IGVC 2026 Design Report

Lawrence Technological University

Robotic Engineering – AutoNav

Team: Blue Devil Robotics (LTU Eng)

Robot: Turbo Blue 2.0

Date: May 15 2026

Team Captains	Alyssa Bloomfield	abloomfie@ltu.edu
	Travis Bowman	tbowman@ltu.edu
Team Members	Miguel Aguirre del Valle	maguirred@ltu.edu
	Anthony Gabrail	agabrail@ltu.edu
	Amaya Pitts	apitts2@ltu.edu
	Sam Stepanek	sstepanek@ltu.edu
Faculty Advisor	Gaurav Singh	gsingh@ltu.edu

Faculty Advisor Statement:

I, Gaurav Singh of the Department of Mechanical, Robotics, and Industrial Engineering at Lawrence Technological University, certify that the design and development of Turbo Blue 2.0 research platform by the individuals on the design team is significant and is either for-credit or equivalent to what might be awarded credit in a senior design course.

Gaurav Singh

1. Team Identification and Organization

1.1. Team Introduction

The IGVC AutoNav team at Lawrence Technological University is composed of six senior Robotic Engineering students supported by one faculty advisor. The team is organized into three primary subsystem groups — Mechanical, Electrical, and Software — with two team captains coordinating overall project direction, design reviews, and integration. Roles were assigned based on each member's coursework focus, prior project experience, and area of interest, with deliberate cross-staffing so that every subsystem has at least two contributors. Team meetings follow a weekly status format in which each member presents the work completed during the previous week, raises any concerns or blockers, and outlines plans for the coming week. This cadence keeps every member accountable, surfaces integration issues early, and ensures that progress in one subsystem is visible to the others.

1.2. Team Organization

Table 1. Team Members and Responsibilities

Name	Degree Program	Primary Responsibilities	Hours
Alyssa Bloomfield	B.S. Robotics Engineering	Team Captain, Systems Design	600
Travis Bowman	B.S. Robotics Engineering, B.S. Computer Science	Team Captain, Software Design	600
Miguel Aguirre del Valle	B.S. Robotics Engineering	Electrical Design	250
Anthony Gabrail	B.S. Robotics Engineering	Mechanical Design	400
Amaya Pitts	B.S. Robotics Engineering	Electrical Design	300
Sam Stepanek	B.S. Robotics Engineering	Mechanical Design	450

2. System and Subsystem Requirements

2.1. Engineering Processes

The engineering process followed for this project was structured around identifying high-level system requirements derived directly from the IGVC 2026 competition rules and overall performance objectives. These requirements included constraints on vehicle size (2–4 ft width, 3–7 ft length, and a maximum height of 6 ft), payload capacity (minimum 20 lb), safety systems (manual E-stop, remote E-stop, indicator light), and autonomous operation. The team treated these competition rules as primary design constraints and translated them into measurable engineering specifications.

From these requirements, a system-level architecture was developed, beginning with the mechanical design. The chassis and suspension subsystem were designed to support at least a 20 lb payload while maintaining a low center of gravity for stability. Additional mechanical requirements included mounting locations for sensors (LiDAR and camera positioned at the front, GPS centered on the top rack, and IMU near the center of mass), as well as ensuring sufficient ground clearance and structural rigidity. Decision matrices were used to evaluate different design options, such as different chassis, based on criteria including weight, cost, manufacturability, and adjustability. This led to the selection of a simplistic "box" style frame made of T-slot aluminum for its balance of strength and adaptability.

Once the mechanical architecture was established, focus shifted to the electrical and software subsystems. Requirements for these systems included reliable power distribution to multiple components (sensors, Raspberry Pi, and motors), real-time data processing for perception, and control of the drivetrain. Measurable requirements included communication reliability between sensors, as well as response times for obstacle detection for vehicle control. The software subsystem was required to provide sufficient environmental awareness through LiDAR and camera data, while the control system was responsible for translating this data into accurate and stable vehicle motion.

Each subsystem was developed and refined through iterative design methods, including CAD modeling, simulation, and team-based design reviews. Trade-offs between cost, complexity, and performance were evaluated to ensure the final design met all requirements. This approach ensured that all subsystems remained aligned with the overall objectives and could be effectively integrated into a functional autonomous vehicle.

2.2. Mechanical Design Requirements

For mechanical design, there were four main constraints that were identified: Vehicle Size, Mechanical Contact, Payload, and Speed. These requirements are all designated by the 2026 rulebook. Regarding the vehicle size, it must be within the specified length, width, and height limits. The IGVC rules state that the size of the vehicle must remain within the restrictions. The length of the robot must be within 3 to 7 feet while the width must be within 2 to 4 feet. There is no minimum height, but a maximum of 6 feet. This will be evaluated by measuring the farthest points on each axis of the vehicle using a tape measure. The target values would be any value within the necessary bounds. For mechanical contact, the design of the vehicle must contain a form of propulsion that maintains contact with the ground, such as wheels, tracks, or pods. The IGVC rules state that the design must be a ground vehicle that is propelled by direct mechanical contact to the ground, with suggestions of wheels, tracks, pods, etc. The evaluation will be done by visual inspection before and during travel of the vehicle, with the goal of visualling seeing the vehicle maintaining contact during propulsion. The other two important design concepts are ability to secure a payload and the vehicle speed. The IGVC rules state that the design must be able to support a 16” x 8” 20 lb payload and ensure it remains secure while the robot is in motion. The design will include a section for this specific size payload in it. In order to validate the efficiency of the mount, the team will use pass/fail criteria. Testing will include mounting the payload into the chassis and then using TeleOp practices to run the robot and make sure the payload remains secure. The IGVC rules state that the vehicle must travel a minimum of 1 mph and a maximum of 5 mph. This speed limit will be primarily addressed with code rules. Testing using TeleOp and autonomous mode will be measured with a radar in order to confirm that the appropriate speed limit is met.

2.3. Safety Design Requirements

There are two main components of the safety system: E-Stop and Safety Light. Each robot is required to have both a hardwired mechanical E-Stop and a wireless E-Stop. The IGVC rules require a mechanical emergency stop system to ensure immediate shutdown of the vehicle in unsafe conditions. Visual inspection of a round 1” button located in the rear of the chassis will validate that this component is present. Additionally, this requirement will be verified by measuring that the vehicle comes to a complete stop upon activation. The IGVC rules require a wireless emergency stop system to ensure immediate shutdown of the vehicle in unsafe conditions. Visual inspection of a labeled button on the tele-op controller will validate that this component is present. Additionally, this requirement will be verified by measuring that the vehicle comes to a complete stop upon activation. IGVC rules require a visible safety indicator light to communicate the vehicle’s operational state to surrounding personnel. This will be validated by ensuring the light visibility, and accurate state indication at all times.

2.4. Electrical Design Requirements

Electrical system requirements were defined based on individual component requirements and IGVC rules. A majority of the sensing components were already owned by the team and therefore the power requirements were predefined. For this reason, the electrical subsystem is split into two grounded power buses: a 24V high-power bus (two 12V 35Ah lead-acid batteries connected in series) dedicated to propulsion, and a 12V 22Ah accessory bus that feeds a 5V DC-DC converter, a 12V-to-110V AC inverter, and all sensors and computers. This is in compliance with the need for an onboard power system. In addition to the component defined requirements, it was also important to design around the IGVC competition rules. Page 13 of the 2026 IGVC rulebook states that "Judges will disqualify any vehicle which appears to be a safety hazard, degrades the course or violates the safety requirements during the competition." For this reason, the team utilized wire harnesses for longer cords. Insulated wires and shrink tube were used for routing the electrical system. This ensured that there was no unintended wire contact and added reassurance to solder joints. As a methodology of checking this design requirement was achieved, visual inspections would occur. In addition to the visual inspection, it was important to the team to keep the electrical system organized and labeled. One key design component was ensuring one ground, especially considering the dual battery system. This requirement was confirmed by ensuring all ground wires and terminals were green. Green wires were also not to be used for any other connections. According to page 11 of the 2026 IGVC rulebook, no tactical sensors should be used. To compile with this rule, the team ensured to only use position and motion sensors in the design of the robot.

2.5. Perception Requirements

Perception will be accomplished by two main components: white line detection and obstacle detection. For white line detection, The robot must identify the white painted lines of the course boundary using a camera. As defined by the IGVC rulebook, a lane is defined by white lines on an asphalt parking lot. This requires a vision pipeline capable of reliably differentiating the white lines from the asphalt. For obstacle detection, The robot must detect and localize physical obstacles like barrels, potholes, or barriers placed on the course. This requires a sensor that provides spatial data to feed into the costmap for collision avoidance.

2.6. Driving Logic Requirements

The driving logic was to be based around following boundaries and avoiding obstacles. The robot is required not to cross the white lines, so the driving logic must detect the line and apply corrective steering to keep the robot centered. When an obstacle is detected ahead, the robot must navigate around it without stopping indefinitely, requiring the local planner to replan dynamically.

2.7. Key Performance Indicators

The primary IGVC scoring metric is that the robot must navigate the full course without disqualifying infractions (crossing lines, hitting obstacles), so this directly measures whether perception and driving logic are working correctly. Autonomous distance traveled: IGVC scores points per distance the robot completed autonomously without human intervention or disqualifying infractions.

3. Mechanical Design

The mechanical design of Turbo Blue 2.0 is driven by IGVC competition rules, course navigation requirements, and the need to support the electrical and sensing systems on a stable platform. Turbo Blue 2.0 was derived from the previous year's IGVC vehicle (seen in Figure 1), but updated to fit current design needs. The largest adjustment that was made between Figure 1 (old) and Figure 2 (current) were adjusting the overall height. This would allow for more space to properly seat the batteries, payload, and the electrical system. Because Turbo Blue 2.0 uses differential drive, therefore, stopping is handled through the electrical and software control system by disabling or commanding zero output to the drive motors. This mechanical focus included drivetrain stability, wheel traction, chassis extension, frame rigidity, and secure component mounting.

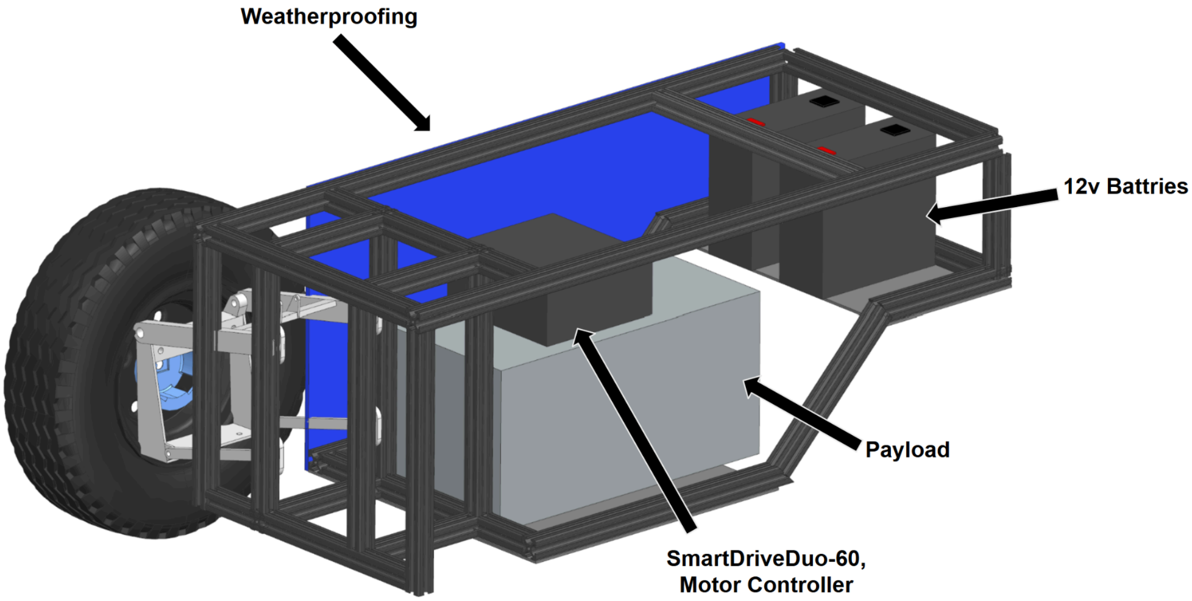

Figure 1. Mechanical design for the original Turbo Blue.

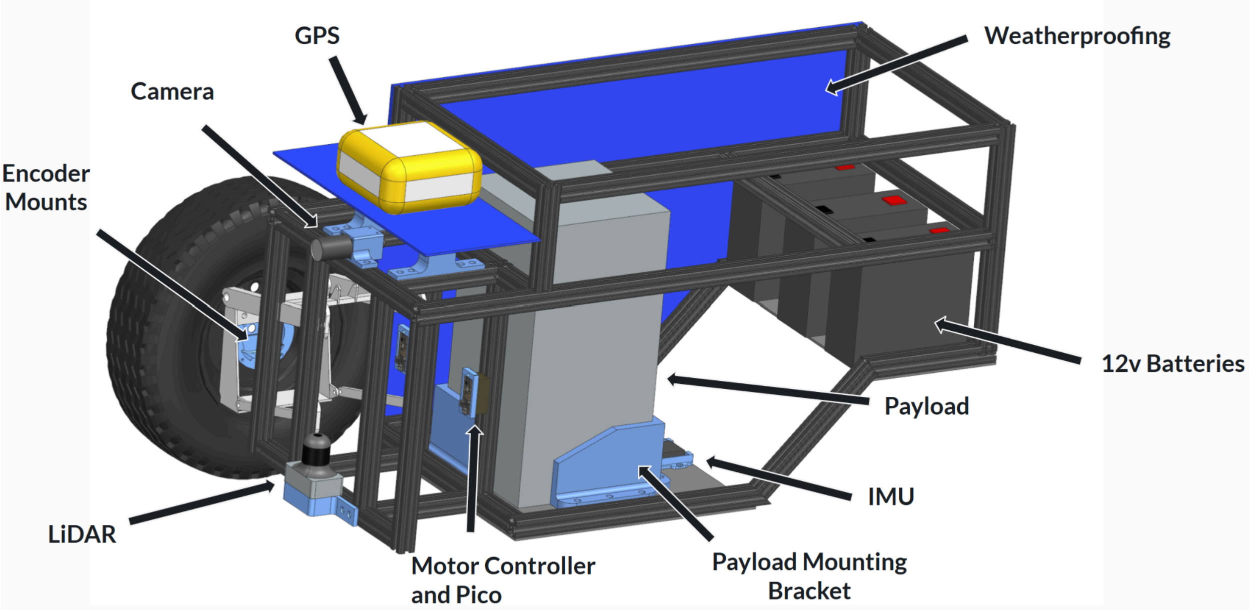

Figure 2. Updated mechanical design for Turbo Blue 2.0.

3.1. Frame and Structure

The frame is constructed from 6063-T6 T-slotted aluminum extrusion, selected for its strength, low weight, and the ability to easily mount and adjust components as needed. The chassis maintains a rectangular, box-style geometry that provides sufficient rigidity while allowing flexible integration of sensors, batteries, and payload without requiring major modifications. The principal modification made for Turbo Blue 2.0 was a vertical extension of approximately six inches at the top of the chassis (16 × 28.75 × 6 in) to provide additional internal volume for the payload, batteries, and electronics, addressing the cramped packaging of the previous version.

The decision to use T-slotted extrusion instead of a fully welded frame was made to enable easier assembly, adjustment, and future modifications, while still providing enough rigidity to support all required components. It also made mounting sensors, electronics, and weatherproofing much more straightforward compared to a fixed welded structure. Avoiding aluminum welding for non-perpendicular joints was based on the team's experience level and available resources; by prioritizing a design assembled with standard brackets, fasteners, and thread-locking Loctite, the team reduced build complexity and improved assembly reliability. It was found that a few places would require aluminum welding rather than just bracket connections due to the harsh angles. These welds occur in four places, at each end of the slanted bars.

In order to confirm the strength of the chassis multiple Finite Element Analysis (FEA) studies were completed. Both of these were done using a 25kg load placed at the low center of the chassis (similar to the location where the payload would be placed). There is a fixed support where the suspension attaches to the chassis, as well as where the caster is located. The results of these studies can be seen in Figures 3 and 4. The results of both were satisfactory, showing minimal displacement and deformation.

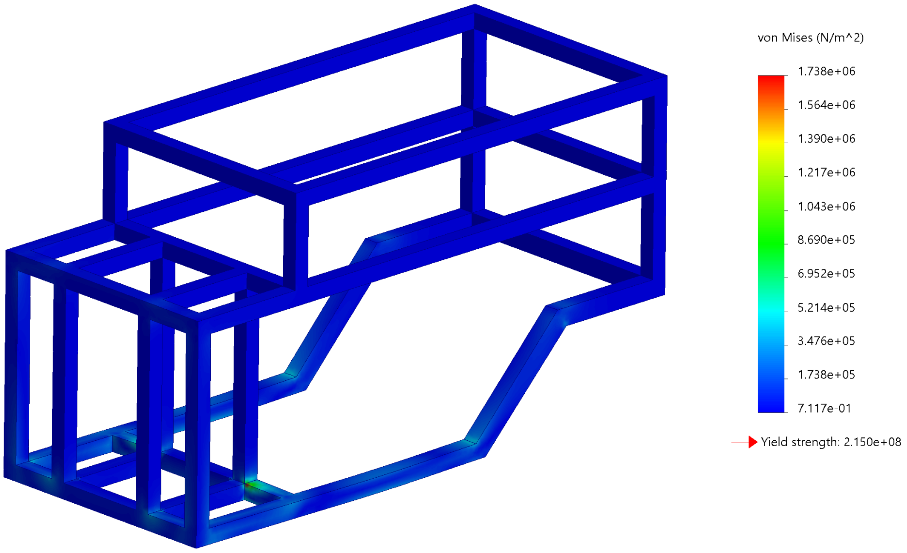

Figure 3. FEA principle stresses for Turbo Blue 2.0 chassis.

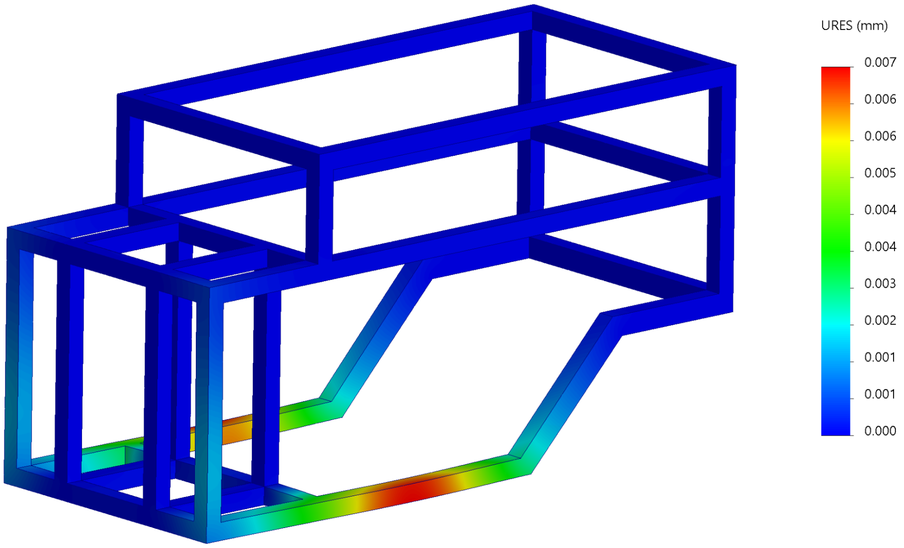

Figure 4. FEA Displacement for Turbo Blue 2.0 chassis.

3.2. Drive system and Suspension

The primary drive system is differential drive, implemented with two commercially available 24V NPC-T74 geared DC motors at the front of the chassis. These motors have a 20:1 gear ratio and an estimated continuous torque of approximately 6 Nm, which is intentionally generous to overcome the inclined section of the IGVC course. The motors are connected to the chassis through custom swing arms on each side that act as a four-bar linkage, allowing the wheels to articulate vertically while keeping the motor housings effectively fixed relative to the wheel axis. A passive caster wheel at the rear provides trailing support.

Suspension on Turbo Blue 2.0 is provided by a coil-over strut mounted between the top of the chassis and the swing arm that holds each drive wheel. The four-bar geometry of the swing arm keeps the motor and wheel correctly aligned through suspension travel, isolating the chassis from impacts and preserving traction over uneven ground.

3.3. Housing and Weatherproofing

Key components such as batteries and electronics are mounted to a baseplate within the chassis using brackets that keep them secure. Heavier components are positioned low to maintain a stable center of gravity and improve overall vehicle performance. Blue acrylic panels enclose the chassis to protect internal components from environmental conditions such as rain and debris, and Flex Seal is applied along panel seams to improve water resistance while still allowing some panels to be removable for payload placement, electrical maintenance, and debugging.

3.4. Mechanical Requirements vs. Measured Values

Table 2 compares the principal mechanical requirements with the most recently measured values on Turbo Blue 2.0.

Table 2. Measurement Comparison Chart

Mechanical Requirement	Target Values	Actual Measured Values	Measurement Method
Vehicle Length	3-7 ft	37.5 in	Tape measure, front-most to rear-most point
Vehicle Width	2-4 ft	37.5 in	Tape measure, widest point including wheels
Vehicle Height	< 6 ft	34 in	Tape measure from the ground to highest point
Ground Clearance	0.75 ft	1.25 in	Measure from ground to lowest point on the chassis
Static Load Support	System can support the full payload	No failures under load	Fully assembled robot inspection
Dynamic Load Support	No shifting after motion testing	All components remained in place	Fully assembled robot inspection

The measured dimensions all sit comfortably inside the IGVC envelope, confirming Turbo Blue 2.0 is physically eligible for competition inspection. Some wheel slip was observed during teleoperation testing, which is consequential for differential drive. Steering depends on the speed difference between the left and right wheels; further tuning is required prior to competition. The chassis showed no visible deformation, fastener loosening, or component shifting during testing with the full battery, electronics, motor controller, sensor, wiring, and payload load.

3.5. Effective Innovations In Vehicle Design

The most consequential innovation in this year's vehicle was the disciplined pivot from the original Blue Box four-wheel swivel-drive platform to the Turbo Blue 2.0 differential-drive platform once the swivel-drive moment-loading problem was identified. Rather than commit further budget and schedule to a mechanism that could not meet competition stability requirements, the team carried forward the lessons and component set from Blue Box (sensor mounts, payload bracket, weatherproofing strategy, and the entire electrical and computing stack) onto a hardened legacy chassis. Because both chassis use 6063-T6 T-slotted aluminum extrusion, the transplant required minimal redesign of mounts and brackets, demonstrating the value of designing subsystems to be platform-independent.

The original "Blue Box" concept (seen in Figure 5) featured four independent swivel-drive struts with integrated linear-rail suspension, slip rings, and lazy susan bearings to enable continuous 360° rotation of each wheel. After fabrication, the moment acting on the strut bearing relative to the mounting location proved too great for the PLA mounting components and the bearing support, leading to misalignment under load. A retainer ring riding on adjustable ball rollers was designed and fabricated to share the moment load with the chassis, but the corrected assembly still did not meet competition stability standards within the remaining schedule. With time and budget exhausted, the team transitioned to modifying the prior-year chassis as Turbo Blue 2.0. The Blue Box experience drove the current emphasis on bearing support, load-path design, and on transferring the proven sensor mounts, payload bracket, and electrical layout onto a known-good mechanical platform.

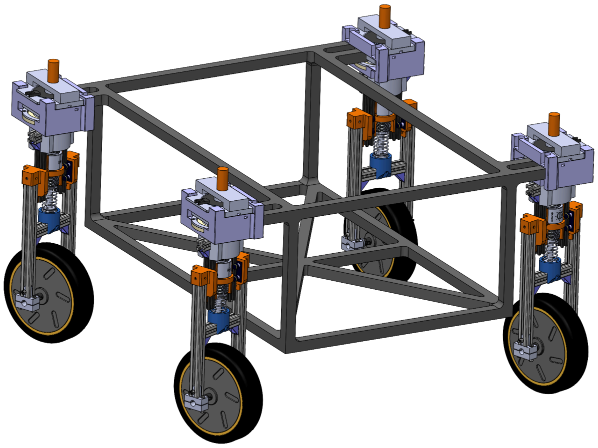

Figure 5. Previous mechanical design for Blue Box

4. Safety Design

4.1. General Safety

Safety was taken to be the highest priority subsystem when designing the entirety of the project. This was also important with the modes of the robot outside of the competition like transportation, parked, and charging. For transportation, all of the components are secured within the robot, which means during transportation, there is minimal risk of items falling apart. In order to ensure full safety during transportation, the chassis will be strapped to the vehicle to prevent any movement. For parked, due to the weight of the robot, it would require a significant initial torque for the robot to be in motion. This means that somebody would have to push the robot for it to move.Without the wheels and a constant force, the robot naturally exists at rest. For charging, both of the power systems can be easily detached using a quick connect source. The team has implemented an internal policy that each battery will be monitored while charging. This includes checks with the indicator light on the charger and also checking with a multimeter. For the 24V system, each battery will be charged separately. Once charging has been completed, the covers for the battery terminals will be placed back into the proper coverage area.

4.2. Mechanical E-Stop

The mechanical e-stop is located on the rear end of the chassis. It is in the center and located on the top panel. This button is red and one inch in diameter. It is also located on a yellow base. There is nothing within six inches that would obstruct use of this button. This allows for the e-stop to be easily recognized and accessible. When pressed, the button will immediately send the robot into brake mode. This will bring the robot to an instantaneous stop. The e-stop will also have to be unlatched for the robot to completely reset and start working at all. The robot will remain in this braked state until it is switched into tele-operations mode and then back to autonomous mode.

4.3. Wireless E-Stop

The wireless e-stop is triggered by a button on the RC controller. This button is clearly labeled, as to prevent any confusion about its purpose. As long as the RC controller is in range (500m-1,500m) of the transmitter that is located on the robot, it will be able to stop the robot. The range of the transmitter can be at the lower end if there is significant noise in the surrounding area. This functions very similarly to the mechanical e-stop. When pressed, the button will immediately send the robot into brake mode. This will bring the robot to an instantaneous stop. The e-stop will also have to be unlatched for the robot to completely reset and start working at all. The robot will remain in this braked state until it is switched into tele-operations mode and then back to autonomous mode.

4.4. Light Stack

There is a light stack with a red indicator light located on the rear side of the robot. This light will be a solid red if the robot is powered on and in tele-operations mode. The light will be a slowly blinking red if the robot is powered on and in autonomous mode. The light will be a rapid blinking red if there is an error detected with any of the electrical/software components.

4.5. Safety Requirements

The safety requirements for Turbo Blue include the mechanical E-stop, remote E-stop, safety indicator light, and safe software response to stop commands. The target is for the robot to stop safely whenever either E-stop is activated and for the safety indicator light to clearly show the robot’s current state. Table 3 shows the safety requirements, target values, most recently measured or inspected values, and the testing method used or planned for validation.

Table 3. Safety Requirement Testing for Turbo Blue

Requirement	Target Value	Measured Value	Test Method	Status
Mechanical E-stop accessibility	Red button, at least 1in diameter, rear mounted, visible, and unobstructed	Mechanical E-stop installed and accessible	Visual inspection and tape measure	Excellent
Mechanical E-stop height	Minimum 2 ft from ground	>2ft	Tape measure from ground to center of button	Excellent
Mechanical E-stop response	Robot stops when button is pressed	Electrical connection checked; full motion stop test pending	Activate during low-speed drive test and record stop distance/time	Pending
Remote E-stop range	Reliable activation up to 100 ft	Full range test pending	Test activation from a distance	Pending
Remote E-stop response	Robot stops when remote E-stop is pressed	Controller input checked; full motion stop test pending	Activate during low-speed drive test record stop distance/time	Pending
Safety indicator light visibility	Visible from all required viewing angles	Light stack installed and visible	Visual inspection from front, rear, left, and right	Excellent

5. Electrical Design

5.1. Power system Overview

The electrical system is split into two galvanically separate buses that share a common ground, as seen in Figure 6. The 24V propulsion bus is built from two 12V 35Ah lead-acid batteries connected in series and dedicated to the drive motors and motor controllers. The 12V accessory bus is a single 12V 22Ah lead-acid battery that powers all sensors, computers, the router, and the AC inverter, feeding a 12V-to-5V DC-DC converter for the Raspberry Pi 5, Picos, encoders, and IMU, and a Kinverch 750W 12V-to-110V AC inverter for any AC accessories. Each bus is protected by its own blade fuse block: 40A fuses sit on the motor branches because the NPC-T74 motors draw high inrush current, while the 12V branches use 2.5-5A fuses sized to each component.

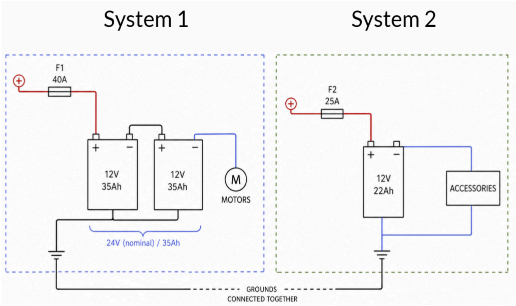

Figure 6. Electrical subsystem architecture

Splitting the buses was a deliberate design decision. During earlier testing, when all batteries were connected in series and the accessory loads shared the propulsion rail, motor inrush current would pull the rail down enough to send the smaller components into safety shutdown. Both the mechanical and wireless E-stops are wired directly into the

24V system so propulsion can be killed as quickly as possible while the computer and sensors remain alive for diagnostics.

5.2. Electronic and Compute Components

The total continuous power consumption of the 24V propulsion bus, including both motors and their Talon SRX controllers, is approximately 388.8 W. The total continuous power consumption of the 12V accessory bus, including LiDAR, GNSS, camera, Raspberry Pi 5, inverter idle, three Picos, two encoders, IMU, Ethernet switch, and router, is approximately 55.6 W. The specific breakdown for each component can be found in Table 4

Table 4. Component Consumption Chart

Electrical Component	**Current Draw (A)**	**Components in System**	**Operating Voltage (V)**	**Power Consumption (W)**
Motors (24V NPC-T74 Motors)	8	2	24	384
Motor Controllers (CTRE Talon SRX)	0.1	2	24	4.8
Total Consumption				**388.8**
TIM561 (2D Lidar)	0.33	1	12	3.96
Atlas Link (GNSS)	0.25	1	12	3
Lucid Camera (TDR0545-CC)	0.3	1	12	3.6
iRasptek Raspberry Pi 5	2.76	1	5*	13.8
Kinverch 750W Power Inverter	1.5	1	12	18
RP2040 Pico w/ CAN Hat (1528-5724-ND)	0.05	3	5	0.75
Encoders (Adafruit AS5600 Magnetic Angle Sensor)	0.0065	2	5	0.065
Yahboom IMU (CMP10A)	0.02	1	5	0.1
e-Link Ethernet Switch (LNK-IMC104GP-SFP)	0.4	1	12	4.8
Router (GL-AXT1800)	1.5	1	5	7.5
Total Consumption				**55.575**

Run time was estimated using a 50% depth-of-discharge cap (the standard limit for lead-acid chemistry to avoid permanent damage) and bus efficiencies of 90% for the high-load propulsion bus and 95% for the accessory bus. Using Equation 1, and the above assumptions the propulsion bus runs for approximately 1.02 hours and the accessory bus for approximately 2.14 hours, both of which exceed the IGVC course time.

$$T_r = \frac{V_T \times C \times DoD \times E}{W_T} \qquad \textbf{(1)}$$

Recharging is performed with a NOCO Genius5 6V/12V 5A smart charger. Because the charger is designed for 12V batteries, the two 35Ah cells of the propulsion bus are charged separately. Using the same 50% DoD and bus-efficiency assumptions and Equation 2, the propulsion bus recharges in approximately 3.68 hours and the accessory bus in approximately 2.44 hours.

$$T_c = \frac{C \times DoD}{5 \times E} \qquad \textbf{(2)}$$

The full breakdown for the entire electrical system can be found in Figure 7. All of these components are rigidly connected to the frame of the robot. This ensures that the connections remain secure while the robot is in motion.

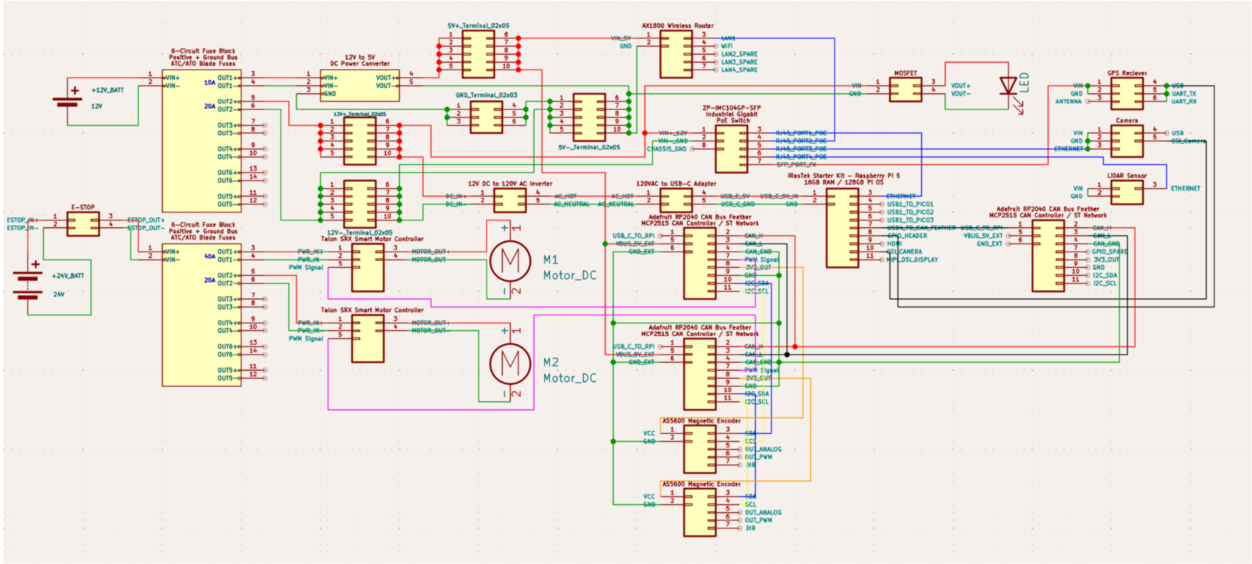

Figure 7. Full Electrical Schematic

5.3. Battery and Encoder Calibration Results

After the original lead-acid batteries were found, through multimeter testing, to be unable to charge above approximately 11.5V (well below the 12.1V floor that corresponds to a healthy lead-acid 50% DoD), the team replaced them. The replacement batteries measured 13.4V and 13.3V on the propulsion bus (26.7V series, target 24V) and 13.2V on the accessory bus (target 12V), well within healthy operating range. Testing for both the original batteries as well as the new batteries were done using the same methodology - multimeter testing.

Encoder calibration was performed by mounting each AS5600 to its drive wheel, powering the sensor from VIN rather than 3Vo (testing showed VIN was required for stable readings), and running each wheel through ten full rotations while the Pico counted ticks. The left motor produced 243,040 ticks over ten rotations (24,304 ticks/rotation, 67.5 resolution, 0.13% error) and the right motor produced 829,600 ticks (82,960 ticks/rotation, 230.4 resolution, 0.11% error). Both errors are below 0.15%, which is acceptable for closed-loop velocity control. The Talon SRX controllers were verified through their built-in indicator LEDs (yellow when stopped, green forward, red reverse, flashing red on signal loss) and the Picos through the solid green LED that indicates a running program.

6. Software System

6.1. Perception and Localization

Perception runs on the laptop as a set of ROS 2 nodes that consume the live camera feed, LiDAR scans, and GNSS fixes. The camera node publishes frames on /sensing/camera; downstream perception subscribes to that stream and segments each frame into five vertical bands. Within each band, a white-pixel count is taken under an HSV filter whose thresholds adjust dynamically to compensate for changing outdoor lighting, allowing the system to maintain reliable white-line detection as the sun moves and shadows fall across the course. Lateral lane position is then read directly from which bands contain white pixels.

Obstacle awareness is provided by a SICK TIM561 2D LiDAR mounted on the front of the robot with a 180° forward field of view. The TIM561 publishes approximately 811 points per scan line at roughly 15 Hz. A 20-point noise threshold suppresses spurious returns and an exclusion zone around the chassis filters out points that strike the robot itself before they can influence path planning. Localization in non-lined sections of the course is provided by the Atlas Link GNSS receiver, which supplies the latitude and longitude used for waypoint navigation.

6.2. Mapping and Navigation

Rather than fuse the three modalities into a unified occupancy or cost map, the driving logic uses the live sensor values directly as a lightweight feature set. The camera band white-pixel distribution, the latest LiDAR range array, and the most recent GNSS fix are each consumed from their dedicated topics on every control cycle, so the internal representation is always current without an explicit map-update step. Lane following analyzes which of the five camera bands contain white pixels: detections in the outermost bands trigger a gentle steering correction toward center, while detections in the inner bands indicate the robot is close to crossing and trigger a sharper correction. When the LiDAR reports an obstacle within the critical range, the robot reduces velocity and steers left until either the obstacle is no longer in the forward path or a white boundary line is reacquired in the camera, at which point lane following resumes.

In sections of the course where boundary lines are absent, the robot transitions to GNSS-based waypoint navigation. Predefined waypoints are loaded as latitude/longitude pairs before the run, and the robot computes heading corrections by adjusting linear velocity along the longitudinal axis and angular velocity along the lateral axis until the next waypoint is reached, after which lane following is re-engaged when white lines come back into view.

6.3. Teleoperation

Teleoperation is provided through an RC controller paired to an iBus receiver on the vehicle. The controller transmits velocity and steering setpoints, which are arbitrated against autonomous commands before being forwarded over CAN to the Talon SRX motor controllers. The same controller carries a clearly labeled wireless E-stop button that has the highest arbitration priority, ensuring that operator intervention always overrides autonomy.

6.4. Drive-by-Wire Control

Vehicle motion is generated as ROS 2 Twist messages on the laptop. Twist commands are translated into per-wheel velocity setpoints, sent over Ethernet to the Raspberry Pi 5, and bridged onto the CAN bus by the Pi's CAN-HAT-equipped Pico. Receiver Picos decodes the unique CAN ID of each message to determine which actuator to drive and by how much. Feedback from the AS5600 encoders and the IMU travels back through the same CAN bus to the Pi and out to the laptop, which closes the loop by adjusting commanded velocity over time. The cycle continues until either an E-stop is triggered or the course is completed.

6.5. Object Detection and Classification

In addition to the geometric perception pipeline, a YOLO v11 detector runs on the laptop and consumes the same camera stream to identify potholes, which are not reliably distinguishable by HSV segmentation alone. The model was trained on labeled pothole imagery so that, once a detection clears its confidence threshold, the pothole is treated as an obstacle by the driving logic and triggers the same avoidance maneuver used for physical barriers.

6.6. Scripted Route System

For sections of the course where boundary lines are absent, a scripted route is loaded as an ordered list of latitude/longitude waypoints. The behavior layer feeds the next waypoint to the planner, which commands linear velocity along the longitudinal axis and angular velocity along the lateral axis to drive the heading error to zero. As each waypoint is reached, the next is consumed; when the perception layer acquires a white boundary line, the scripted route hands control back to the camera-based lane following.

7. Analysis of Complete Vehicle

7.1. Construction and Integration

Several lessons emerged during construction and integration. The most important was that integrating subsystems is much harder than building any of them in isolation: even when individual mechanical, electrical, and software components passed unit tests, combining them surfaced unexpected issues with intercomponent communication and physical packaging on the chassis. Component placement and design changes needed to be planned earlier and communicated more aggressively, since several parts had to be moved or redesigned multiple times because of interference with other components or wiring constraints. Wire management, initially neglected, became a major source of debugging friction; once the harnesses were reorganized and properly secured, both troubleshooting time and overall system reliability improved noticeably. The team's clearest takeaway is to integrate and test full subsystems together earlier and more often, so that integration bugs surface while their causes are still localizable.

7.2. Robot Safety and Reliability

Turbo Blue 2.0 has multiple different hardware and software components that allow for overall safety. Per the rulebook, speed has to be limited to a maximum of 5 mph. For Turbo Blue 2.0, this was implemented with a 2 m/s firmware cap. This was selected in order to comply with competition rules and to prevent damage to the robot, specifically the suspension linkages. All of the electrical components were secured using terminal lugs or screw terminals, with any wire junction also being covered with shrink tube. This ensured that all components followed proper SAE guidelines and ensured safety to the team members. Directly connected to the 24V bus is both a mechanical E-Stop (red button on labeled yellow support) and a wireless E-Stop (remote operates up to 100ft) which allow for an immediate stoppage of power to the motors and therefore bring the robot to an immediate stop. Additionally, the robot has an indicator mode to allow the surrounding people to understand the robot's state. The indicator light is solid when the robot is powered on in TeleOp mode and it is flashing when the robot is in autonomous mode. Throughout all the design phases, safety was the highest priority and therefore it was intertwined with each step.

7.3. Robot Control

High-level decisions are made on the laptop, where a set of ROS 2 nodes form the control architecture. Twist commands are forwarded to the onboard Raspberry Pi 5, which uses a Pi Pico equipped with a CAN HAT to broadcast unique-ID CAN messages to receiver Picos. Each receiver Pico decodes the ID to identify the target actuator and the commanded value, and drives the corresponding Talon SRX accordingly. Encoders and IMU feedback travel back along the same CAN structure to the Pi and out to the laptop, which adjusts the next command on the next cycle. This decentralized architecture separates high-level planning from low-level motor control and gives real-time, closed-loop control over both drive motors. The loop runs continuously until the E-stop is pressed or the course is complete.

7.4. Software Development and Version Control

The software stack is distributed across six dedicated GitHub repositories, organized by subsystem to promote modularity, ease of access, and streamlined collaboration. Each repository owns a distinct component of the system, allowing developers to clone, contribute to, or deploy individual parts of the stack independently rather than working in a monolithic codebase. This structure simplifies version control and continuous integration: updates to a single component can be developed, reviewed, and tested in isolation before being integrated into the broader subsystems. Rather than relying on full simulation environments such as Gazebo, the team validates each component through unit testing, which keeps the testing loop fast, simple to reproduce, and tightly coupled to real hardware.

7.5. Physical Testing and Performance

Physical testing was done on each individual component and then successively for subsystems and thereafter systems. Table 5 indicates the different tests that were performed and the outcome of them.

Table 5. Physical Tests and their Statuses

Physical Test		Status
Controls	Differential Steering	Functional, Tested
	Raspberry Pi 5	Functional, Tested
	GL-AXT1800 Router	Functional, Tested
	e-Link Ethernet Switch	Functional, Tested
	FLYSKY Teleop Controller	Functional, Tested
	CTRE Talon SRX Motor Controllers	Functional, Tested
	RP2040 Pico w/ CAN Hat	Functional, Tested
	CAN Bus Network	Functional, Tested
Power	22 Ah 12V Batteries	Functional, Tested
	12 Ah 12V Batteries	Functional, Tested
	Kinverch 750W Power Inverter	Functional, Tested
	12V - 5V DC-DC Buck Converter	Functional, Tested
Sensors	Lucid Camera (TDR0545-CC)	Functional, Tested, Available in ROS Network
	TIM561 (2D Lidar)	Functional, Tested, Available in ROS Network
	Atlas Link (GNSS)	Functional, Tested, Available in ROS Network
	Yahboom IMU (CMP10A)	Functional, Tested, Available in ROS Network
	AS5600 Magnetic Angle Encoders	Functional, Tested, Calibrated
Safety	Mechanical E-STOP	Functional, Tested
	Remote E-STOP	Functional, Tested
	Safety Light	Functional, Tested

Beyond the bench-level tests captured in Table 4, fully integrated software-and-hardware testing on the IGVC course is still in progress. Key performance indicators are monitored in real time during testing using PlotJuggler, which allows the team to inspect the publish rates of each sensor (LiDAR, camera, IMU) as well as inter-node message latency, confirming that data is flowing at the frequencies the perception and control loops expect and surfacing dropouts when they occur. Target frequencies are chosen based on the minimum rate the perception and control pipeline needs to react to the course quickly enough to remain competitive.

8. Cyber Security Analysis

Modern robotic and human-driven vehicles are network-connected, contain large amounts of software from multiple vendors, and are an inviting target for cyber-attacks. On Turbo Blue 2.0, the primary attack surfaces are the radio control link, the CAN bus, and the ROS 2 communication layer. A malicious actor who identifies and spoofs the radio control frequency could effectively hijack or disable the robot, rendering it unresponsive to legitimate commands. Similarly, physically tapping into the CAN bus and injecting or manipulating messages on the wiring harness could produce unpredictable and uncontrolled vehicle behavior, posing a safety risk to anyone in the operating area. ROS 2 uses DDS middleware which by default broadcasts topic data openly over the local network, meaning any device on the same network could potentially subscribe to sensitive sensor topics or publish malicious commands to control topics — compromising the integrity of the perception and driving logic.

To mitigate these risks today, the vehicle employs iBus arbitration to manage controller priority, ensuring that only one authorized controller can command the vehicle at a time and preventing rogue or conflicting CAN messages from taking effect. The E-stop system is assigned the highest priority in the arbitration hierarchy, meaning it can override any other controller input regardless of the source — providing a reliable last line of defense against unintended or malicious commands.

Three vulnerabilities to harden before series production Radio Control Spoofing. The RC radio link operates on an open frequency that could be intercepted or spoofed by a malicious actor with the right equipment, potentially allowing unauthorized control of the vehicle. Before series production, this should be mitigated by implementing encryption between the handheld controller and receiver, ensuring that only paired and authenticated transmitters can issue commands to the vehicle.

CAN Bus Exposure. The CAN bus lacks native authentication or encryption, so any device physically connected to the network can inject arbitrary messages. If CAN IDs are known or discoverable, an attacker could craft and inject messages that target specific motor controllers, causing unintended and potentially dangerous behavior. In a production vehicle, this should be addressed by implementing CAN bus message authentication, obfuscating or masking CAN IDs to prevent unauthorized nodes from identifying and targeting safety-critical messages, and physically securing the wiring harness behind tamper-evident enclosures.

ROS 2 / DDS Open Discovery. ROS 2's DDS middleware broadcasts topics openly over the local network by default, so any device on the same network could subscribe to sensor topics or publish malicious data to control topics. Before production deployment, this should be hardened by enabling the DDS Security plugins (authentication, encryption, and access control), isolating the ROS 2 network on a dedicated VLAN, and restricting which nodes are permitted to publish to safety-critical topics.