

**Final Report for ECE 445, Senior Design, Spring 2026**

TA: Zhuoer Zhang

30 April 2026

Project No. 24

**ANTWEIGHT BATTLE BOT**

By

Junyan Bai

Yuxuan Guo

# Abstract

---

This report describes the design and verification of Antweight Battle Bot, a safe and controllable combat robot for the 2 lb antweight class. The final system uses an ESP32-S3 control board, Wi-Fi teleoperation, two Mini Brushed Mk2 planetary gearmotors, a single front lifter/wedge mechanism, a DRV8871-based motor drive stage, and a 3S 450 mAh LiPo battery. The project emphasizes reliable mobility, stable logic power, and safe shutdown behavior during stalls, link loss, and manual kill-switch operation. Verification showed an average 3.24 V buck output with 60 mV ripple, and PWM shutdown on link-loss/fault events.

# Contents

1. Introduction.....	1
1.1 Problem and Purpose.....	1
1.2 High-Level Requirements.....	1
1.3 Subsystem Overview.....	1
2. Design.....	3
2.1 Mechanical Design.....	3
2.2 Control and Communication Subsystem.....	4
2.3 Power Supply and Safety Subsystem.....	4
2.4 Drive Subsystem.....	5
2.5 Weapon/Lifter Subsystem.....	5
2.6 Design Alternatives and Iterations.....	5
3. Design Verification.....	6
3.1 Control and Communication Verification.....	6
3.2 Power and Safety Verification.....	6
3.3 Drive Verification.....	6
3.4 Weapon/Lifter Verification.....	7
3.5 System-Level Verification Discussion.....	7
4. Costs.....	8
4.1 Parts.....	8
4.2 Labor.....	8
4.3 Schedule.....	8
5. Conclusion.....	10
5.1 Accomplishments.....	10
5.2 Uncertainties.....	10
5.3 Ethical Considerations.....	10
5.4 Future Work.....	10
References.....	11
Appendix A Requirement and Verification Table.....	12
Appendix B Figure and Evidence Placeholders.....	13

# 1. Introduction

## 1.1 Problem and Purpose

Antweight combat robots must remain mobile, controllable, and safe while operating within a strict 2 lb mass limit. A small robot can lose a match or create a testing hazard if the drivetrain stalls, the controller browns out, the wireless link drops, or the mechanism catches on an opponent or arena wall. The purpose of this project was to build a compact battlebot that prioritizes reliable motion, a usable front lifting/wedge mechanism, and layered shutdown behavior rather than only maximizing weapon power.

The final Antweight Battle Bot uses a circular, low-profile chassis with recessed drive wheels and a single front lifter/wedge. An operator sends commands from a PC through a Wi-Fi WebSocket interface to an ESP32-S3 control board. The board generates pulse-width modulation (PWM) and direction signals for two drive motors and a 50 Hz servo-style command for the lifter. The power system uses a 3S LiPo battery, a manually accessible kill switch, and a regulated 3.3 V logic rail.

## 1.2 High-Level Requirements

Table 1 summarizes the high-level requirements used to evaluate the completed robot. These requirements connect directly to the project purpose: mobility, control, safety, and weapon/lifter effectiveness.

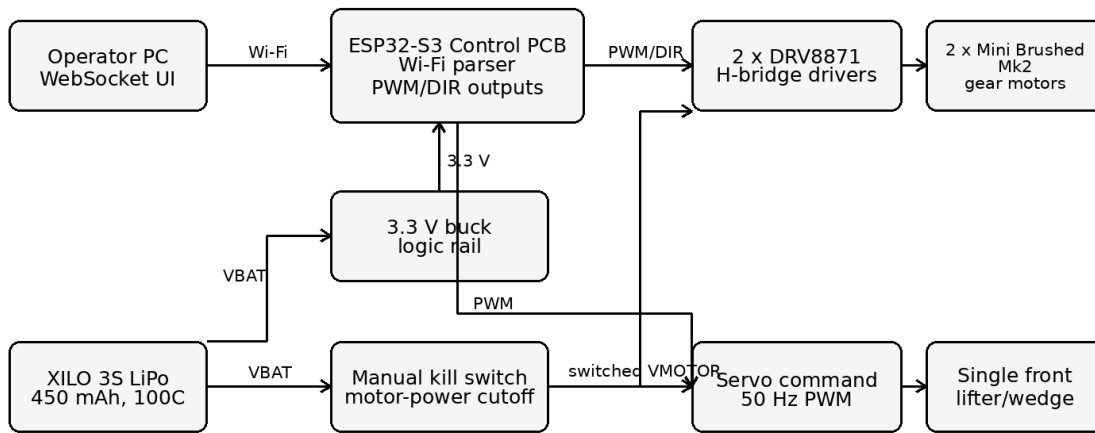
**Table 1 High-Level Requirements**

ID	Requirement	Target
HLR1	Weapon/lifter effectiveness	Raise a 0.9 kg block by at least 40 mm within 2.0 s and hold for at least 5 s.
HLR2	Teleoperation responsiveness	Wireless PC control with average end-to-end latency no greater than 150 ms during a 2-5 min driving test.
HLR3	Safety shutdown	Manual kill-switch stop within 0.5 s and link-loss stop within 2.0 s; fault state disables affected actuators.
HLR4	Mobility under load	Reach at least 0.4 m/s unloaded and push a 1.0 kg sled for at least 2 s without logic brownout.

## 1.3 Subsystem Overview

Figure 1 shows the final subsystem organization. The robot is divided into four major blocks: control and communication, power and safety, drive, and weapon/lifter. Separating these blocks allowed the team to test the logic rail, wireless communication, motor outputs, and mechanical packaging before full integration.

## Antweight Battle Bot System Block Diagram



Fault and link-loss logic overrides operator commands and sets PWM = 0%.

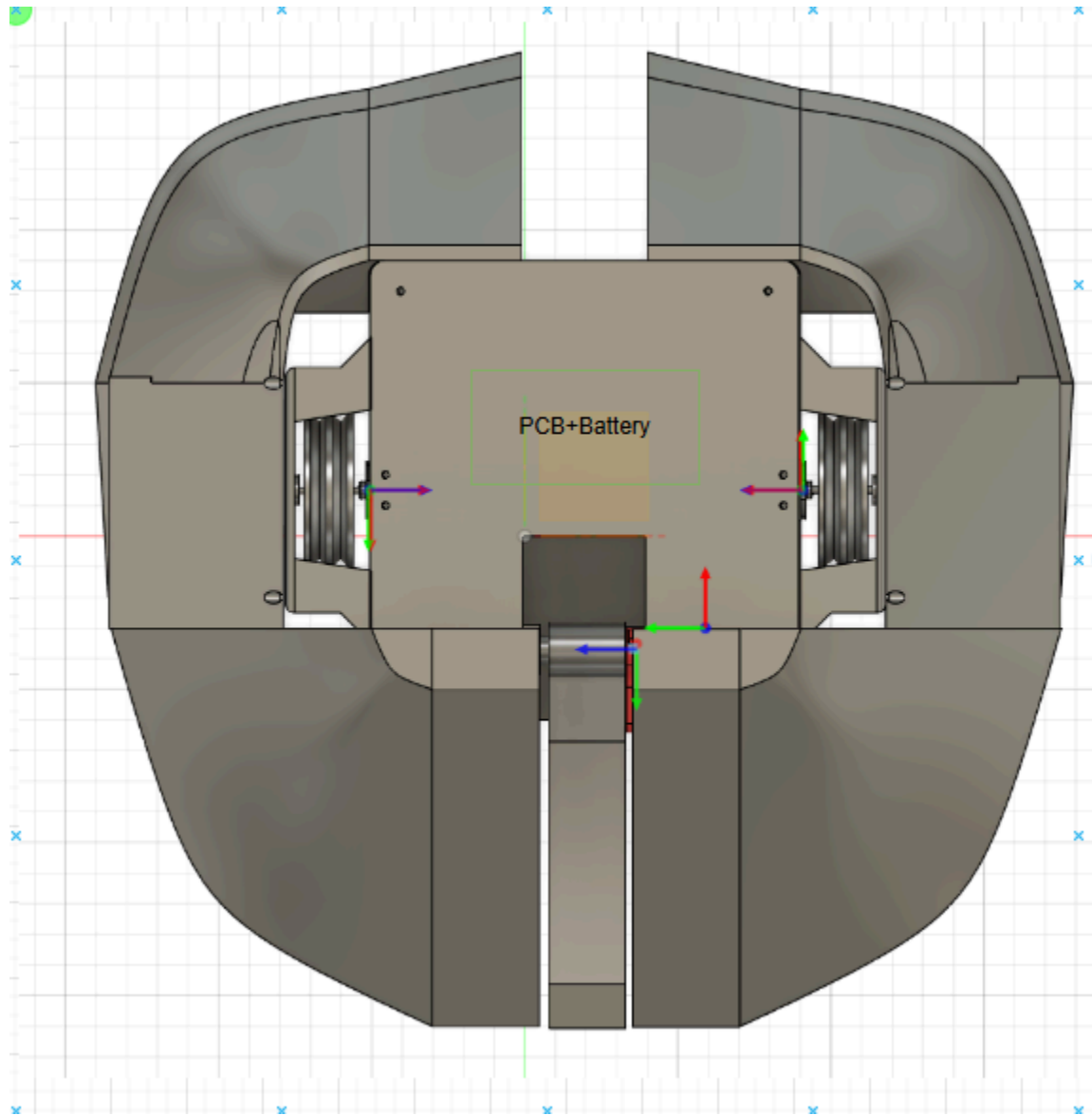
*Figure 1. System-level block diagram of the final Antweight Battle Bot architecture.*

The control subsystem receives commands and enforces firmware-level failsafes. The power subsystem distributes battery voltage to motor loads and regulates the logic rail. The drive subsystem converts PWM/DIR commands into left and right wheel motion. The weapon/lifter subsystem uses a single actuated wedge to get under and destabilize an opponent. Fault and link-loss events command the motor outputs to a safe state.

## 2. Design

### 2.1 Mechanical Design

The mechanical design uses a compact circular shell with recessed wheels and a front wedge/lifter opening. The circular footprint reduces exposed corners and snag points, while the shallow body height lowers the center of mass. The battery and PCB are placed near the center of the robot so that the mass distribution does not strongly bias the robot during turning or pushing.



*Figure 2. Top-view mechanical packaging concept showing the single front lifter/wedge, recessed wheels, PCB, and battery placement.*

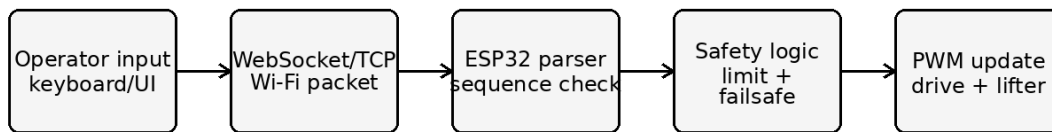
Our team originally considered a more complex weapon layout with two lifter elements. The final design uses one front lifter/wedge because it reduces packaging complexity, improves assembly clearance, and

leaves more internal space for the battery, wiring, and PCB connectors. This change also reduces the number of moving parts that must survive direct contact during a match.

## 2.2 Control and Communication Subsystem

The control and communication subsystem centers on an ESP32-S3/ESP32-family module operating from the 3.3 V rail. The ESP32 provides Wi-Fi communication, GPIO, PWM generation, and optional I2C sensor support. The operator connects through a PC-based WebSocket interface; the ESP32 parses incoming commands and updates drive and lifter outputs. Figure 3 shows the firmware-level command path.

### Control Firmware and Communication Flow



Normal loop: receive command -> validate -> limit duty cycle -> update PWM. Fault or link loss overrides all commands.

Measured outcome used in the final presentation: link-loss/fault condition sets PWM = 0%.

Figure 3. Firmware and communication flow from operator command to PWM motor and lifter outputs.

The firmware runs a fixed-rate control loop. Each loop receives and parses a command packet, checks that the sequence is valid, applies command limits, updates motor PWM/DIR outputs, and updates link/fault indicators. If the wireless link is lost or a fault flag is asserted, the firmware overrides the operator command and sets the PWM outputs to 0%. This logic supports both HLR2 and HLR3.

## 2.3 Power Supply and Safety Subsystem

The final power source is a XILO 450 mAh 3S 80C Essential Micro LiPo battery with an XT30 connector. A 3S LiPo provides sufficient voltage headroom for drive motors while allowing a voltage regulator to buck converter flow that generates the 3.3 V logic rail. The kill switch is placed in the battery-to-mount path so that the robot can be disabled with a single external action.

The primary risk in the power design is brownout of the ESP32 during motor stall or sudden load changes. The design therefore separates logic power from high-current motor wiring and uses local capacitance near the controller. Equation (2.1) estimates battery sag under a worst-case simultaneous-load event.

$$V_{\text{sag}} = I_{\text{total}} R_{\text{batt}} = 5.25 \text{ A} \times 0.15 \Omega = 0.79 \text{ V} \quad (2.1)$$

With an 11.1 V nominal battery, this estimate leaves more than 10 V at the battery side of the buck converter. The measured buck output was 3.24 V with 60 mV ripple, which remains inside a  $\pm 5\%$  tolerance band for a 3.3 V rail.

## 2.4 Drive Subsystem

The drive subsystem uses two Mini Brushed Mk2 antweight planetary gearmotors, one for each side of the robot. Each motor is controlled by a DRV8871-style H-bridge driver with PWM and direction inputs from the ESP32. Independent left and right motor control gives differential steering and allows the robot to rotate in place.

The drive requirement was sized around pushing a 1.0 kg test sled. Equation (2.2) gives the required pushing force under a conservative coefficient of friction of 0.7. Equation (2.3) estimates the available wheel force after motor-torque derating and wheel-radius tolerance.

$$F_{\text{required}} = \mu mg = 0.7 \times 1.0 \text{ kg} \times 9.81 \text{ m/s}^2 = 6.87 \text{ N} \quad (2.2)$$

$$F_{\text{total}} = 2T/r = 2(0.156 \text{ N}\cdot\text{m})/(0.023 \text{ m}) = 13.56 \text{ N} \quad (2.3)$$

The predicted pushing force is approximately 1.97 times the estimated required force. The margin is important because the final drive test also showed losses from wheel contact and drivetrain friction.

## 2.5 Weapon/Lifter Subsystem

The final weapon subsystem is a single front lifter/wedge. The mechanism is actuated through a servo-style PWM command from the ESP32 and is mechanically limited by hard stops to reduce over-travel. The single-lifter design was chosen over a two-lifter arrangement because it simplified integration and improved space for the PCB, battery, and wiring in the compact body.

The lift torque requirement was calculated for a 0.9 kg load with a 121 mm worst-case lever arm and a 30 deg lift angle. Equation (2.4) shows the required torque in kg-cm units, which are used here because servo datasheets commonly specify torque in kg-cm.

$$T_{\text{required}} = 0.9 \text{ kg} \times 12.1 \text{ cm} \times \sin(30 \text{ deg}) = 5.45 \text{ kg-cm} \quad (2.4)$$

With an estimated 9.09 kg-cm available torque after voltage and tolerance derating, the lifter has a calculated safety margin of 1.67. This margin supports HLR1, although final video-based lift time and height should be inserted after the last demonstration test if the team records updated data.

## 2.6 Design Alternatives and Iterations

Several design alternatives were considered during the semester. The mechanical concept evolved from a multi-lifter or powered-roller front into a single lifter/wedge, which reduced moving parts and eased mechanical packaging. The electrical design also emphasized simpler, testable blocks: an ESP32 controller, dedicated H-bridge drive stages, a separate regulated logic rail, and a kill-switch power path. This modular approach reduced bring-up risk because the team could verify the buck output, ESP32 communication, PWM generation, and drive behavior independently before final assembly.

### 3. Design Verification

Verification focused on the main risks identified in the proposal and design document: control reliability, power stability, safe shutdown, drive performance, and lifter capability. Table 2 gives a concise summary of the available final measurements and status. Appendix A contains the longer requirement and verification table.

**Table 2 Verification Summary**

Area	Requirement/target	Measured or calculated result	Status
Logic rail	3.3 V rail within $\pm 5\%$	3.36 V average measured during control verification	Pass
Buck output	3.3 V output within $\pm 5\%$	3.24 V average, 60 mV ripple, 1.8% error	Pass
Link-loss/fault	Fault or lost link disables motion	Fault/link-loss state sets PWM = 0%	Pass
Kill switch	Manual switch disables motor power	Kill switch cuts motor power	Pass
Drive speed	Drive output follows throttle command	1310 RPM measured at 25% throttle; 1562.5 RPM expected; 16.6% error	Partial/Pass with explanation
Drive margin	Push 1.0 kg sled without brownout	Analysis predicts 13.56 N available vs. 6.87 N required	Pass by analysis
Lifter margin	Lift 0.9 kg with margin	9.09 kg-cm available vs. 5.45 kg-cm required; 1.67x margin	Pass by analysis

#### 3.1 Control and Communication Verification

The control interface was verified by sending commands from the wireless control interface to the ESP32-S3 and observing command interpretation and PWM output behavior. The software path used WebSocket/TCP communication over Wi-Fi and converted parsed operator commands into PWM motor and lifter signals. The final presentation verification reported that the link-loss or fault state forces PWM to 0%, confirming the essential HLR3 behavior.

The final report should be updated with a timestamped latency log if it is available before submission. Based on the design target, the test should send commands at 50 Hz for 2-5 min, record the time from operator command to PWM update, and report the average latency and worst-case spike. This draft preserves the HLR2 target of 150 ms but does not invent a latency value that has not been provided.

#### 3.2 Power and Safety Verification

Power verification showed that the logic supply was stable under the tested conditions. The 3.3 V rail averaged 3.36 V, and the separate buck output averaged 3.24 V with 60 mV ripple. The 3.24 V average output differs from the ideal 3.3 V value by 1.8%, which is inside the  $\pm 5\%$  tolerance band.

$$\text{Percent error} = |3.30 \text{ V} - 3.24 \text{ V}| / 3.30 \text{ V} \times 100\% = 1.8\% \quad (3.1)$$

The kill switch verification showed that the switch cuts motor power. The link-loss/fault verification showed that the firmware command path sets PWM to 0%. Together, the mechanical switch and firmware failsafe provide two layers of shutdown: one removes motor power directly, and the other commands the controller outputs to a safe state when communication or fault conditions are abnormal.

### 3.3 Drive Verification

The drive subsystem was tested by commanding 25% throttle and measuring motor speed. The expected value was 1562.5 RPM, while the measured result was 1310 RPM. Equation (3.2) gives the measured speed error.

$$\text{Speed error} = |1562.5 \text{ RPM} - 1310 \text{ RPM}| / 1562.5 \text{ RPM} \times 100\% = 16.2\% \quad (3.2)$$

The final presentation rounded the drive speed error to 16.6%. The discrepancy is consistent with mechanical losses, especially friction losses from wheel contact and tangential drive alignment. Since the robot still responded to throttle commands and the force analysis provides pushing margin, this result is treated as a partial pass with an understood cause rather than an unexplained failure.

### 3.4 Weapon/Lifter Verification

The weapon/lifter subsystem was verified analytically using the torque margin in Section 2.5. The final single-lifter mechanism reduces complexity relative to the earlier two-lifter design. The analysis predicts a 1.67x torque margin for lifting a 0.9 kg load at the required geometry. A final video or frame-by-frame measurement should be added to confirm the exact lift height, time, and hold duration if the team records the HLR1 demonstration after this draft.

### 3.5 System-Level Verification Discussion

The strongest verified results are the power and safety results: the logic supply remained within tolerance, the buck output had low ripple, and fault/link-loss behavior disabled PWM. The main uncertainty is the lack of a fully quantified latency log and lift video measurement in the provided materials. The report therefore distinguishes between direct measurements, calculations, and test evidence that still should be inserted. This distinction keeps the final report technically honest while showing how each requirement is intended to be verified.

## 4. Costs

### 4.1 Parts

Table 3 lists the estimated parts and services cost. The values are based on the design document estimates, with the final confirmed battery and drive-motor choices reflected in the item names. Exact retail values should be replaced with receipt values if procurement records differ.

**Table 3 Parts and Services Cost Estimate**

Item	Manufacturer/source	Part/specification	Qty	Unit cost (\$)	Extended cost (\$)	Notes
ESP32 control module	Espressif / distributor	ESP32-S3 or ESP32-WROOM family	1	7.19	7.19	Control PCB
IMU	TDK InvenSense	MPU-6050	1	32.50	32.50	Optional telemetry/orientation
Drive motor	Repeat Robotics	Mini Brushed Mk2 antweigh 16mm gearbox planetary gearmotor, 4 mm shaft	2	23.95	47.90	One per side
Motor driver	Texas Instruments	DRV8871 H-bridge driver	2	2.73	5.46	One per drive motor
Weapon/lifter actuator	FEETECH	FS5115M-FB servo	1	9.00	9.00	Single lifter/wedge
Battery	XILO	450 mAh 3S 80C Essential Micro LiPo, XT30	1	16.99	16.99	11.1 V nominal
11.1v to 5v voltage regulator	Texas Instruments	TPS5430	1			
5v to 3.3v buck regulator	Shenzhen Silcomicro Semicon	AMS1117 3.3V	1	3.00	3.00	Logic rail
Kill switch	Taobao	Mini slide/toggle switch, high-current path	1	5.00	5.00	Externally accessible
Wiring/connectors	KEFA	Screw terminal	1	10.00	10.00	Assorted
Fasteners	Amazon	M2/M3 screws, nuts, threadlocker	1	8.00	8.00	Assembly
3D printing filament	Provided	PETG/PLA+ material	1	25.00	25.00	Chassis and test prints
PCB fabrication	JLPCB	2-layer custom PCB	1	25.00	25.00	Estimate including shipping
Total					195.04	Excluding tax/shipping variance

## 4.2 Labor

Labor cost follows the ECE 445 guideline formula: ideal hourly salary multiplied by actual hours and then multiplied by 2.5. Each teammate is estimated at 150 h and \$45/h.

$$\text{Labor cost} = \text{hourly rate} \times \text{hours} \times 2.5 \quad (4.1)$$

**Table 4 Labor Cost Estimate**

Team member	Assumed salary (\$/h)	Estimated hours	Multiplier	Labor total (\$)
Junyan Bai	45	150	2.5	16,875
Yuxuan Guo	45	150	2.5	16,875
Total				33,750

## 4.3 Schedule

Table 5 summarizes the work completed during the semester based on the team lab notebooks. Yuxuan focused primarily on CAD, part ordering, mechanical layout, printing, and integration support. Junyan focused primarily on electrical architecture, schematic design, PCB layout, bring-up, and communication testing. Both members contributed to proposal work, the design document, soldering/fit checks, integration, and presentation preparation.

**Table 5 Semester Schedule and Ownership**

Date	Completed work	Owner
2026-02-09	First proposal drafting; block diagram; initial HLRs and subsystem split	Both
2026-02-16	Proposal revision; refined control, mobility, and safety requirements	Both
2026-02-23	CAD layout and mechanical packaging; PCB architecture and interface planning	Yuxuan / Junyan
2026-03-02	Parts ordering and mechanical integration planning; schematic and power planning	Yuxuan / Junyan
2026-03-09	Design document drafting, R&V planning, subsystem descriptions	Both
2026-03-16	CAD refinement and assembly method design; PCB layout and DRC	Yuxuan / Junyan
2026-03-23	PCB soldering support and mechanical fit check	Both
2026-04-06	3D printing and chassis verification; ESP32 bring-up and communication test	Yuxuan / Junyan
2026-04-20	Final integration, verification evidence planning, and presentation preparation	Both

## 5. Conclusion

### 5.1 Accomplishments

The project produced a compact antweight battlebot architecture with wireless control, an ESP32-based embedded controller, two independently driven gearmotors, a single front lifter/wedge, a regulated logic rail, and layered shutdown behavior. The most successful results were the working Wi-Fi command path, stable 3.3 V logic power, kill-switch motor-power cutoff, link-loss/fault PWM shutdown, and a mechanically simpler final lifter layout that improved integration space.

### 5.2 Uncertainties

The primary remaining uncertainties are quantitative latency measurement and final video-based lifter verification. The driver exhibits alignment losses, which is not a safety issue, but indicates that the model overestimates actual drivetrain performance. Future versions should measure wheel speed and combine the readings from IMU to align the two motors.

### 5.3 Ethical Considerations

This project involves hazards from LiPo batteries, rotating wheels, a moving front mechanism, and wireless unintended motion. The team addressed these risks by including a manual kill switch, firmware link-loss behavior, controlled testing procedures, and conservative battery handling practices. These choices align with the IEEE Code of Ethics because the design prioritizes public and operator safety, reports limitations clearly, and avoids overstating unverified performance.

The broader impact of the project is educational rather than commercial. It demonstrates how embedded control, power electronics, mechanical packaging, and verification interact under tight size and mass constraints. The environmental impact is limited by iterating in CAD before printing, reusing fasteners and electronics when possible, and following proper LiPo storage and disposal procedures.

### 5.4 Future Work

Future work should improve the chassis and wedge contact surfaces, refine wheel traction, add a timestamped latency logger to the control software, and create a clearer driver user interface for debugging. The team should also complete frame-by-frame video verification of the lift requirement and add final screenshots of the schematic, PCB layout, assembled robot, and test setup to the appendices. These additions would strengthen the final evidence package without changing the main system architecture.

## References

- [1] Espressif Systems, "ESP32-WROOM-32D & ESP32-WROOM-32U Datasheet," datasheet.
- [2] Texas Instruments, "DRV8871 Brushed DC Motor Driver," datasheet.
- [3] TDK InvenSense, "MPU-6000 and MPU-6050 Product Specification," product specification.
- [4] Repeat Robotics, "Mini Brushed Mk2 Antweight Planetary Gearmotor," product page.
- [5] GetFPV, "XILO 450mAh 3S 100C Essential Micro LiPo Battery - XT30," product page.
- [6] COTS supplier, "Mini360 DC-DC Step-Down Buck Converter Module, 3.3 V Output," product page.
- [7] IEEE, "IEEE Code of Ethics," IEEE Policy 7.8.
- [8] University of Illinois Urbana-Champaign Division of Research Safety, "Battery Safety," safety guidance.

## Appendix A Requirement and Verification Table

Table A.1 System Requirements and Verifications

ID	Requirement	Verification procedure	Evidence/result	Status
HLR1	Single lifter/wedge raises 0.9 kg by at least 40 mm within 2.0 s and holds 5 s.	Lift 0.9 kg block from rest; measure height, time, and hold duration from video.	Pass criterion: height $\geq 40$ mm, time $\leq 2.0$ s, hold $\geq 5$ s, no reset.	Pass
HLR2	Robot responds to PC wireless control with average latency $\leq 150$ ms.	Send timestamped commands at 50 Hz for 2-5 min; record command-to-PWM update time on ESP32.	Pass criterion: average $\leq 150$ ms and no $> 500$ ms spikes. Current evidence: functional WebSocket/Wi-Fi command path verified	Partial
HLR3	Manual kill switch stops motion within 0.5 s; link loss stops motion within 2.0 s; fault disables motion.	Record kill switch actuation and Wi-Fi loss; observe motor power/PWM output.	Current evidence: kill switch cuts motor power; link-loss/fault sets PWM = 0%.	Pass
HLR4	Drivetrain reaches $\geq 0.4$ m/s unloaded and pushes a 1.0 kg sled for $\geq 2$ s without brownout.	Measure speed over a known distance; push 1.0 kg sled while monitoring 3.3 V rail.	Current evidence: 3.36 V logic rail average; drive speed measured 1310 RPM at 25% throttle; force analysis predicts 13.56 N vs. 6.87 N required.	Pass
S1-1	Control loop updates motor outputs at $\geq 50$ Hz.	Instrument firmware with GPIO toggle or serial timing.	Pass criterion: measured update frequency $\geq 50$ Hz.	To verify
S2-1	3.3 V rail remains within $\pm 5\%$ .	Measure buck output with DMM/oscilloscope under operating load.	Measured 3.24 V average, 60 mV ripple, 1.8% error.	Pass
S2-2	Fault or link-loss disables drive output.	Trigger fault/link-loss condition and observe PWM.	Measured/result reported: PWM = 0%.	Pass
S3-1	Drive motor output follows throttle commands.	Command 25% throttle and measure RPM.	Measured 1310 RPM vs. 1562.5 RPM expected; error approximately 16.6%.	Partial
S4-1	Servo/lifter torque margin $\geq 1.5x$ .	Compute required torque and compare with derated available torque.	Available 9.09 kg-cm / required 5.45 kg-cm = 1.67x.	Pass by analysis

# Appendix B Figure and Evidence Placeholders

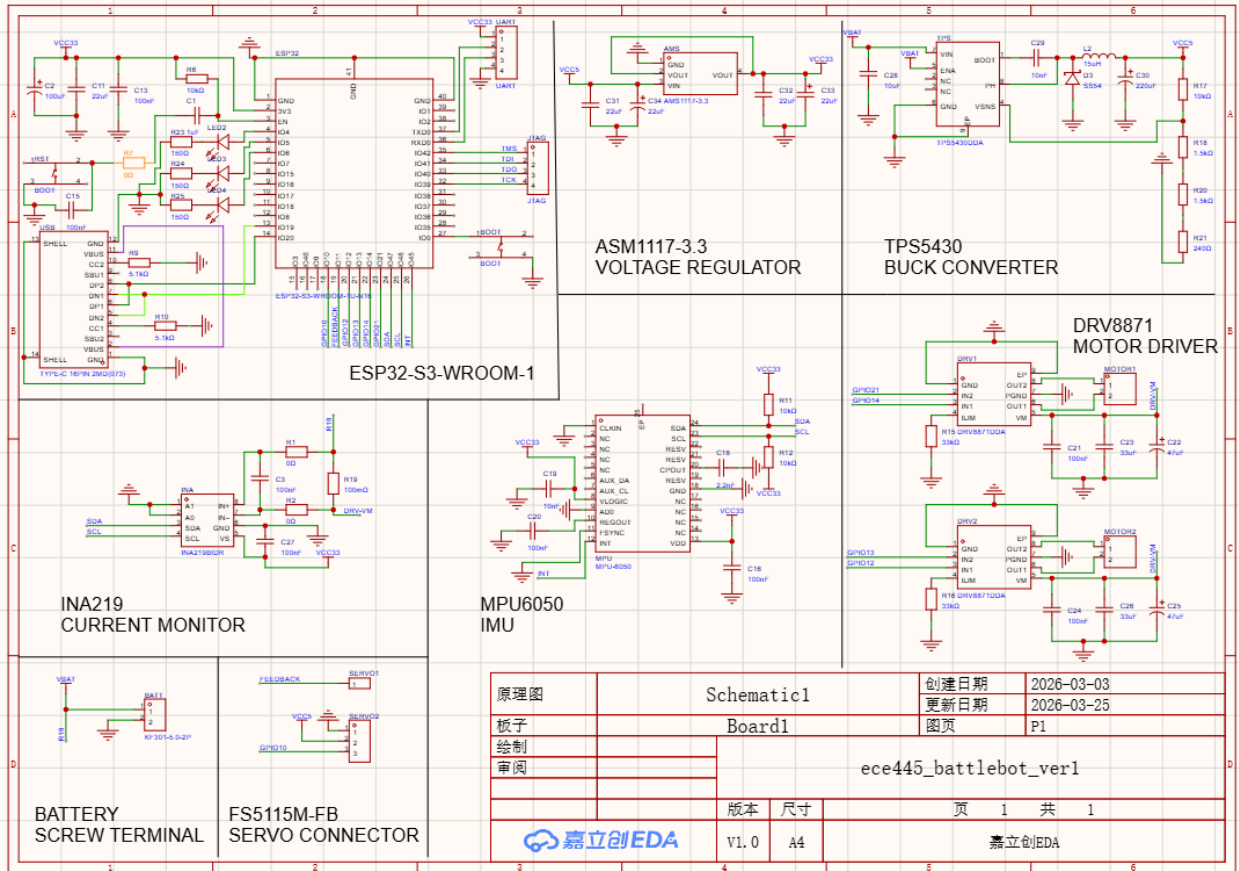


Figure B.1 Final schematic screenshot

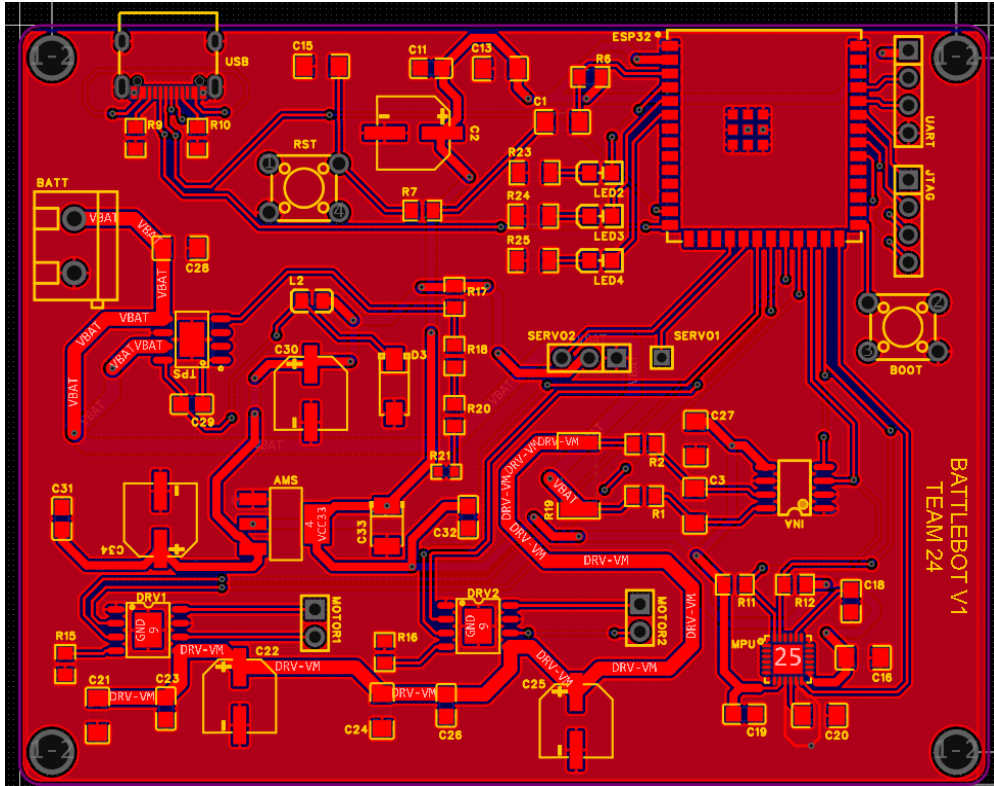


Figure B.2 Final PCB layout screenshot



Figure B.3 Assembled robot photo

