ECE445 FINAL REPORT

By

Mig Umnakkittikul

Yanhao Yang

Yunhan Tian

Final Report for ECE 445, Senior Design, Fall 2025

TA: Gayatri Chandran

December 9, 2025

Project Team 9

Abstract

This report presents the design, development, and analysis of an antweight battlebot for ECE 445 Senior Design, featuring a front-hinged lifting wedge shovel weapon actuated by a high-torque servo. The 3D-printed chassis uses PLA for all parts, including the weapon, ensuring durability under combat stresses while meeting the <2 lb weight limit and approved materials rules. Controlled wirelessly via Bluetooth Low Energy (BLE) with an ESP32-S3 microcontroller, the bot employs two DRV8871 H-bridge drivers handling 12V directly for the dual DC drivetrain motors, paired with two buck converters supplying 3.3V for the ESP32 and 5V for the servo. Powered by a 3S 450 mAh LiPo battery, the system targets 0.4 m/s mobility, 1.87 Nm weapon torque, <300 ms control latency, and >2-minute runtime, with tolerance analysis confirming a 2.16 safety margin on 14.6A peak draw.

Contents

1. Introduction

This project addresses the engineering challenge of creating a compact antweight combat robot that can reliably compete in two-minute, elimination-style matches while satisfying strict design rules on size, weight, materials, and safety. The system is a fully 3D-printed, sub-2 lb BattleBot that uses a front-hinged lifting shovel to flip or destabilize opponents, driven by a drivetrain and weapon system powered from an onboard battery and coordinated by a Bluetooth-enabled ESP32 microcontroller on a custom PCB. The robot's purpose is to demonstrate how careful integration of power delivery, wireless control, maneuverability, and offensive capability can produce a durable, agile platform that either outlasts or disables opposing bots under realistic competition constraints, making it a useful testbed for compact combat-robot design.

The project has four interconnected subsystems shown in Figure 1.1: drivetrain, power, weapon, and control. The wireless control subsystem utilizes the BLE module built into the ESP32-S3 microcontroller to process inputs with a latency under 250 ms, generating GPIO signals. Two DC motors are controlled by H-bridges for drivetrain mobility, and one servo motor receives 5 V and PWM signals from the ESP32. The power system takes in 12 V and outputs 3.3 V for the ESP32 and 5 V for the servo motor. The chassis and wheels are 3D-printed from PLA with a hollow structure to contain the PCB and battery inside. All these components have been integrated into a single PCB, resulting in a system that is efficient, mobile, and offensive.

During development, several changes were made. The motor drivers were changed from dual H-bridge DRV8833RTY chips to single H-bridge DRV8871 chips, doubling the chip count but allowing direct 12 V input; DRV8833RTY is limited to 10.8 V, and accurately regulating 12 V down to 10.8 V proved impractical, so DRV8871 was a better choice. The chassis and wheels went through multiple iterations, adding mounting points for screws to secure the PCB and connect the chassis to the upper battery cover. The weapon was modified to be narrower but longer for improved stability. Screws were added at the rear to increase the weight after the assembled BattleBot was measured at only 1 lb. A small breadboard was added to route the microcontroller and servo supplies in parallel so that when the servo draws current, the ESP32's supply does not dip below 3.3 V and reset. These modifications reflect a practical approach to meeting design specifications while working within project constraints.

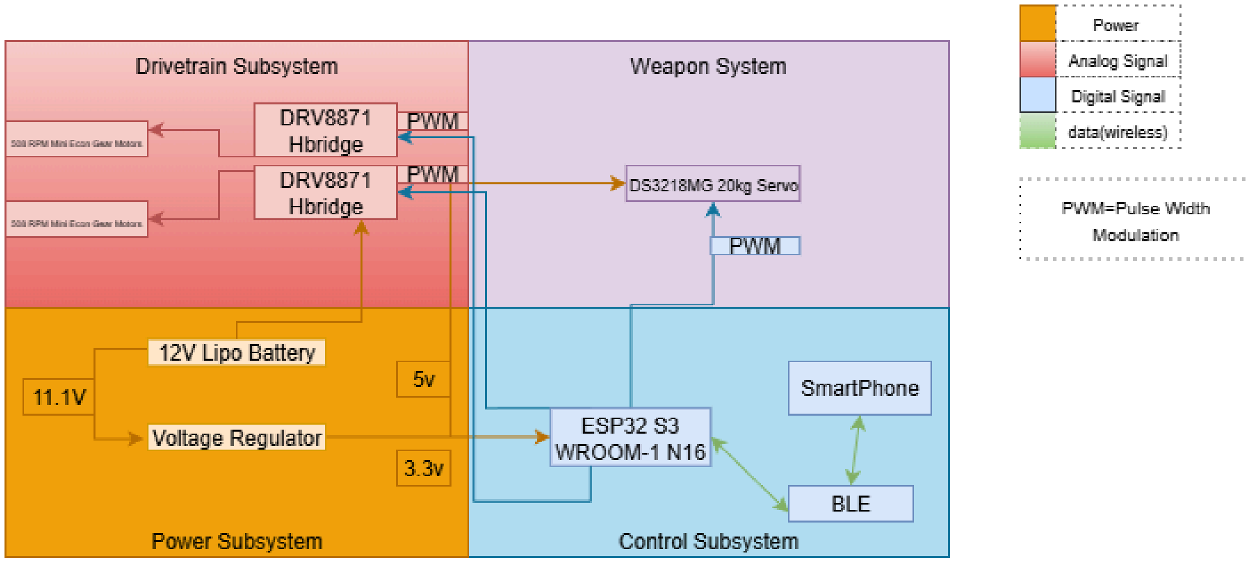

Figure 1.1: Block Diagram of the Project

1.1 High-Level Requirements

1. Performance requirement: The lifting wedge must be able to lift a 1 lb load to around 30° within 2 seconds and return to the starting position within seconds.

2. Control requirement:
The control system must process commands and actuate movement/weapon within 300 ms of the remote command and maintain stable wireless communication over at least 5 meters without signal loss.

3. Mobility requirement:
The drivetrain must move the bot at a minimum forward speed of 20 cm/s on a flat surface and produce enough power to move around for the entirety of the 2-minute battle time

2 Design

2.1 Design Procedure

Our antweight battlebot development followed a systematic engineering design methodology. The team began by establishing a contract to clearly define roles and responsibilities, enabling parallel work streams where each member owned specific subsystems and had final authority over their area. The workflow leveraged KiCad for detailed PCB design, the Arduino IDE for firmware, and an ESP32-S3 DevKit as the primary microcontroller platform, while Bambu Studio and a PLA-based 3D printing pipeline supported rapid iteration of the chassis and mechanical components.

Design started from competition constraints, particularly the 2 lb weight limit and 2-minute runtime, leading to an iterative process that emphasized mass budgeting, performance optimization, and safety compliance. Key decisions included using builtin Bluetooth Low Energy (BLE) on the ESP32-S3 for low-latency wireless control, selecting gear motors for precise and responsive mobility, and creating a spatial chassis that balances rigidity with minimal material. The weapon system evolved into a 5 V servo-driven shovel optimized for torque transfer, powered alongside the rest of the bot from a 3-cell 12.6 V Li-ion battery chosen for its energy density within the weight cap. Validation combined analytical calculations, tolerance and power-budget analysis, and physical testing of prototypes against competition metrics, allowing progressive refinement into a robust, competition-ready antweight battlebot that satisfies all technical and performance requirements.

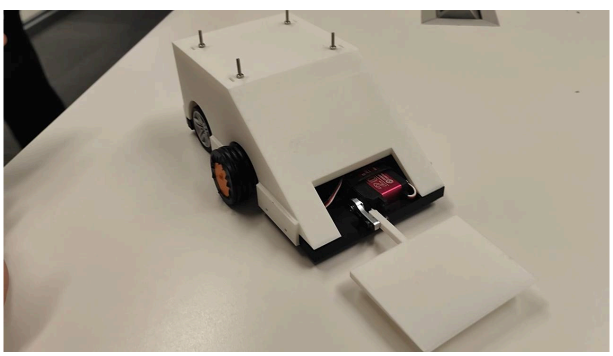
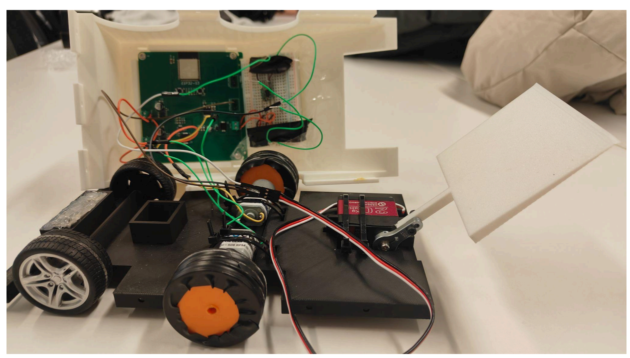

Figure 2.1 & 2.2: Exterior & Interior of BattleBot

2.2 Design Details

The design is composed of 4 main subsystems: Drive Train, Weapon, Power, Control. Each subsystem was developed to integrate together to meet technical requirements with weight control.

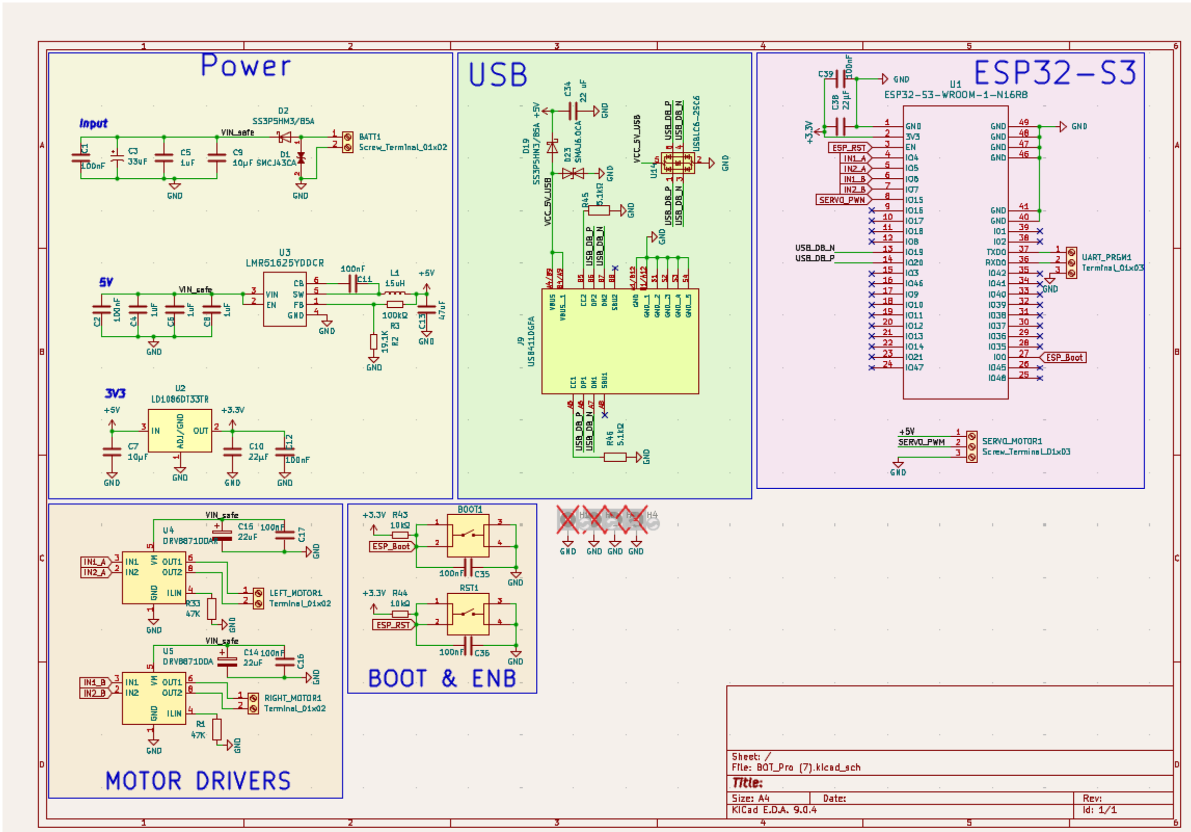

Figure 2.3: Final Schematic for BattleBot

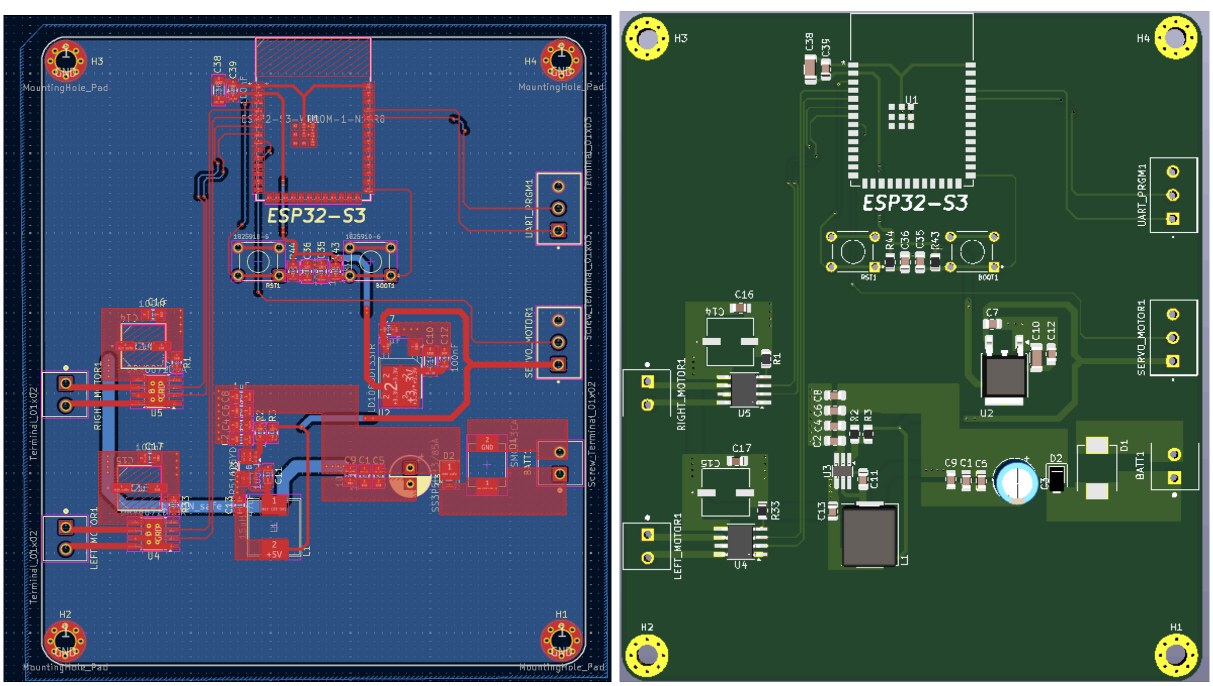

Figure 2.4 & 2.5: BattleBot PCB & its 3D rendering

2.2.1 Drivetrain Subsystem

We use four small wheels (2.25″), with the two rear wheels driven by high-torque 508 RPM, 12 V DC motors for our drivetrain subsystem. The small wheels lower the bot's ride height, reducing the center of gravity to improve stability and decrease the chance of being flipped, while still providing good traction. The selected motors strike a good balance between speed and torque, offering sufficient pushing power to maneuver our heavily armored bot effectively.

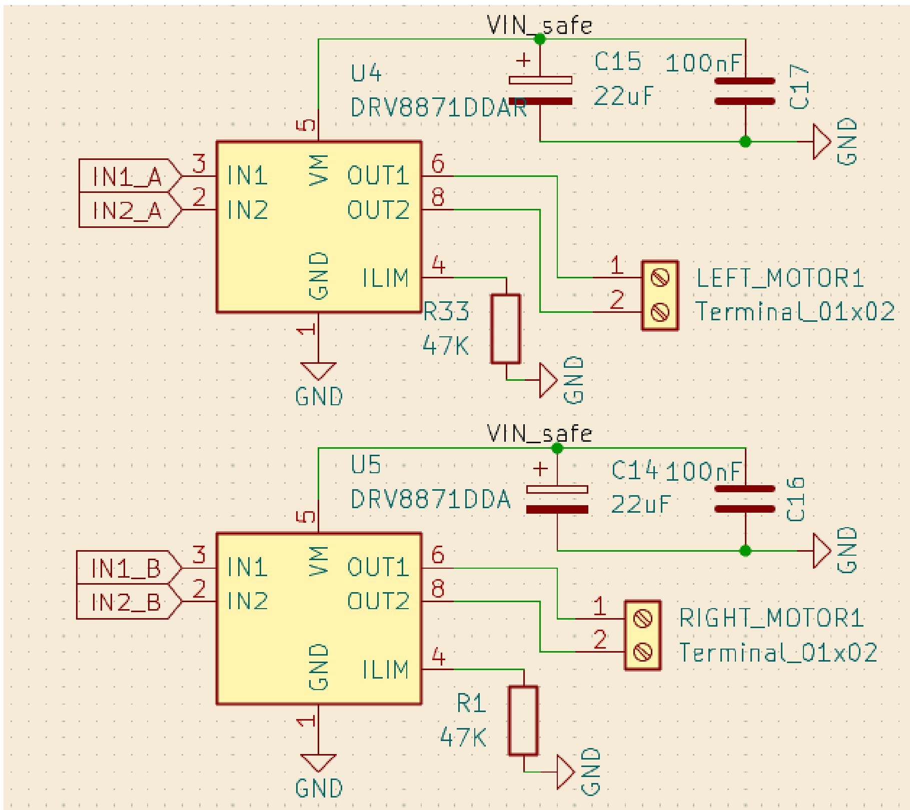

Figure 2.6 Drivetrain Subsystem Schematic

The drivetrain subsystem is responsible for all propulsion and maneuvering of the antweight BattleBot. It converts electrical energy from the 3-cell 12 V lithium-polymer battery into mechanical motion at the wheels, enabling the robot to accelerate, steer, and reverse during combat. Our drivetrain utilizes two 508 RPM Mini Econ Gear Motors (ServoCity), each driving one wheel through a differential control configuration. These 12V brushed DC motors provide enough stall torque and stall current to move the 2 lb robot quickly while still remaining efficient and lightweight. Operating them from a 12 V supply corresponds to an approximate wheel speed of 450 RPM and a top linear velocity of around 0.5 m/s for 2.25'' wheels.

The theoretical linear velocity is:

$$v = (\pi \cdot d \cdot N) / 60$$

where d is the wheel diameter (0.057 m) and N is the rotational speed (450 RPM). Substituting: $v = (\pi \times 0.057 \times 450) / 60 = 1.34$ m/s theoretical maximum. The measured speed of around 0.5 m/s reflects losses from friction, load, and voltage drop during operation.

This speed range is ideal for antweight combat robots, providing quick directional changes without sacrificing traction or control.

Motor control is handled by the DRV8871 dual H-bridge driver. This chip is an efficient solution for controlling two DC motors independently. The DRV8871 is controlled by an ESP32 microcontroller, which supplies four PWM and logic signals, IN1_A, IN2_A, IN1_B, and IN2_B, to adjust the direction and speed of each motor. This design allows smooth steering adjustments through differential speed modulation. Overall, the combination of the 508 RPM Mini Econ Gear Motors and the DRV8871 H-bridge driver forms an efficient, lightweight, and robust drivetrain solution. The motor's torque-to-weight ratio is well-suited for a 2 lb robot.

2.2.2 Weapon Subsystem

For our weapon subsystem, we use a front-hinged lifting wedge ("shovel") as our primary weapon to attack, destabilize, and flip opponent bots. The wedge will be 3D-printed in PLA for impact resistance and will be reinforced at hinge and linkage points to withstand stress. It spans 70% of the bot's width and features a low, angled tip that allows it to slide effectively under opponents.

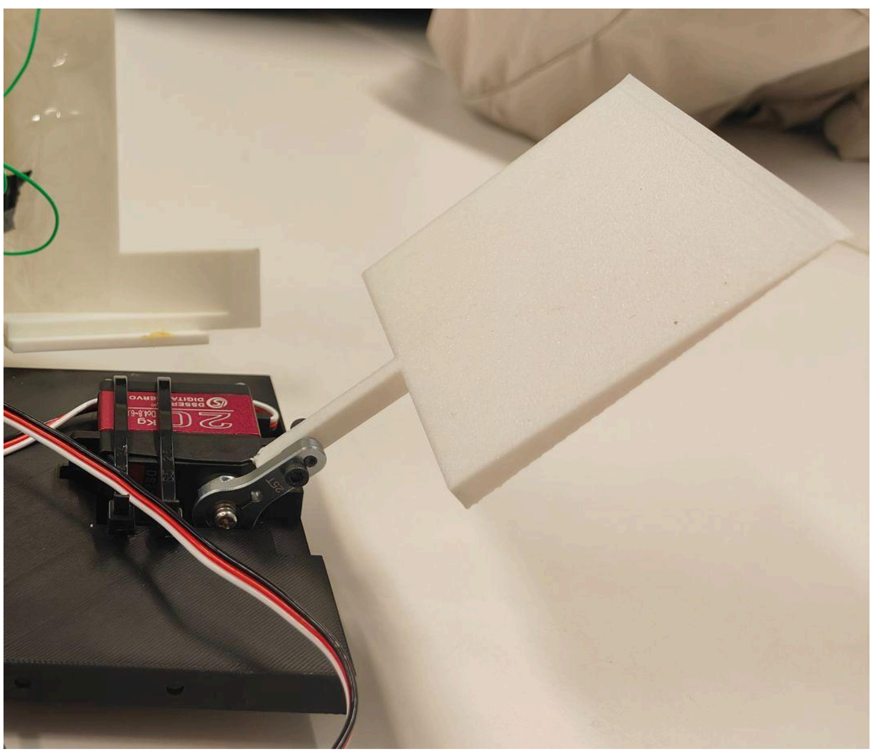

Figure 2.7: Weapon System

For the weapon's actuation, we use a High Torque Waterproof Metal Gear Digital Servo (20KG, 270°, IP54) operating at 12V with a shovel arm length of around 14cm. This servo provides robust torque output of approximately 20 kg·cm (1.96 N·m) at the specified voltage.

The minimum torque required to lift an opponent is:

$$\tau = m \cdot g \cdot L$$

where *m* is the load mass (1 lb = 0.454 kg), *g* is gravitational acceleration (9.81 m/s²), and *L* is the moment arm (0.14 m). The required torque is τ = 0.454 × 9.81 × 0.14 = 0.62 N·m. The servo's 1.96 N·m capacity provides a safety factor of 3.2×.

The metal gear construction ensures durability under combat conditions, while the IP54 waterproofing rating protects against dust and moisture. The servo's 270° range of motion allows for a full lifting and flipping action. For our design, we only use a 40° lift angle, which is enough to lift up the opponent battlebot. The lever linkage amplifies the servo's torque through mechanical advantage, enabling the wedge to destabilize and flip the 2-lb target effectively. The digital control interface provides precise position control and rapid response to input commands, ensuring reliable weapon deployment during combat.

2.2.3 Power Subsystem

The power subsystem is responsible for supplying the necessary voltage for the other subsystems. We use a Lithium-Polymer (LiPo) battery: 3S, 12 V, 450 mAh with a continuous discharge rate of 80C (Turnigy Nano-Tech Plus). This amount of capacity will ensure that the battlebot can last for 2 minutes in the competition.

The theoretical runtime at peak current draw is:

$$t = (C \times 3600)/I\_peak$$

where *C* is battery capacity (0.45 Ah) and *I_peak* is peak current (14.6 A): t = (0.45 × 3600) / 14.6 = 111 seconds. Since average current during combat is lower than peak, actual runtime exceeds the 2-minute match requirement.

The total peak current demand is:

$$Ipeak = I\_ESP32 + I\_motors + I\_servo = 0.8 + 8.6 + 5.2 = 14.6\ A$$

The battery's maximum continuous discharge is:

$$I\_max = C \times Crating = 0.45\ Ah \times 70C = 31.5\ A$$

The safety margin is:

$$Safety\ Margin = I\_max/I\_peak = 31.5/14.6 = 2.16$$

This confirms the battery can handle peak demands with significant headroom.

We utilize 2 buck converters for them, one LD1086DT33TR chip converts 12 V to 3.3 V for the ESP32 power input, and another LMR51625 chip converts 12 V to 5 V for power input into the servo motor(weapon subsystem). The drivetrain subsystem with the H-bridge chips DRV8871 can directly take the full 12 V voltage supply, so there is no need for it to have a buck converter.

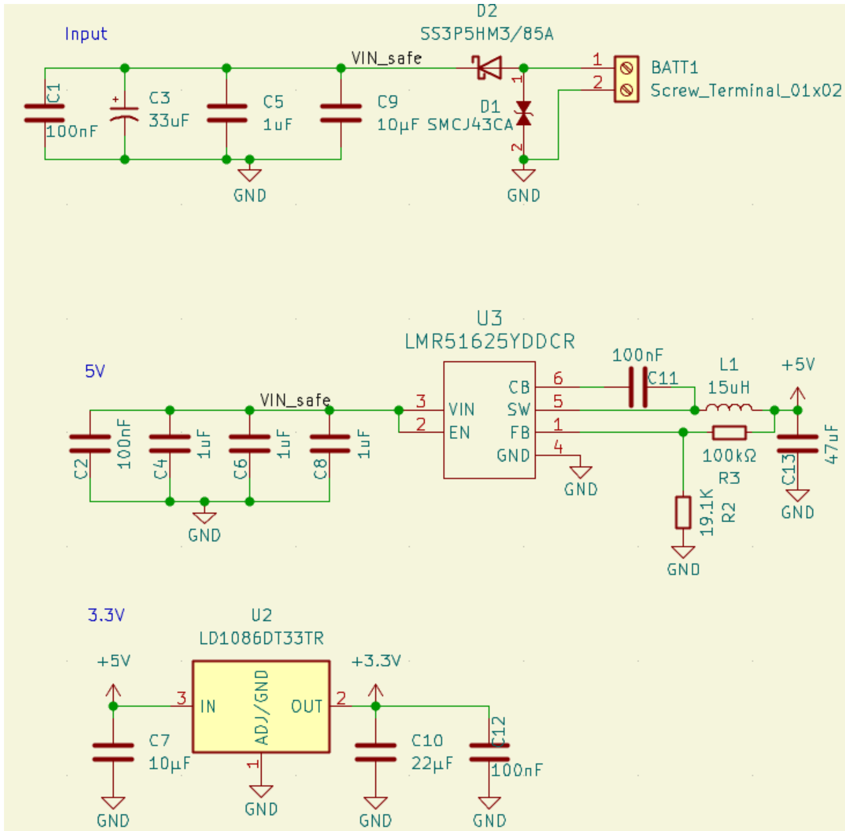

Figure 2.8: Power Subsystem Schematic

2.2.4 Control Subsystem

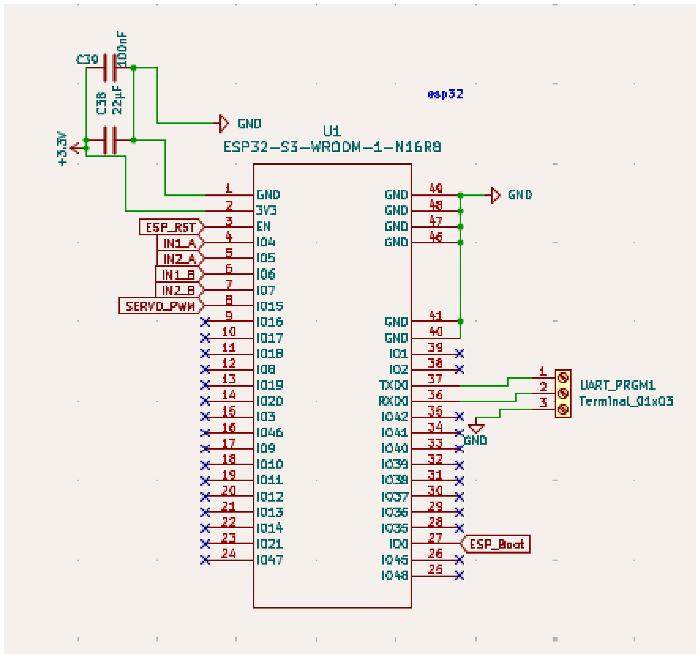

Figure 2.9 Drivetrain Subsystem Schematic

The control subsystem manages wireless communication, generates control signals for drivetrain motors and the servo motor, and coordinates all subsystem interactions. It is built around the ESP32-S3-WROOM-1-N16R8 microcontroller, selected for its integrated Bluetooth Low Energy (BLE), dual-core 240 MHz processor, and lightweight 6.5 g form factor. The integrated wireless capability eliminates the need for external Bluetooth modules, saving approximately 15–20 g compared to Arduino-based alternatives.

The battlebot is controlled via BLE using the Bluefruit Connect app, which provides a control pad interface with continuous command transmission, the robot moves while a button is held and stops upon release. BLE was chosen over Wi-Fi for its lower power consumption and simpler pairing protocol. Verification testing confirmed end-to-end latency under 250 ms, meeting our 300 ms requirement with margin.

The ESP32 controls the two DRV8871 H-bridge drivers using digital GPIO outputs. Pins IO4, IO5, IO6, and IO7 connect to IN1_A, IN2_A, IN1_B, and IN2_B, providing independent control of the left and right drive motors. The firmware translates Bluefruit commands into motor signals as shown in Table 2. When no button is pressed, all outputs are set LOW, placing both motors in coast mode and stopping the robot.

Command	Left Motor	Right Motor	Result
Forward	Forward	Forward	Straight Ahead

Backward	Reverse	Reverse	Straight Back
Left	Reverse	Forward	Rotate Left
Right	Forward	Reverse	Rotate Right

Table 1. Motor control signal mapping for movement commands.

The weapon servo is controlled via GPIO pin IO8, generating a 50 Hz PWM signal with pulse widths from 500 µs to 2400 µs. Two Bluefruit buttons control the lifting wedge: one raises the wedge to 40°, and another returns it to 0° for the next attack.

The ESP32 operates from the 3.3 V rail supplied by the LMR51625YDDCR linear regulator. Decoupling capacitors C38 (22 µF) and C39 (100 nF) are placed within 5 mm of the power pins to filter noise during wireless transmission. A 3-pin UART header provides the programming interface, with boot and reset buttons enabling firmware updates.

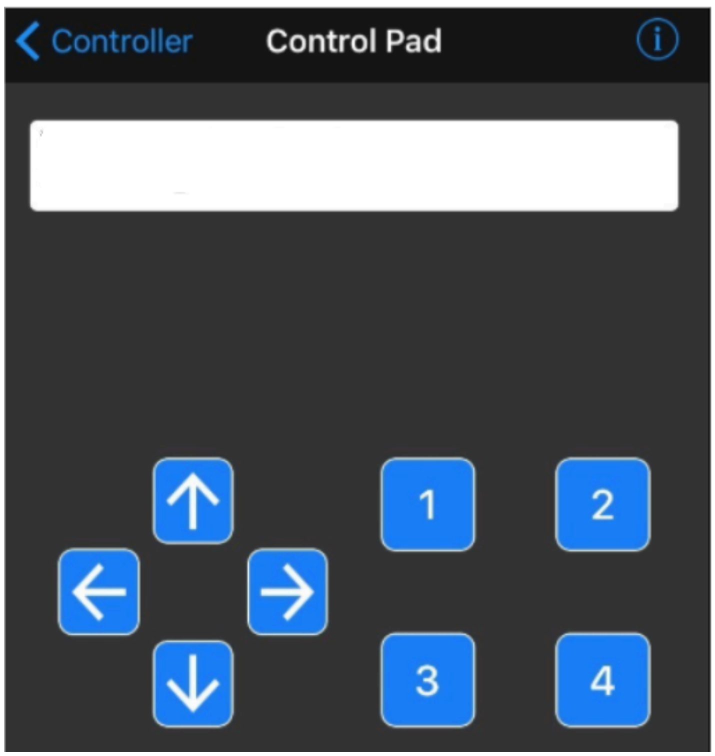

Figure 2.10 The Bluefruit Connect Control Pad is used for movement and turning controls. Button 3 moves the weapon up, and Button 4 moves it down.

3. Design Verification

3.1 Drivetrain Subsystem

The drivetrain subsystem is responsible for providing all translational and rotational motion of the battlebot, enabling it to accelerate, reverse, and turn within the arena while supporting the required offensive maneuvers. It converts electrical power from the battery into mechanical motion at the wheels through the DC motors and H-bridge drivers, and its performance directly determines the robot's mobility and agility.

One key requirement of the drivetrain subsystem is that the robot must be able to move both forward and backward. This is verified through a simple functional test: control commands are issued to drive the bot in each direction, and a visual inspection confirms that the robot translates smoothly forward and then backward without stalling or unintended rotation. A second quantitative requirement specifies that the linear speed of the bot must reach at least 20 cm/s. To verify this, the robot is driven across a measured distance while timing the motion with a stopwatch; the speed is computed as distance divided by time, and the result is 50 cm/s, checked against the 20 cm/s target. Finally, the drivetrain must allow the robot to turn left and right reliably. Verification is again performed visually by commanding left and right turns and confirming that the left and right drive motors spin in opposite directions as expected and that the chassis rotates in place or along an arc in both directions.

3.2 Weapon Subsystem

The weapon subsystem is responsible for providing the robot's primary offensive capability by using a front-mounted shovel to lift, destabilize, or tip opponents. It converts electrical power and control signals from the ESP32 into controlled rotational motion of a servo, which in turn raises and lowers the shovel.

One functional requirement is that the shovel must be able to lift a 1 lb load—modeled as a 1 lb water bottle—from the ground to a height of at least 5 cm. This is verified experimentally by placing the 1 lb bottle on the shovel, commanding a lift, and measuring the vertical distance between the bottle's base and the ground; the requirement is met if the measured height is at least 5 cm. A second requirement is that the shovel, when commanded by the control system, must reach its target position within 3 s. To verify this, a timer or stopwatch is used to record the time from the instant the activation command is sent to the moment the shovel reaches its final position; the subsystem passes if this time does not exceed 3 s.

3.3 Power Subsystem

The power subsystem is responsible for delivering stable voltages and sufficient current to all other subsystems, ensuring the robot can operate at full performance for the entire match duration. It conditions the 12 V output of the Li-ion battery into regulated rails for the drivetrain, control electronics, and weapon servo while preventing brownouts or unsafe operating conditions.

One key requirement is that the battery must power the robot for at least the full 2-minute battle duration. This is verified by operating the bot under combat-like conditions—drivetrain and weapon activated in typical duty cycles—and using a timer to confirm that the system remains functional without shutdown or noticeable performance degradation for at least 2 minutes. Another requirement is that the power subsystem must provide a stable 12 V supply to the H-bridge motor drivers. Verification is performed using a voltmeter to measure the voltage at each H-bridge input while the motors are running, confirming that it remains close to 12 V. In addition, the regulators must deliver 3.3 V to the ESP32 and 5 V to the servo. These rails are checked with a voltmeter at the microcontroller and servo supply pins under load to ensure they remain within allowable tolerances for reliable operation.

3.4 Control Subsystem

The control subsystem is responsible for interpreting wireless inputs from the user, generating appropriate motor and servo commands, and coordinating all subsystem behavior in real time. It uses the ESP32's Bluetooth Low Energy link to receive commands, then outputs PWM and GPIO signals to the drivetrain H-bridges and weapon servo while maintaining low latency and robust connectivity.

One key requirement for the control subsystem is that the command latency must not exceed 300 ms from user input to observable motion of the robot. This is verified by filming the control interaction with a high-frame-rate camera: a clear input event (such as pressing a key or moving a joystick) and the corresponding robot response are captured in the same video, and the latency is calculated by counting frames between the two events and converting to time using the frame rate. Another requirement is that the system must maintain reliable control at a practical operating distance with obstacles present. This is tested by operating the battlebot while standing at least 5 meters away, with tables and benches placed between the operator and the robot, and verifying that all driving and weapon commands are executed consistently without dropouts or loss of responsiveness.

4. Costs

Labor: $40/hour - Software Development (20 hours) - Schematic and PCB Design (30 hours) - CAD Design (15 hours) - 3D Printing (10 hours) - Electrical Assembly (10 hours) - Mechanical Assembly (5 hours) - Testing and Iteration (10 hours)

Total cost: $10,000 (labor)

Our PCBs and stencils are being printed by orders from PCBWay, which cost us a total of $83.47, including shipping, tariffs, and customs clearance fees.

All of our battlebot's chassis, weapons, and front 2 wheels are 3D-printed with the Bambu X1 Carbon 3D Printer in 2070 ECEB with no additional cost to print any of the parts.

For other parts not included in the parts table below, they are minor, personally owned items and do not account for the cost of this project.

Senior Design Lab Resources: We have utilized various items in the senior design lab at ECEB2070, including the 3D printer, oscilloscope, multimeter, soldering irons, solder paste, DC power supplies, and hot glue.

4.1 Parts

Part	Manufacturer	Qty	Retail Cost ($)	Actual Cost ($)
Servo (ANNIMOS DS3218MG 20KG)	ANNIMOS	1	$13.99	$13.99
DC Motor (ServoCity 508 RPM Mini Econ)	ServoCity	2	$12.99	$25.98
Wheels (48mm Rubber Tire, 10-pack w/shafts)	URIMPAVIDO	2	$8.99	$17.98
Battery (OVONIC 11.1V 3S 450mAh 2-pack)	OVONIC	1	$24.23	$24.23
ESP32-S3-WROOM-1-N16R8	Espressif	1	$6.56	$6.56
DRV8871DDAR Motor Driver	Texas Instruments	2	$2.73	$5.46
LMR51625YDDCR Buck Converter	Texas Instruments	1	$2.03	$2.03
LD1086DT33TR Linear Regulator	STMicroelectronics	1	$1.07	$1.07
100nF Capacitor 0805 (KGM21NR71H104JT)	KYOCERA AVX	9	$0.10	$0.90
1µF Capacitor 0805 (GMC21X7R105J50NT)	Cal-Chip	4	$0.12	$0.48
10µF Capacitor 0805 (GRM21BR61H106KE43L)	Murata	2	$0.15	$0.30
22µF Electrolytic (EEE-FN1J220XP)	Panasonic	2	$0.45	$0.90
22µF Ceramic 1206 (CL31A226KAHNNNE)	Samsung	2	$0.25	$0.50
33µF Electrolytic (336CKE063M)	Cornell Dubilier	1	$0.85	$0.85
47µF Capacitor 0805 (C2012X5R1A476M125AC)	TDK	1	$0.35	$0.35
15µH Inductor (CDRH8D38NP-150NC)	Sumida	1	$1.50	$1.50
10kΩ Resistor 0805 (RK73H2ATTD1002F)	KOA Speer	2	$0.01	$0.02
19.1kΩ Resistor 0805 (RK73H2ATTD1912F)	KOA Speer	1	$0.01	$0.01
47kΩ Resistor 0805 (RK73H2ATTD4702F)	KOA Speer	2	$0.01	$0.02
100kΩ Resistor 0805 (RG2012P-104-B-T5)	Susumu	1	$0.15	$0.15
2-Pin Screw Terminal (1776275-2)	TE Connectivity	3	$0.50	$1.50
3-Pin Screw Terminal (TAD05-03-1-L-G)	GCT	2	$0.60	$1.20
Tactile Switch (1825910-6)	TE Connectivity	2	$0.13	$0.26
SMCJ43CA TVS Diode	Taiwan Semiconductor	1	$0.45	$0.45
SS3P5HM3/85A Schottky Diode	Vishay	1	$0.35	$0.35
Total		**48**		**$107.04**

Figure 4.1: Components Table

4.2. Schedule

Week of 9/29	● Finalize component selection and system architecture for control subsystem (All) ●Create initial KiCad schematic (ESP32, voltage regulator, motor drivers, weapon motor, sensors) (All)
Week of 10/6	• Test initial proof-of-concept on breadboard (basic motor control) (All) • Finalize and submit PCB design for fabrication (All) • Order all electronic components and mechanical parts (All)
Week of 10/13	• Build breadboard prototype with all subsystems connected (All) • Develop core firmware (Bluetooth communication, PWM motor control, voltage regulator, weapon control) (All) • Begin chassis design and 3D printing structural components (John)
Week of 10/20	• Complete mechanical chassis assembly and test fitment (John) • Implement weapon servo control and emergency stop functionality (Jimmy) • Test and tune motor control for required speed and torque specifications (Jimmy and John)
Week of 10/27	• Demonstrate functional breadboard prototype with wireless control (John) • Verify all performance requirements (~1 m/s speed, <300 ms latency, 10 m range, 2 min runtime) (Mig) • Refine code for reliability (Jimmy and Mig)

Week of 11/3	• Receive and assemble a custom PCB with all components (Mig) • Test PCB functionality and debug any hardware issues (Jimmy) • Begin integration of PCB with mechanical chassis (All)
Week of 11/10	• Complete full system integration (PCB, motors, battery, chassis, weapon) (Jimmy and Mig) • Wire all subsystems and verify electrical connections (John) • Conduct comprehensive system testing(drivetrain, weapon actuation sensors) (All)
Week of 11/17	• Run full combat simulations and stress tests (All) • Fix any identified issues and improve robustness (All) • Optimize control responsiveness and weapon performance (All)
Week of 11/24	• Final reliability testing and durability improvements (All) • Add strain relief, secure loose components, reinforce weak points (All) • Prepare backup components and document troubleshooting procedures (All)
Week of 12/1	• Complete final performance testing and data collection (All) • Prepare demonstration and document system performance (All)
Week of 12/8	• Complete documentation with final design, results, and analysis (All) • Deliver presentation and demonstrate complete battlebot (All)

5. Conclusion

For this project, we successfully produced a compact antweight BattleBot that meets the competition's strict requirements on weight, materials, functionality, and safety. Through the integration of drivetrain, power, weapon, and control subsystems, the final design demonstrates reliable maneuverability, low-latency wireless control, and effective lifting performance suitable for two-minute combat matches.

Key improvements made during development, such as switching from DRV8833 to DRV8871 motor drivers for direct 12 V operation, reinforcing the 3D-printed chassis, revising wheel and wedge geometry, and adding power routing parallel with the PCB to prevent power distribution issues, which substantially increased system stability and consistency during testing.

Overall, the completed robot is durable, agile, and capable of coordinated offensive action. The project demonstrates how careful electrical integration, mechanical iteration, and subsystem-level testing can produce a robust combat-ready platform within tight engineering constraints.

5.1 Ethical considerations

Our project follows the IEEE Code of Ethics (IEEE, 2025) and the ACM Code of Ethics (ACM, 2025) to ensure responsible engineering practice. Key ethical considerations for our BattleBot include the safety of participants and bystanders, fair competition, respect, and team integrity. We will prioritize the health and safety of all team members, competition officials, and bystanders. Risks such as LiPo batteries, high-speed motors, and weapon systems will be addressed with proper handling procedures and kill-switch buttons. For fair competition, we will comply with all rules in BattleBot and make sure we don't exceed the weight or use materials other than the allowed ones. For Respect and Team Integrity, we will treat all participants fairly and with respect, avoiding harassment, discrimination, or misconduct in any form.

5.2 Challenges and Solutions

The most significant challenge involved power distribution stability. When the weapon servo activated, its high inrush current caused the 3.3 V rail to drop below the ESP32's minimum operating voltage, resetting the microcontroller and disconnecting Bluetooth. This limited weapon uses a single activation before requiring re-pairing. Although the PCB-mounted buck converter functioned correctly under steady-state conditions, it could not handle the transient load. We resolved this by adding an external buck converter module on a breadboard, powered in parallel directly from the battery and dedicated solely to the servo, which isolated the current transients and eliminated the reset issue.

Additional challenges arose in the mechanical subsystems. The purchased wheels had bore sizes incompatible with the motor shafts, so we 3D-printed custom wheels with matching dimensions. However, the PLA surface lacked sufficient traction, causing wheel slip during movement. We addressed this by wrapping electrical tape around the wheel circumference to increase friction. The weapon also required redesign after the initial shovel cracked during load testing due to stress concentration at a single attachment point. The revised geometry distributes forces across a wider hinge and linkage interface, enabling the shovel to lift a 2 lb load without structural failure.

5.3 Safety

Safety is the most important aspect of combat robotics. Our team will comply with NFPA electrical safety standards (NFPA, 2025) and competition rules (NHRL, 2025) to minimize risks. Key measures include:

● Electrical Safety: The 12V LiPo battery will be housed in an insulated, impact-resistant enclosure and charged only with approved balance chargers. We will follow IEEE 1725-2011 battery management guidelines to prevent overheating, short-circuits, or fire hazards.

● Mechanical Safety: Moving parts such as wheels and weapons will not be activated outside of enclosed test zones. A kill switch will be implemented to immediately cut power in emergencies.

● Wireless Communication: Since the robot uses Bluetooth/Wi-Fi, we will configure secure authentication to prevent unauthorized access and ensure the robot enters a safe shutdown state if communication is lost.

● Lab and Shop Safety: We will wear PPE (goggles, gloves) while handling batteries, soldering, or machining parts, and follow UIUC lab safety rules. Lockout/tagout (LOTO) procedures will be followed during maintenance.

5.4 Future Improvements

Several improvements could enhance the robot's durability and combat effectiveness in future iterations. The chassis should be redesigned to provide greater protection for the servo motor, which is currently exposed to potential impacts during matches. An enclosed servo compartment with reinforced walls would shield the actuator from direct hits and debris. Additionally, increasing the servo's range of motion would improve offensive capability against larger or irregularly shaped opponents. The current 40° lift angle proved insufficient in some matches; expanding this to 60° or beyond would allow the shovel to achieve steeper lift angles and more effectively destabilize heavier robots.

The most critical improvement involves adding current sensing to the weapon system. During our third competition match, the shovel became lodged under an arena chair, causing the servo to stall and draw excessive current. When the shovel was suddenly released, the stored energy caused the servo to overshoot its intended position, mechanically damaging the actuator and ending our participation. Implementing a current sensor on the servo power line would enable the firmware to detect stall conditions in real time. When current exceeds a defined threshold, the system could automatically cut power to the servo or reverse its direction, preventing both electrical damage from overcurrent and mechanical damage from uncontrolled motion upon release.

References

[1] IEEE, "IEEE Code of Ethics," IEEE Policies, Section 7 - Professional Activities. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8. [Accessed: Oct. 13, 2025].

[2] Association for Computing Machinery, "ACM Code of Ethics and Professional Conduct." [Online]. Available: https://www.acm.org/code-of-ethics. [Accessed: Oct. 13, 2025].

[3] National Fire Protection Association, "NFPA - Home." [Online]. Available: https://www.nfpa.org/. [Accessed: Oct. 13, 2025].

[4] Espressif Systems, "ESP32-S3-WROOM-1 & ESP32-S3-WROOM-1U Datasheet," ESP32-S3-WROOM-1_WROOM-1U datasheet, Ver. 1.6, 2023.

[5] ServoCity, "508 RPM Mini Econ Gear Motor." [Online]. Available: https://www.servocity.com/508-rpm-mini-econ-gear-motor/. [Accessed: Oct. 13, 2025].

[6] DFRobot, "Fermion: BMP388 Barometric Pressure Sensor." [Online]. Available: https://www.dfrobot.com/product-2787.html. [Accessed: Oct. 13, 2025].

[7] HobbyKing, "Turnigy Nano-Tech Plus 450mAh 3S 70C Lipo Pack w/XT30." [Online]. Available: https://hobbyking.com/en_us/turnigy-nano-tech-plus-450mah-3s-70c-w-xt30.html. [Accessed: Oct. 13, 2025].

[8] HiLetgo, "TB6612FNG Dual Motor Driver Module." [Online]. Available: https://www.amazon.com/HiLetgo-Stepper-Controller-Tb6612fng-Replace/dp/B00UYIFY CW. [Accessed: Oct. 13, 2025].

[9] Texas Instruments, "LM2596 SIMPLE SWITCHER 3-A Step-Down Voltage Regulator," LM2596 datasheet. [Online]. Available: https://www.digikey.com/en/products/detail/texas-instruments/LM2596S-3-3/3701219. [Accessed: Oct. 13, 2025].

[10] Texas Instruments, "MCF8316A Sensorless BLDC Motor Driver," MCF8316A1VRGFR datasheet. [Online]. Available: https://www.mouser.com/ProductDetail/Texas-Instruments/MCF8316A1VRGFR. [Accessed: Oct. 13, 2025].

Appendix A - Requirements and Verification Table

Subsystem	Requirement	Verification	Verification status (Y or N)
Drivetrain subsystem	The robot is able to move forward and backward	Visual check	Y
	The linear speed of the bot has to reach 20 cm/s speed	Record the distance traveled and the time it took	
	The robot is able to turn left and right	Visual check (left and right motor move in opposite directions when turning)	
Weapon subsystem	The shovel is able to lift 1 lb	Check height above the ground for 5 cm	Y
	The shovel should be activated by the control system and reach the destination position in 3s	Measure using a timer	
Power subsystem	The battery has to last for at least 2 minutes	Timer to measure whether the robot can function for the entirety of the 2-minute battle time	Y
	12 V supplying H-bridge	A voltmeter to check each output of the buck converter	
	3.3 V supplying ESP32		
	5 V supplying the servo		
Control subsystem	Command latency of at most 300 ms	Filming a video of controlling the battlebot and using video frames to calculate latency	Y
	Maintain control of the robot by controlling it from a distance	Being able to control the battlebot standing at least 5 meters away, with tables and benches as obstacles	