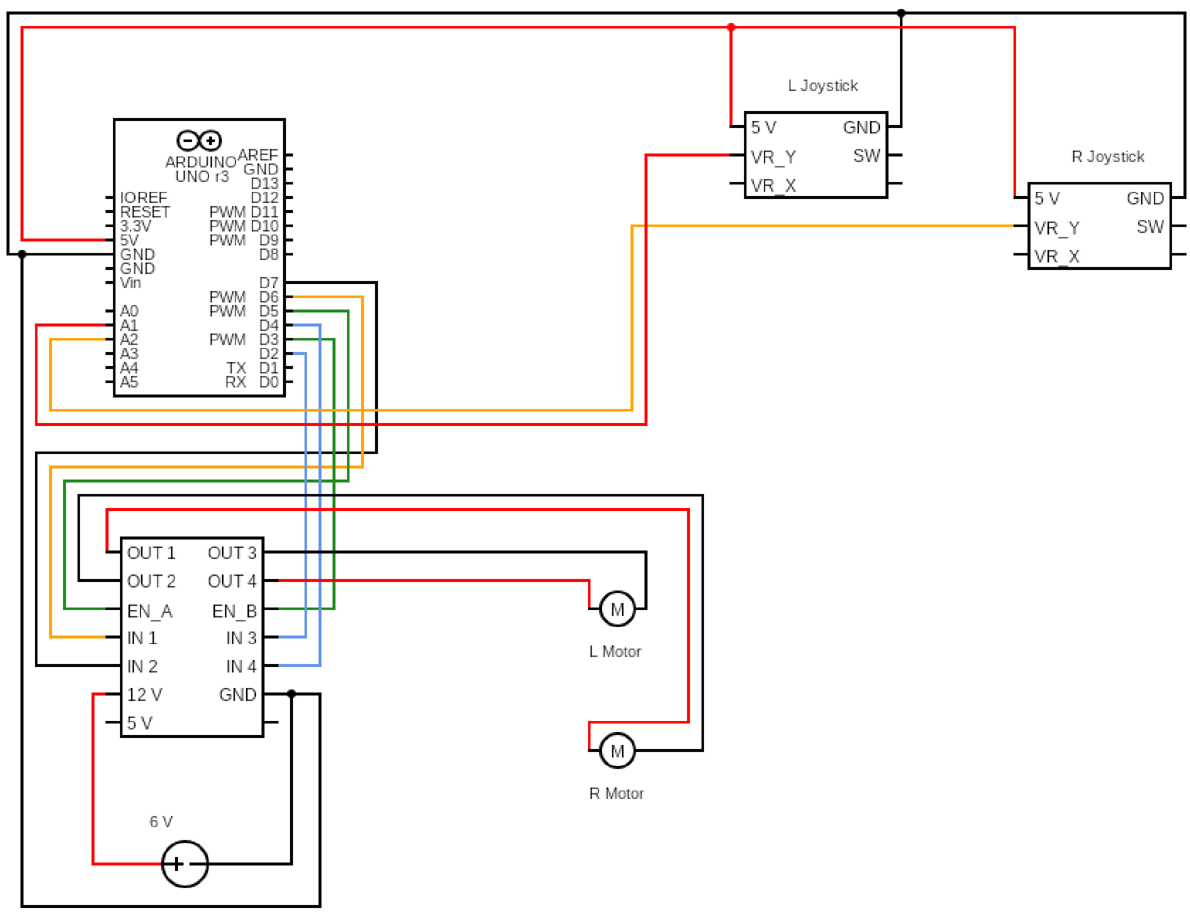
```
    digitalWrite(LEFT_IN1_PIN, LOW);
    digitalWrite(LEFT_IN2_PIN, LOW);
    analogWrite(LEFT_ENABLE_PIN, 0);
  }
}

void controlRightMotor(int speed) {
  if (speed > 0) {
    digitalWrite(RIGHT_IN3_PIN, HIGH);
    digitalWrite(RIGHT_IN4_PIN, LOW);
    analogWrite(RIGHT_ENABLE_PIN, speed);
  } else if (speed < 0) {
    digitalWrite(RIGHT_IN3_PIN, LOW);
    digitalWrite(RIGHT_IN4_PIN, HIGH);
    analogWrite(RIGHT_ENABLE_PIN, abs(speed));
  } else {
    digitalWrite(RIGHT_IN3_PIN, LOW);
    digitalWrite(RIGHT_IN4_PIN, LOW);
    analogWrite(RIGHT_ENABLE_PIN, 0);
  }
}
```

APPENDIX B: SCHEMATICS

Propulsion System v1

The following schematic details the integration of the Smart RC Rescue Rover's propulsion sub-system. This circuit interfaces the Arduino microcontroller with the BOJACK L298N H-Bridge motor driver



APPENDIX C: LOG

Date	Activity	Team Member	Task Details / Notes
10/28/2025	**Project Initialization**	**Lea**	Defined Project Scope: Overview, Deliverables, Constraints, and Uncertainties.
10/28/2025	**Project Initialization**	**Gavin**	Defined Acceptance Criteria for the rover.
10/28/2025	**Project Initialization**	**Edgar**	Defined initial Features & Functions list.
10/28/2025	**Subsystem Research**	**Lea**	Conducted component research for: Propulsion Axle, Payload, Heat Sensors, Live Feed Camera, Obstacle Avoidance modules, Alarms, and Communication protocols.
10/28/2025	**Computer-Aided Design**	**Lea**	Completed initial 3D model of the rover in Fusion 360. Exported .3mf files for printing.
11/09/2025	**Procurement & Manufacturing**	**Lea**	Transmitted rover design files (.3mf) to Gavin for 3D printing.
11/15/2025	**Status Verification**	**Gavin**	Status Update: Confirmed receipt of files
11/17/2025	**Fabrication Update**	**Gavin**	Submitted processed print files for the chassis & payload.
12/02/2025	**Mechatronic Prototyping**	**Lea**	Designed and coded the propulsion system firmware; assembled a temporary rover chassis for the presentation and built a fully functional wired temporary controller for demonstration.
12/02/2025	**Controller Design**	**Gavin**	Scheduled start date for 3D modeling of the handheld controller. Deadline set for Jan 12, 2026.
12/12/2025	**Final Report Draft**	**Lea**	Completed sections: Abstract, Preface, Acknowledgments, Project Management, Electronic Design, Software Design, Appendices A/B/C/D.
12/12/2025	**Final Report Draft**	**Edgar**	Assigned sections: Background, Conclusion.
12/12/2025	**Final Report Draft**	**Gavin**	Assigned sections: Introduction, Rover & Controller Design.
12/12/2025	**Presentation Prep**	**Lea**	Designed slide deck layout; created "Our Project," "Key Features," and "Progress" slides.
12/12/2025	**Presentation Prep**	**Gavin**	Prepared content "Challenges," "Mechanical Design Goals," and "Weight Distribution."

Date	Activity	Team Member	Task Details / Notes
12/12/2025	Presentation Prep	Edgar	Prepared content "Materials for Assembly," "Planned Systems & Integrations," and "Next Steps."
01/05/2026	Hardware Retreival	Lea	Picked up Yuumi Rover main body from print lab
01/06/2026	Hardware Integration	Lea	Soldered pins for gyroscope, IR temp sensor, & time of flight sensors
01/07/2026	Autonomous Logic Development	Lea	Defined sensors in autonomous systems usage and plan, started developing firmware for whole autonomy system
01/13/2026	Schedule Assessment	Lea	Assessed status of controller design; evaluated potential delays
01/15/2026	Inter-team integration	Edgar	Coordinated with design lead for physical handover of payload
01/16/2026	Unit Testing & Verification	Lea	Developed VL53L0X Time of Flight sensor & MPU-6050 Gyroscope unit test files
01/16/2026	System Validation	Lea	Verified MPU-6050 Gyroscope & VL53L0X TOF sensor accuracy; passed all diagnostic parameters
01/16/2026	Coordination	Gavin	Updated team on progress of controller design, ETA given: 01/19
01/16/2026	Mechanical Design	Lea	Modeled and printed VL53L0X encasings
01/17/2026	Mechanical Design	Gavin	Updated team on controller 3D model deliverable, adjusted project schedule
01/17/2026	Video Pipeline Implementation	Lea	Developed firmware for real-time video streaming and ESP32-CAM integration
01/17/2026	Network Architecture	Lea	Configured rover Access Point (AP) and established communication protocols for camera feed.
01/19/2026	Mechanical Design	Lea	Modeled and printed ESP32-CAM mount
01/24/2026	Asset Acquisition	Lea	Downloaded RC tire molds for yuumi rover, modeled by BuildItBetter

Date	Activity	Team Member	Task Details / Notes

APPENDIX D: APPROVED VENDORS

https://www.adafruit.com/

https://www.sparkfun.com/all-categories

https://www.amazon.com/

https://www.digikey.com/

https://www.automationdirect.com/adc/home/home