ECE 445 Spring 2026

Senior Design Project Proposal

Circular Antweight Battlebot

By Group 24

Junyan Bai, Yuxuan Guo

TA: Zhuoer Zhang

Contents

1. INTRODUCTION

1.1 Problem

In order to compete in the antweight (≤ 2 lb) 3D-printed battlebot competition, our robot must satisfy several strict constraints: it must weigh less than 2 lb, be primarily 3D printed using allowed plastics (PET, PETG, ABS, PLA/PLA+), include a locomotion system and an active fighting tool, and be controlled from a PC over Bluetooth or Wi-Fi using a custom PCB (microcontroller, Bluetooth/Wi-Fi receiver, and H-bridge motor control). Additionally, the robot must support an easy manual shutdown and automatically disable if the RF link is lost, and external RC car-style receivers are not allowed.

Within these constraints, the main goal of the competition is to design a robot that can disable or disrupt opposing robots during a two-minute match in a controlled 10’×10’ plexiglass arena. In practice, many combat robots fail not because their concept is weak, but because they get stuck on opponents or arena walls, lose traction during pushing, or experience motor stalls that cause brownouts and wireless dropouts. These issues are not only performance problems; they also create safety and reliability concerns because an uncontrollable robot can behave unpredictably during development and testing. Therefore, our problem is to build a compact, durable antweight robot that can consistently win positional control and remain fully controllable under impacts and stalls while meeting the competition’s safety requirements.

1.2 Solution

We propose a circular “UFO-shaped” antweight battlebot that prioritizes control, robustness, and consistent engagement. The robot will use a protected two-wheel drivetrain integrated into a low-profile circular chassis, reducing exposed corners and snag points while improving survivability in collisions. The primary fighting tool will be a motor-driven shovel/lifter mechanism mounted at the front. During a match, the operator will drive the robot to align the shovel edge under an opponent and actuate the shovel to lift or destabilize the opponent, enabling follow-up pushing, pinning, or redirection into arena boundaries.

Electrically, the robot will be controlled from a PC via Bluetooth or Wi-Fi using a custom PCB that integrates the microcontroller, wireless receiver, and motor control hardware. The design will incorporate both (1) a manual disconnect that can quickly shut down all motion and (2) an automatic link-loss shutdown so the robot stops moving and disables the shovel mechanism if communication is lost. This approach emphasizes controlled, repeatable performance under real match conditions rather than relying on fragile high-energy weaponry.

1.3 Visual Aid

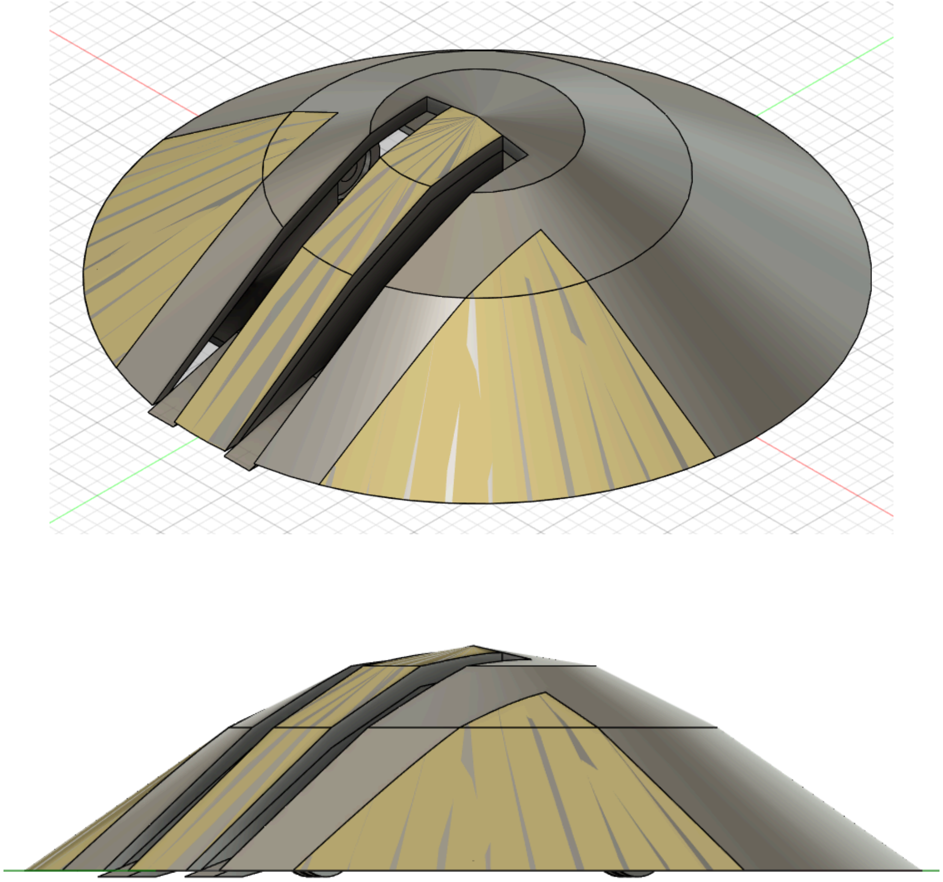

Figure 1: Circular "UFO" antweight battlebot with motorized shovel mechanism.

The robot is driven by two wheels recessed into the chassis for protection. A front shovel assembly extends from the body and is actuated by an internal motor to lift/scoop an opponent. The operator controls the robot from a PC over Bluetooth or Wi-Fi; commands are sent wirelessly to the custom PCB inside the robot. If the wireless link drops, the robot automatically shuts down drive and shovel actuation.

1.4 High-Level Requirements

1. The shovel mechanism must be able to raise a 2lb test block by ≥ 10 mm within ≤ 1.0 s, then across 1 meter on the arena surface without MCU reset or power brownout.
2. The robot must be fully controllable from a PC over Bluetooth or Wi-Fi using the onboard custom PCB motor control system, and it must respond to operator commands with an average end-to-end latency of ≤ 150 ms during testing.
3. The robot must support manual shutdown and must automatically disable all motion within ≤ 2 seconds after RF link loss to ensure safe and predictable operation during testing and competition.

2. DESIGN

2.1 Block Diagram

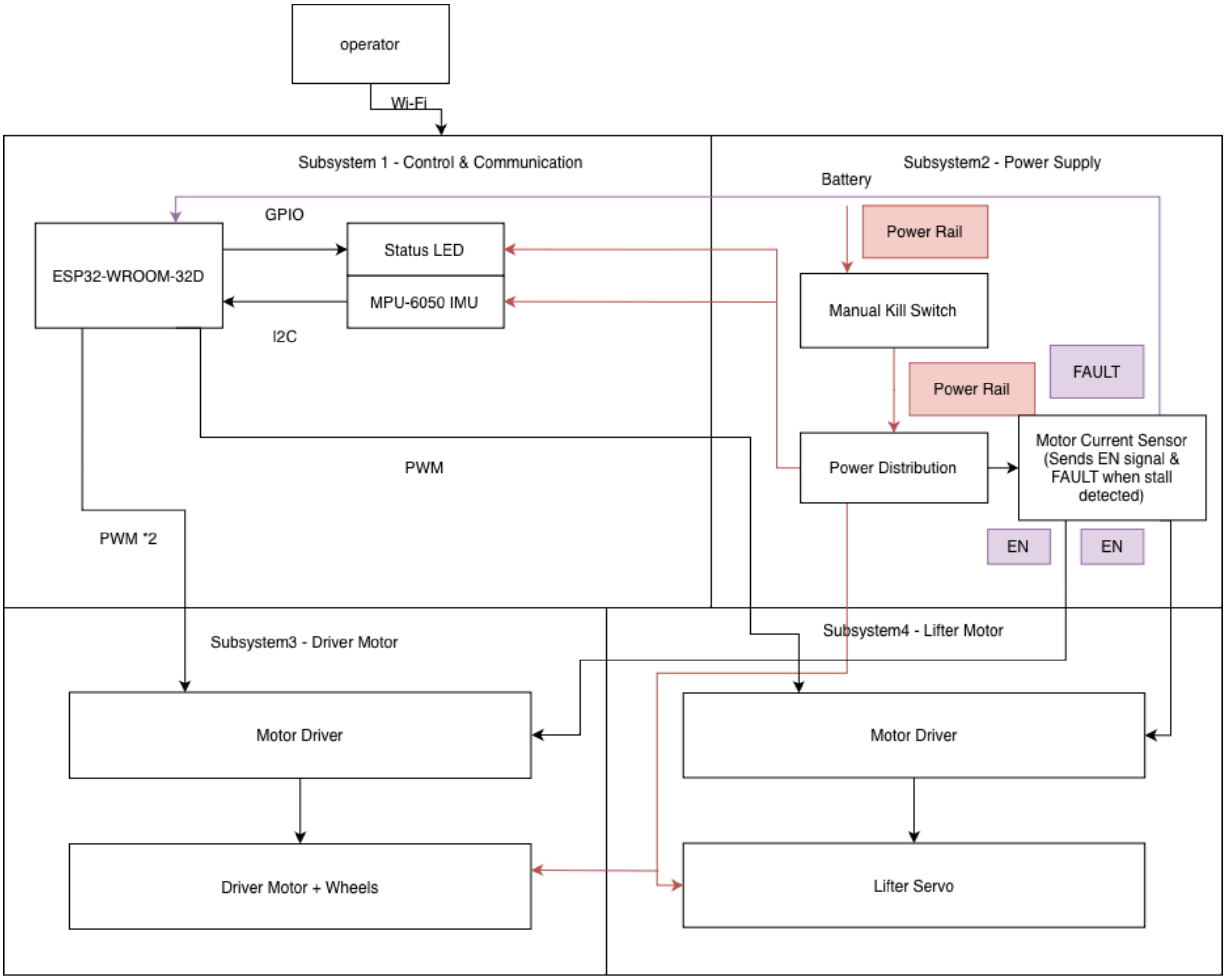

Figure2. System-level block diagram

Figure 2 shows the current system-level block diagram for our circular antweight battlebot. The operator sends commands from a PC over Wi-Fi to the on-robot ESP32-WROOM-32D, which drives two motor subsystems (drive and weapon) through PWM control signals. The ESP32 also communicates with onboard sensors (MPU-6050 over I2C) and drives status LEDs for arming/link feedback. Power originates from the battery and flows through a manual kill switch and power distribution stage. A motor current sensor monitors motor current and produces an EN (enable) signal and FAULT output that can rapidly disable motor drive in stall/overcurrent conditions, adding a hardware safety layer on top of firmware failsafes. This modular structure follows the proposal guideline that each block should be a self-contained subsystem with clear interconnections and quantitative requirements.

2.2 Subsystem Overview

2.2.1 Control & Communication

The ESP32-WROOM-32D receives operator commands via Wi-Fi and translates them into motor control outputs for drive and weapon actuation. It also monitors robot state through an IMU (MPU-6050) over I2C and drives status LEDs to communicate armed/link/fault states to observers.

2.2.2 Power Supply & Safety

The battery feeds a main motor power rail through a manual kill switch and distribution stage. A current-sensing protection block detects stall/overcurrent events and generates FAULT/EN signals that can disable motor drivers to prevent overheating, brownouts, or wiring damage.

2.2.3 Drive Motor (Mobility)

This subsystem includes the drive motor driver and the drive motor + wheels. It converts ESP32 PWM commands into controlled bidirectional wheel motion for steering, acceleration, and pushing.

2.2.4 Weapon Motor (Shovel/Lifter Actuation)

This subsystem includes the weapon motor driver and the weapon motor + roller/shovel mechanism. It provides controlled actuation of the front mechanism used to scoop/lift opponents and maintain positional control during a match.

2.3 Subsystem Requirements

2.3.1 Subsystem 1 — Control & Communication (ESP32 + IMU + LEDs)

This subsystem is responsible for predictable control and meeting the high-level responsiveness and safety goals required for a remotely operated battlebot (PC control over Wi-Fi/Bluetooth is mandated by the competition format).

Block contribution to high-level requirements: Ensures timely command response, stable teleoperation, and safe shutdown behavior (link-loss and fault handling), enabling reliable mobility and weapon control.

Requirements:

1. The ESP32 must maintain a Wi-Fi command link with end-to-end command-to-actuation latency of average ≤ 150 ms, measured over 2–5 minutes of driving (measured as time from operator input to PWM update at the motor driver).
2. The control loop must update motor commands at ≥ 50 Hz (20 ms period) to ensure smooth steering and shovel/weapon actuation.
3. The ESP32 must communicate with the MPU-6050 over I2C at standard I2C fast mode (400kHz) and read IMU updates at ≥ 50 Hz for orientation/impact detection and optional control features.
4. Status indication must provide at least:
    a. Main power / armed indicator visible from outside the robot (LED on when logic is powered and robot is armed).
    b. Link indicator (LED on when valid command packets are being received at the required rate).
5. Firmware must implement a link-loss failsafe: if no valid command packets are received, all motor PWM outputs must transition to a disabled/safe state within ≤ 2.0 s after link loss (0 throttle / brake as supported), and remain disabled until commands resume and the robot is explicitly re-armed.

2.3.2 Power Supply & Safety (Battery + Kill Switch + Distribution + Current Sense)

This subsystem provides stable power delivery and hardware-layer safety enforcement. It is also directly tied to competition safety expectations such as manual shutdown and safe behavior under loss of control link.

Block contribution to high-level requirements: Enables uninterrupted operation during a match (prevents brownouts/resets), protects electronics and wiring under stalls/impacts, and guarantees immediate manual disable capability.

Requirements:

1. Motor power must be disconnected within one switching action.
2. Power distribution must supply:
    a. A logic rail suitable for the ESP32 + sensors (target 3.3 V ± 5%, continuous ≥ 500 mA budget).
    b. A motor rail sourced from the battery for motor drivers (battery voltage direct or regulated; final value TBD).
3. The current-sense protection block must detect overcurrent, assert FAULT within ≤ 250 ms and disable within ≤ 200 ms if current exceeds a programmable threshold. (Threshold = 2× rated operating current (or 80–90% of driver peak) sustained for > 100 ms.).
4. Upon FAULT assertion, the protection block must deassert/disable EN to motor driver(s) within ≤ 50ms, preventing prolonged stall heating.
5. The power subsystem must tolerate peak motor loads without logic brownout; during a worst-case push + shovel actuation event, the logic rail must remain above 3 V (brownout margin) for continuous control.

2.3.3 Subsystem 3 — Drive Motor (Motor Driver + Drive Motor + Wheels)

This subsystem converts the controller's PWM commands into bidirectional traction for the two-wheel drivetrain. Choose from N20 and 16mm gearmotor.

Block contribution to high-level requirements: Provides mobility and pushing capability, enabling the robot to reach opponents, maintain contact, and reposition for shovel engagement.

Requirements:

1. The drive motor driver must accept PWM control at ≥ 1 kHz (or the driver's specified PWM range) and support forward/reverse operation with braking or coast mode as implemented.

2. The drivetrain must achieve a minimum straight-line speed of ≥ 0.4 m/s on arena-like flooring with the full robot mass.
3. The drivetrain must be capable of pushing a 1.0 kg test sled (or equivalent friction load) for ≥ 2 s without causing a logic brownout reset.
4. The drive motor driver must handle ≥ 3 A continuous and ≥ 5 A peak (≤ 1 s) per drive motor (exact values to be finalized after motor selection), without thermal shutdown during a 2-minute match duty cycle.

2.3.4 Subsystem 4 — Weapon Motor (Motor Driver + Shovel/Roller Actuator)

This subsystem controls the front mechanism responsible for lifting/scooping opponents (your "shovel" weapon). In your current diagram it is labeled "roller motor," but functionally this is the dedicated actuator for the front engagement mechanism. Planning to use MG996R servo or FS5115M-FB servo (implanted with angle feedback for stall detection).

Block contribution to high-level requirements: Provides the match-winning control action—lifting or destabilizing the opponent—while remaining safe under stalls via FAULT/EN gating.

Requirements:

1. The weapon mechanism must lift the leading edge of a 0.9 kg (2 lb) test block by ≥ 40 mm within ≤ 2 s when starting from the closed position, and maintain the lifted position for ≥ 5 s without servo thermal shutdown.
2. The weapon motor driver must support commanded actuation updates at ≥ 50 Hz (matching the controller loop) and must tolerate stall events safely:
   a. If the mechanism jams, the FAULT/EN protection must disable the motor within the specified ≤ 50 ms window (from Subsystem 2).
3. The weapon subsystem must not exceed the thermal limits of its driver during a 2-minute match; target sustained duty cycle ≥ 30% on-time at typical load without thermal shutdown.

2.4 Tolerance Analysis

2.4.1 Drive System

In this section we analyze the tolerance of 'pushing a 1kg sled'.

We estimate the friction constant to be worst case 0.7, thus calculation below:

F_required = $\mu \times m \times g = 6.87$N

Available motor torque (worst case):

1. Battery voltage under load: 9.0V (from 11.1V nominal, 3S LiPo sag)
2. Rated torque at 12V: 2.5 kg-cm
3. Torque at 9V: 2.5 × (9/12) = 1.875 kg-cm
4. With -15% motor tolerance: 1.875 × 0.85 = 1.594 kg-cm
5. Convert to N-m: 1.594 kg-cm × 0.0981 = 0.156 N-m

Wheel radius (Nominal):

1. 22.5 mm - With +0.5mm tolerance: 23.0 mm = 0.023 m
2. Force per motor: F_motor = T / r = 0.156 / 0.023 = 6.78 N

Total force (2 motors): F_total = 2 × 6.78 = 13.56 N > 6.87 N required

2.4.2 Weapon System

In this section we calculate the necessary torque to lift 0.9kg at 120mm arm (height extended to 60mm so enough margin for 40mm requirement):

Worst case conditions:

1. Arm length: 120mm + 1mm (tolerance) = 121mm = 12.1cm
2. Lift angle: 30°
3. Load: 0.9kg
4. Servo voltage: 5.5V (battery sag)

T_required = m × L × g × sin(θ) = 0.9kg × 12.1cm × sin(30°) = 0.9 × 12.1 × 0.5 = 5.45 kg-cm

Servo torque available (worst case):

1. MG996R at 6V nominal: 11 kg-cm
2. At 5.5V: ~11 × (5.5/6) = 10.1 kg-cm
3. With -10% tolerance: 10.1 × 0.9 = 9.09 kg-cm

Safety margin = T_available / T_required = 9.09 / 5.45 = 1.67

2.4.3 Power System

In this section we'll discuss the risk of brownout when all motors/servo in stall condition. Such a condition brings max current thus lowest voltage that might cause ESP32 to shut down.

Current draw analysis (worst case):

1. 2× 16mm motors stalling: 2 × 3.5A = 7.0A
2. MG996R servo stalling: 2.5A
3. ESP32-S3 + MPU-6050 + LEDs: 0.35A
4. Total current: I_total = 9.85A

Battery performance under load:

1. Maximum discharge of 3s LiPo: 450mAh × 45C = 20.25A
2. Internal resistance: R_batt ≈ 150mΩ
3. Voltage sag: V_sag = I × R = 9.85A × 0.15Ω = 1.48V
4. Battery voltage under load: 11.1V - 1.48V = 9.62V

Buck converter to 3.3V logic rail: - Input voltage: 9.61V

1. Output voltage nominal: 3.3V
2. With -5% tolerance: 3.3V × 0.95 = 3.135V

ESP32 datasheet brownout threshold voltage is 2.8V, so we have 0.335V voltage margin, pretty close but still tolerance. We propose adding a 1000µF or larger capacitor near ESP32 to handle load fluctuations.

3. ETHICS AND SAFETY

3.1 Ethics

Our antweight battlebot is designed to apply force intentionally in a controlled competition, so our primary ethical responsibility is to protect people, property, and bystanders during development and operation. Guided by IEEE/ACM ethics principles (prioritize public welfare, avoid harm, be honest about limitations, and accept responsibility), we will: (1) choose designs that fail safely, (2) report testing results and failures honestly, and (3) operate only in approved settings. [1], [2]

3.2 Safety

Even at 2 lb scale, hazards include pinch/crush points, unexpected motion, electrical overheating, and battery risk. We reduce risk through layered controls:

1. Mechanical: run only in a test box/contained area, add hard stops to prevent over-travel/jams, use locking fasteners (threadlocker/nylocks), and guard any exposed moving couplers/gears. [3]
2. Electrical/Battery: include a manual kill switch, a link-loss failsafe that disables motors when commands stop, and hardware overcurrent/stall protection (FAULT/EN gating) to limit heating during jams; use appropriate wire gauge/connectors and strain relief; follow disciplined LiPo handling (correct charger settings, never unattended, inspect after impacts). [3]

Unintended motion is the biggest operational risk for a wireless robot. We will use: (1) a clear power/arming procedure (controller on first, robot powered/armed last), (2) visible status indicators (armed/link/fault), (3) automatic disable on link loss, and (4) an accessible kill switch. To reduce misuse, we will only operate in approved spaces, and transport/store the robot unpowered with the battery disconnected. [3]

3.3 Societal, Economic, Environmental, and Global Impact

1. Societal/public welfare: the failsafes, fault detection, and conservative power design used here directly map to safer real-world robotics and embedded systems. [1], [2]

2. Economic: strict mass/space constraints force realistic engineering tradeoffs between performance and reliability.
3. Environmental: we will minimize waste by iterating CAD before printing, reusing parts/fasteners, and disposing of damaged batteries properly.
4. Global/cultural: combat robotics is a broad community; emphasizing safety and transparent documentation supports responsible engineering norms.

4. REFERENCES

[1] IEEE, “IEEE Code of Ethics,” IEEE. Accessed: Feb. 13, 2026. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html

[2] Association for Computing Machinery, “ACM Code of Ethics and Professional Conduct,” ACM. Accessed: Feb. 13, 2026. [Online]. Available: https://www.acm.org/code-of-ethics

[3] University of Illinois Urbana-Champaign, ECE 445, “Antweight Battlebot Competition Rules / Proposal Requirements (course handout),” course materials. Accessed: Feb. 13, 2026.